

Jenny Carter
Michael O'Grady
Clive Rosen *Editors*

Higher Education Computer Science

A Manual of Practical Approaches

Second Edition

 Springer

Higher Education Computer Science

Jenny Carter · Michael O'Grady · Clive Rosen
Editors

Higher Education Computer Science

A Manual of Practical Approaches

Second Edition

 Springer

Editors

Jenny Carter 
Department of Computer Science
University of Huddersfield
Huddersfield, UK

Michael O'Grady 
Department of Computer Science
University of Huddersfield
Huddersfield, UK

Clive Rosen
Passerelle Systems
Newcastle-under-Lyme, UK

ISBN 978-3-031-29385-6 ISBN 978-3-031-29386-3 (eBook)
<https://doi.org/10.1007/978-3-031-29386-3>

© Springer Nature Switzerland AG 2018, 2023

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors, and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, expressed or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Foreword: Higher Education Computer Science in a Post-pandemic World

The preface of the first edition of this book began with the phrase “it is cliché to say that higher education is changing”. Little did the editors know then that a few years’ later higher education, and life in general, would undergo an unprecedented change as a result of the COVID-19 pandemic. Swift action was demanded from academics across the UK and the world. The initial obligation was to move to fully online teaching and assessment almost overnight to be able to complete the academic year 2019/2020 in the midst of a national lockdown. The same academics then had to prepare for a year where flexibility and adaptability were an absolute priority, to cope with the inevitable disruption from staff and student illness, together with lockdown reactions to further outbreaks of the virus.

On this extraordinary journey, computer science academics had to overcome potentially one less barrier compared to academics in less computing-oriented disciplines: their digital literacy skills. Such academics engage with technology as part of their discipline, and as such, they were more able to embrace the challenges posed by an immediate transition away from face-to-face teaching. Prior to the pandemic, technology was in the main used to supplement the delivery of face-to-face teaching. As soon as the effects of the pandemic presented themselves, technology was *central* to the delivery of learning, teaching and assessment. This transition, however daunting, became a catalyst for innovative practice with many groundbreaking paradigm shifts, which the cases within this book serve to explicate.

However, some pertinent themes have emerged as a result of the rapid transference to online learning and our subsequent immersion in it ever since.

The demise of traditional lectures. The pandemic has led many lecturers to gradually move away from traditional oral, in-person delivery in front of a classroom. After experimenting with various alternatives during the pandemic years, from bite-sized recordings to flipped classroom settings, more sophisticated, learner-centred approaches to teaching have emerged. Whilst the traditional mode of delivery is still present when it is important to explain a complex notion or threshold concept, it is now often interspersed with student-led and tutor-facilitated activities that bear some resemblance to elements of synchronous online during lockdown periods. Several benefits to students result from this change, such as improved, deeper understanding

due to active, practical learning rather than passively listening, and also increased confidence from working independently and as part of a team.

The importance of engagement. Many academics will have experienced the despair of sitting in a virtual meeting room asking a question to an audience whose cameras and microphones are switched off, and the joy of seeing a text message “pop-up” in the chat as a response after a few minutes, except that it felt like hours of silence. Situations like these led academics to explore different ways of making their sessions engaging and interactive, ensuring that students at home were actively participating and remaining interested and engaged throughout. Such good practice is applicable to any synchronous or asynchronous session, online or in-person and includes the use of interactive learning and presentation platforms such as Mentimeter and Kahoot to engage students. This is especially beneficial in the case of computer science students who often demonstrate more introversion perhaps than in other disciplines, and who might be less likely to respond directly to a question in front of others, but they are willing to engage through text-based chat or anonymised quiz responses.

Bite-sized learning. The aforementioned innovative delivery approaches have also had a significant influence in the way knowledge is packaged. Moving away from traditional in-person lecturing signifies a departure from large amounts of content delivered in one go, favouring instead bite-sized *chunks* of knowledge mixed with activities, quizzes, Q&A and other ways of assimilating knowledge. Bite-sized learning, in turn, requires a rethink of the way content is provided to students through virtual learning environments, encouraging more fine-grained clustering than large thematic blocks. This allows students to more clearly understand how different elements of taught material interact and the alternative learning journeys they can follow by navigating through content. From a tutor’s perspective, there is the added value of being able to more accurately track the progress of students, as well as understand which content receives more attention (or engagement) than others, or which topics are more difficult to comprehend.

Re-thinking examinations. A major challenge for academics throughout the pandemic and afterwards has been to prevent academic misconduct in examination-based summative assessment that is conducted remotely. In many computer science courses, this has led to a gradual phasing-out of most end of year examinations, in favour of coursework-based summative assessment and formative online quizzes. Of course, examinations may still play an important role in modules of a more mathematical nature, but in modules whether there is a practice-based element, such as software development, assessing through a portfolio or project-based coursework may improve attainment by allowing students to apply and hone their skills over a longer period, rather than being assessed during a one-off 2-hour session for example. It may also reduce the likelihood of collusion by, for instance, including an individualised component within the coursework, so that if students collaborate, they can learn from each other, rather than merely copy from each other.

Management and leadership challenges. In a post-pandemic higher education, effective leadership is ever more important and far more challenging. In the aftermath of more than two years of increased workload as a result of preparing material or

covering for missing staff, the risk of burnout has increased, especially if there is still disruption in one form or the other. Mitigating this risk requires managers to put mechanisms in place to maintain workload at reasonable levels, through robust staff contingency and continuity plans and moving towards leaner courses. In turn, workload rationalisation can improve the likelihood of maintaining the academic staff base even in times of high volatility by reducing staff turnover and increasing incentives.

On the whole, the pandemic period for computer science higher education has been as challenging as it has been transformative and despite the difficulties, especially at the beginning of the pandemic, rewards in the form of lessons learned and good practice have been significant. These add to a wealth of knowledge that preceded the pandemic and are still pertinent and influential. Computer science academics should strive to keep the best of both worlds, maintaining what is still suitable from before the pandemic and enriching their practice with successful initiatives that were borne out of the pandemic.

This second edition of the *Higher Education Computer Science* book is a manifestation of this, showcasing a wide range of practical approaches that have been developed by staff who did what they do best: making the teaching work.

George Baryannis
Department of Computer Science
University of Huddersfield
Huddersfield, UK
g.bargiannis@hud.ac.uk

Richard Hill
Department of Computer Science
University of Huddersfield
Huddersfield, UK
r.hill@hud.ac.uk

Preface

The preface to this second edition of this book discusses the radical changes that have been imposed on higher education as a result of the global COVID pandemic and the ensuing lockdowns. The sudden acceleration of the switch to a high proportion of online learning has changed the way in which higher education is delivered probably forever and has raised fundamental questions regarding the role of HE educators (a theme picked up in the postscript to the book). Many of these changes were in the wind anyway as the preface to the first edition noted, but the consequences of the pandemic have made it even more imperative that we question what we do and how we do it.

As the preface to the first edition noted, it is cliché to say that higher education is changing. There has been continuous change at least since the 1970s. Nevertheless, the changes that are occurring at the moment seem to be more profound and more widespread than ever. All institutions, however prestigious or uncelebrated are being affected. For some of the most prestigious, the shock of the change has been greatest, and this is new. Universities that have prided themselves on their research records are being asked to reconsider their teaching capabilities and their relationships with and attitudes to their students. They are being challenged on their records on student diversity. They are being required to justify their utility to the economy.

There are other forces at play. Increased competition for students between universities, (both globally and within country) and the requirement to be self-financing, are driving universities to satisfy prospective students that they offer value for money, both in terms of financial investment and the long-term skills students will need to prosper in a rapidly changing employment market. Technological change in the form of access to information, both about the universities themselves and the subjects they teach is powering a trend towards consumerism amongst students. People are asking the question “what are universities for?”. Teaching materials in the form of MOOCs are freely available and of high quality; and if not, there is always Wikipedia. Social media has just about extinguished the last vestiges of deference. Finally, the global pandemic has driven even more resources online and probably changed the balance of online learning and face-to-face contact forever.

In the light of all these social changes, government continues to weigh in (and waded in) to insist on accountability. Initially (in the UK), this was to justify expenditure on research. More recently, the Office for Students and the “Teaching Excellence Framework” have sought to establish measures of student experience and teaching quality. The validity and even the reliability of these measures have been questioned, but whatever their academic credibility, the truism that “whatever we measure we change” has already affected the university sector.

Computer science and its related disciplines have been more exposed to these forces than most subject areas. The industry-oriented nature of the subject has resulted in high volatility in student application numbers as market trends affect demand for graduates. Technical change within the subject area has caused curricula instability. Waves from almost mystical adoration of computers and computing to commodified dismissal (what’s the difference between a computer and a washing machine?) combined with perceptions of subject complexity and gender stereotyping, exacerbate the cyclical trends in subject’s popularity.

These generic factors together with subject characteristics such as its basic intangibility and intellectual complexity have made CS and its allied subject areas inherently difficult to teach. Large student numbers lead to diverse student populations. Poor coverage of the subject area at pre-university level results in bipolar distributions of subject knowledge amongst university entrants. The gap between physical constructs and the subject’s virtual concepts that create an intellectual schism students must navigate to make progress.

Teachers have been aware of these problems for many years and have tried various approaches to address the issues. Yet the increased pressure generated by the recent, intense scrutiny has meant that the urgency to find solutions has intensified further. This book gathers together a range of approaches that individual instructors have found helpful in addressing these common problems. These are practical applications that experienced practitioners have adopted to meet the needs of their students. The combined experience of contributors to this book is approaching 500 years. We cannot claim to have found solutions, that are unlikely to ever happen. But, by bringing together this community of practice in one volume, we hope to stimulate your own ideas, vitalise your teaching and enhance your practice.

The book is divided into three parts: *Approaches to Learning*, *Teaching: Examples of Practice* and *Employability and Group Work*. The *Approaches to Learning* part, whilst based on personal experience as is the whole book, offers some ideas about how we can move away from didactic delivery stage front. The *Teaching: Examples of Practice* part addresses some specific problem areas in teaching CS: programming, information systems management and design as well as some ideas about delivery to diverse student cohorts and automatic marking of programming work. The final part, the *Employability and Group Work*, does what it says on the tin, providing some novel ways of approaching employability.

Liz Coulter Smith’s opening chapter on student “multitasking” in the classroom, in some ways, does not quite fit with the rest of the book as it is not strictly focussed on computer science students. It is included here because (a) computing students are amongst the most likely groups of students to engage in multitasking and (b) by

looking at the changed culture and experience this current generation of students have grown up with, it sets the scene so well for the chapters that follow. Diane Kitchin's chapter on active learning offers one of the ways in which we can respond to these changes and address our students' needs and expectations. The "flipped" classroom, Michael O'Grady's chapter, provides an alternative approach. Clive Rosen's chapter offers a theoretical perspective on learning to program that presents an alternative approach to the program wars debate. Jenny Carter and Francisco Chiclana's chapter on distance learning and Thomas Lancaster's chapter on academic integrity address two issues that have arisen as a consequence of technological change affecting the classroom environment. Finally in this part, Marcello Trovati's chapter on teaching data science in context addresses the issue of the interconnectivity of computing with many other subject areas and advocates a multidisciplinary approach.

All the chapters in this part share a common orientation; they are student-centric rather than lecturer-centric. This attitudinal shift is one that might not sit comfortably on the shoulders of some staff, but we consider it to be essential if we are to engage with and maintain the engagement of our millennial students. Furthermore, if we can enhance the quality and quantity of engagement, we have a better chance of satisfying the expectations of other stakeholders as well as the students.

Part II, focussed on teaching, arises out of the knowledge and experience of the contributors to this book, of the teaching of computing. It addresses how we can best overcome some of the specific difficulties computer science students face in this most abstract, yet practical of subjects. Dave Collin's chapter offers an approach using graphics to overcome these difficulties with a smile. Steve Wade's chapter is similar, but focusses on information systems management and Carlo Fabricatore and Maria Ximena Lopez look at systems design. Arjab Singh Khuman's chapter takes a broader perspective on student engagement by looking at style rather than content (though he covers both). The section is rounded off by Luke Attwood and Jenny Carter offering some relief for hard-pressed markers of programming assignments. As student numbers increase, resources reduce and pressures on academics grow, this is something more of us may need to become more interested in future.

Part III of the book on employability and group work offers some guidance on how to embrace the employability agenda without compromising academic standards. The Enterprise Showcase outlined by Gary Allen and Mike Mavromihales offers one solution, whilst Clive Rosen's chapter on group projects provides a framework for decision-making regarding the running of group projects as well as some practical suggestions. Chris Proctor and Vicky Harvey suggest that satisfying employers' expectations compliments rather than compromises the learning process. Michael O'Grady and friends cover the question of managing year-long student placements, preparing students for it, supporting them during it and helping them readjust on their return. Sue Beckingham's two chapters complete this part by examining the skills and awareness today's students need. The first addresses the soft skills required by employers and the second, how to make students aware of their online presence and the importance of managing it for their future prospects.

The philosophy underpinning this book is that the relationship between student and instructor is fundamental to the success of the student. It needs to be built on

mutual respect and regard. We aim to maximise the achieved potential of students. The approach is facilitative rather than didactic, supportive rather than patriarchal. This may not suit all pedagogic styles or all students, but we believe that a transition from the traditional master/pupil approach is essential to meet the current and future demands of the educational environment.

One word of caution: students have not necessarily, and may not, buy into the contract of having to commit their own time and intellectual effort in order to be successful. This can be a source of conflict between student and staff. However, one of the implicit terms of the contract is that staff must commit to seeking the best approaches to support the learning of their students. This cannot be abrogated even if students don't keep their part of the bargain. We hope that this book will be of aid to teachers seeking to meet their obligations. There are many of us out there!

Two final points:

1. We are aware of the semantic controversy between the use of the terms pedagogy and andragogy, but we do not wish to intervene. In this volume, both terms are used interchangeably.
2. Similarly, the terms “teaching”, “lecturing”, and nouns “instructor” and “facilitator” are all used in the spirit described above, to support student learning.

We hope you find this book helpful, informative and, dare we say it, enjoyable.

Newcastle-under-Lyme, UK
Huddersfield, UK
Huddersfield, UK

Clive Rosen
Jenny Carter
Michael O'Grady

Contents

Part I Approaches to Learning

| | | |
|----------|---|-----------|
| 1 | Changing Minds: Multitasking During Lectures | 3 |
| | Liz Coulter-Smith | |
| 2 | Active Learning in Large Lectures | 17 |
| | Diane Kitchin | |
| 3 | The Flipped Classroom | 29 |
| | Michael O’Grady | |
| 4 | Applying Cognitive Theory to the Teaching of Programming: Metaphors, Robots and Problem-Based Learning | 51 |
| | Clive Rosen | |
| 5 | Distance Learning: Lessons Learned from a UK Masters Programme | 63 |
| | Jenny Carter and Francisco Chiclana | |
| 6 | Academic Integrity for Computer Science Instructors | 71 |
| | Thomas Lancaster | |
| 7 | Contextualisation in Data Science | 85 |
| | Marcello Trovati | |

Part II Teaching Examples and Practice

| | | |
|----------|--|------------|
| 8 | Using Graphics to Inspire Failing Students | 99 |
| | Dave Collins | |
| 9 | Best Practices for Teaching Information Systems Modelling | 113 |
| | Steve Wade | |

10 Promoting Design Thinking Through Knowledge Maps: A Case Study in Computer Games Design and Development Education 129
 Carlo Fabricatore and Maria Ximena López

11 Teaching Fuzzy Logic Utilising Innovative Approaches 149
 Arjab Singh Khuman

12 Semi-automating the Marking of a Java Programming Portfolio Assessment: A Case Study from a UK Undergraduate Programme 159
 Luke Attwood and Jenny Carter

Part III Employability and Group Work

13 The Enterprise Showcase Experience 171
 Gary Allen, Mike Mavromihales, and Jenny Carter

14 Task Versus Process: A Taxonomy for Group Projects 187
 Clive Rosen

15 Realising the Threshold of Employability in Higher Education 203
 Chris Procter and Vicki Harvey

16 Exploring the Landscape of HE Industrial Placements Within Engineering and Technology Subjects. Observations of Recruitment, Graduate Attributes and Student Experience 223
 Michael O’Grady, Fran Rimmer, Dan Everson, and Luke Swift

17 Baseline Skills—Scaffolding Soft Skills Development Within the Curriculum 253
 Sue Beckingham

18 Professionalism and Online Presence 275
 Sue Beckingham

Postscript 287

Index 295

Part I
Approaches to Learning

Chapter 1

Changing Minds: Multitasking During Lectures



Liz Coulter-Smith

Abstract This chapter takes a multidisciplinary approach to multitasking. Media multitasking has, consequently, become a frequent topic amongst academics yet some remarkable new research is revealing that we may not be taking into full account the changes to our students' ability to learn, given the changes to their brains. The risks of multitasking to student achievement have been well researched yet many of the positive related developments in neuroscience are less well known. This chapter reviews some of this research bringing together: information foraging theory; cognitive control; and confirmation bias as they relate to the multitasking Generation Z student in higher education. Some significant research findings are discussed, including using laptops and similar devices in the classroom. A small survey underpins these discussions at the end of the chapter, highlighting student perspectives on multitasking during lectures.

Keywords Multitasking · Cognition · Information foraging · Academic performance

1.1 Introduction

It is in our nature to do more than one thing at a time: to multitask. Multitasking feels good. Dopamine is released every time we turn to a new task (Strayer and Watson 2012). Our motivation to multitask is a natural human urge—we are foragers, and more recently in our technological history, information foragers (Pirolli and Card 1995).

Multitasking is defined as using two or more media concurrently. It is slightly different from task switching where one switches attention between two tasks. They are closely related, but for our purposes, we will define multitasking, sometimes referred to as media-multitasking, as involving at least one device that coincides with the “performance of two or more functionally independent tasks with each

L. Coulter-Smith (✉)
University of Northampton, Northampton, England
e-mail: liz.coulter-smith@northampton.ac.uk

of the tasks having unique goals involving distinct stimuli (or stimulus attributes), mental transformation, and response outputs”, (Sanbonmatsu et al. 2013).

Multitasking with various devices is also commonplace in university classrooms (Junco 2012). Three out of four students believe technology improves their educational experience and since 2015, 90% of students have both laptops and smartphones¹ (Statista 2017). Media device dependency, especially among 18 to 20-year-olds, shows 44% are compelled to access a device at least once every ten minutes (VitalSource/Wakefield 2015). These factors are profoundly impacting student focus, attention, distraction and consequently academic performance. Nonetheless, these factors are complex yet offer possible solutions that may require substantial shifts in thinking, both on the part of the student and the lecturer.

This chapter discusses why students are compelled to multitask particularly around information-intensive activities. The focus is on multitasking in the classroom of first-year university students but also attempts to understand the current multitasking debates including some problems involved in attention and distraction in the context of teaching computer science in higher education. This discussion then delves into a few of the recent studies in neuroscience to better understand the complex relationships that underpin multitasking. To summarise, this chapter seeks to expand the discussion on multitasking through the lens of a multidisciplinary approach to the topic. Through a small pilot survey at the end of the chapter, we gather data drawn from a group of first-year computer science students as first-hand evidence of the state of the debate.

1.2 Information Foraging Theory and Multitasking Check Para Numbering Here

We have to ask why humans have a compulsion to multitask? What is driving this urge? One theory stands out and helps make sense of this innate drive to multitask where we are in pursuit of information-intensive tasks. Understanding this problem from a behavioural position is vital given the context of teaching and learning in the classroom and given the increasingly sophisticated social and technological tools at the students’ disposal. Information foraging theory (IFT) was developed at the Palo Alto Research Center (PARC), to develop project models for the User Interface Research Area, this theory provided ‘novel’ information visualisation for searching and browsing (Pirolli and Card 1995, p.50). IFT goes some way to explaining our drive as humans to accumulate information. This theory is particularly important due to the level of information available to students and their drive to multitask and task switch. The IFT research team primarily used participants from the areas of business intelligence and MBA students. The team quickly realised the depth and variety of phenomena that needed to be dealt with when handling massive volumes of information, deadline constraints and complex search decisions in the context of uncertainty.

¹ Between 2011 and 2017 smartphone use doubled from 21.6 to 44.9 million in the United Kingdom.

Early on they realised they were dealing with something different from the standard human–computer interaction tasks originating from cognitive engineering models of the 1990s. Comparatively, they recognised the behaviours of people seeking information was largely determined or shaped by the architecture of that content, also referred to as the *information environment*. It was clear the participants’ behaviour was only minimally shaped by the users’ knowledge of the user interface. What is interesting here is how this model maps onto the classroom and the context of learning since Pirolli also found behaviour tended to be dominated by uncertainty and continual evaluation—a common attribute when learning a new skill or concept. Information foraging theory (IFT) was theoretically developed from optimal foraging theory (OFT) (Krebs 1977). OFT is largely a theory developed from predictive models of decision rules used by predators and originating from the theory of natural selection focussed on maximising food intake during foraging (MacArthur and Pianka 1966). Generally, IFT theory asserts that we have evolved to use information to solve problems that can pose a threat to us and our environment. Rather than forage for the food, we now forage for information. The theory goes on to explain that we have adapted cognitive solutions for survival. The technological need for survival forms a basis for human interaction with information technologies as demonstrated by the World Wide Web (Pirolli and Card 1995, p. 51). The earliest discussions about multitasking borrowed heavily from the biological sciences in this paper. The book that followed twelve years later, ‘Information Foraging Theory’ (Pirolli 2007) is a singularly foundational and influential work. More recently and no less important is ‘The Distracted Mind’ (Gazzaley and Rosen 2016) which further develops information foraging theory from a neuroscience perspective. These two works, bringing together information and neuroscience, place a plausible bridge for researchers attempting to explain the phenomenon of the human drive to multitask. If we consider this as a partial framework or model for further exploration, then there is a more positive perspective on multitasking than has previously been published since one can then view it as part of our natural evolution and adaptive ability to gather and make sense of increasingly large volumes of information and data in this era.

1.3 Multitasking Is Multidisciplinary

It became apparent that there was a need to expand this chapter beyond the issues of education and to consider the advances in neurosciences and cognitive psychology. It was apparent that media multitasking and its effects have been investigated exhaustively in many ways. “The problem of how the brain undertakes multiple tasks concurrently is one of the oldest in psychology and neuroscience” (Verghese et al. 2016).

In 2009 a summit at Stanford University’s Center for Advanced Study in Behavioural Sciences (CASBS) considered the impact of multitasking on learning and development. The purpose was to pull together a multidisciplinary, coherent and scholarly research agenda. Participants came from the field of neuroscience, child

development, cognitive science, communication, education, and business policy. Terms were agreed, including using the word multitasking itself, that multitasking had become a universal problem needing urgent attention. Solutions were being demanded by parents, educators, employers, workers, and marketers. Clifford Nass, a professor of communication at Stanford noted, “If you mention multitasking, people go insane—it’s all they want to talk about”. He described the problem of multitasking as “a challenge to human cognition” (Ophir et al. 2009).

1.3.1 Multitasking and the Brain

To better understand how distraction relates to multitasking we will explore a few aspects of neuroscience and our mechanical sensory capacities. To interpret multitasking, we need to consider the brain’s attention networks underlying our ability to switch tasks (Rothbart and Posner 2015, p. 3). Neuroimaging has recently revealed that even subtle shifts in tasks activate neural areas (Rothbart and Posner 2015). The cerebellum has two areas of operation one that uses sensory signals and the other uses motor signals. In effect, the cerebellum is our *motor* for learning, particularly when it comes to learning new motor skills (Hatten and Lisberger 2013, p. 2). The cerebellum is capable of plasticity allowing a neuron to communicate with another neuron (this is a simplified explanation) in dynamic ways. For the most part, the mechanical and sensory parts of the brain operate together as long as a single task is involved. However, introduce more than one task and communication between these parts begins to break down resulting in the grave consequences of driving and texting (Kramer et al. 2007). Most of us have experienced ‘going on autopilot’ and driving from one destination to another without being fully conscious of the trip. This is experienced since we were likely thinking about something else during the mechanical process of driving—the learned mechanical process of driving has been saved to memory. However, introduce another mechanical process, say picking up a mobile, or a third—using ones’ fingers to text, and even a fourth composing a text, and you have a recipe for disaster—the entire process becomes significantly diminished. In the United States alone nearly half a million people were injured or killed in accidents involving this combination of texting and driving (U.S. Department of Transportation 2017).

1.3.2 Action-Based Learning

The environment is significantly different in the classroom, still both the sensory and mechanical parallels for the brain exists with much less catastrophic consequences. Impaired listening or attention are significant to those trying to convey information to students who may be generally unaware that they are missing much of what is being

said. Recent research on plasticity regarding learning suggests that physical movement may activate the hippocampus in ways not previously understood (Cassilhas et al. 2016, p. 168). This discovery is significant concerning Action-based learning (ABL) approaches, since movement supports how the brain connects to preparing itself to learn. ABL is a process or pedagogy of brain-activated learning linked to the action of motor skills. This approach fits in well with the learner, requiring greater stimulus yet it has been observed that ABL is rarely discussed as a potential solution or even partial solution to the problem of distraction or inclination to excessively multitask in the classroom. (An omission that is addressed in this volume in Diane Kitchin's chapter on active learning.)

The problem of how we help students manage or break the cycle of multitasking in class may be diverted or rewired using methods like ABL. ABL requires substantial changes to the way lectures are planned and executed. The current state of most lecturing methods, where a long talk is involved, is yet another reason why lectures are becoming less able to facilitate learning and why ABL has come to the fore as one potentially rich approach.

1.3.3 Gen Z and Boredom

This year (2018) we will see our first Generation of students born between 2000 and the present: Generation Z (Gen Zers or Gen Z) has arrived in higher education. This generation was born into an Internet-connected world, has grown up with the smartphone, and may have spent the past decade using many social networks. The Gen Zers are a generation that prefers communicating through social media over direct communication. For the Gen Zers, waiting is not much of an option and they are conditioned to pick up their smartphone or device and find a release from the boredom. Since the arrival of the smartphone, waiting in lines at the store or for a train have become less of a problem. We can fill that time perusing the news, checking our social networks and email. Gen X and Zers use technology to 'personalise everything', they are technologically skilful and prefer Web applications and email (Reisenwitz and Iyer 2009, p. 91).

It seems logical that if students are physically active and working towards a goal or a solution to a problem they will be less likely to stop, pick up their phone and check Facebook—they will be less likely to interrupt their processes due to boredom.² This generation gets bored fast and the antidote to a nice hit of dopamine in checking in to social media. It activates them, and physiologically this generation has become accustomed to multitasking in this way in the same way that we would probably receive a similar hit from eating something satisfying.

² Our survey found 55% of students multitasked due to boredom. 62% identified lecturers reading from slides as another cause for multitasking during formal lectures.

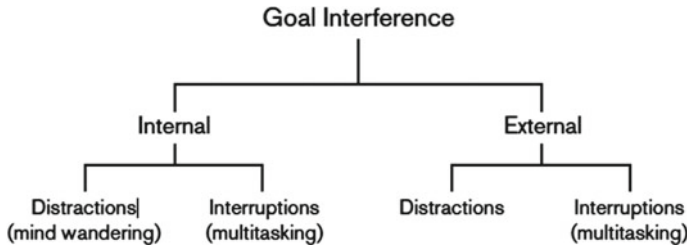


Fig. 1.1 Gazzaley’s conceptual framework for goal interference

1.3.4 Cognitive Systems and Control

Cognitive control and its functions are central to the concept of multitasking. Although we cannot pursue this in depth in this chapter, some basic concepts are considered. Gazzaley breaks this down into internal and external factors about interference (see Fig. 1.1 (Gazzaley and Rosen 2016)). Interference represents those things that distract and interrupt us whether of our internal making or externally driven. The brain is a complex information-processing system. As a system, it is structured and optimised for performance. Again, in Fig. 1.1, Gazzaley shows how goal interference competes with internal and external factors as we try to achieve our aims.

Students, however distracted, are just trying their best to achieve their aims with often incomplete information about how to manage themselves. Perhaps there is a need to help them understand how they can optimise their work through understanding some concepts around multitasking. As discussed earlier, they are living in a more distracted environment than existed a decade ago or for us as academics when we were taking a degree. Of course, experience and management of goal interference will likely swing widely between the individual depending on countless variations. However, several areas can be supported in the classroom by adjusting our teaching methods, by considering recent research, and by embracing rather than negating technological changes.

1.3.5 Confirmation Bias and Supertaskers

Another issue brought up in both our survey and anecdotal with first-year students is the role confirmation bias plays alongside multitasking. Over the past two years, there has been a higher proportion of students who believe they are ‘supertasker’s’ capable of rapid attention shifting with devices in what they believe to be efficiency. It is often talked about as a sought-after skill. It is true the way students often interact with a keyboard and respond to screen-based information is fast. Many studies have tested the supertasker phenomenon (Watson and Strayer 2010; Carr 2010). However,

current laboratory research still asserts that simultaneous task performance suffers during multitasking (Dux et al. 2009; Garner and Dux 2015). The problem is that speed and fluidity do not necessarily translate into the ability to apply it and learn new skills. Even more problematic, how do we help students to understand this when they believe what they have in a sense become indoctrinated into a cult of speed and freedom of unfettered access. Furthermore, studies of the brain have shown (Watson and Strayer 2010; Strayer and Watson 2012) there are only 2% of individuals capable of multitasking or do more than one thing at a time efficiently. However, employers seem to believe multitasking is a sought-after skill and regularly advertise it as such in programming jobs.³ Also, students see other students with similar behaviours in class and come to believe that doing more than one thing at once is expected of them, to further exacerbate this problematic issue.

1.3.6 Academic Writing: A Bridge Too Far

Writing is a higher-order learning skill. It is also an area where academics have seen significant and growing difficulties for students. It is possible that the rise of essay mills may well be related to the problems students are facing having to write an extended academic paper. If, as mentioned earlier, many students are experiencing a reduced depth of processing, increasing stress levels including anxiety due to multitasking, then their ability to invoke creative problem solving will ultimately be hampered (Firat 2013). We are finding that fewer students are often only capable of shallow focus work (Carr 2010) leaving them unable to tackle harder work requiring greater cognitive power and focus. So it is not only the focus, but academic writing is a difficult task that requires deeper thinking and higher cognitive skills than what current students spend most of the time doing, both inside and outside the higher education environment. These problems become most visible in the third year when they are asked to develop a dissertation, a large piece of writing requiring work over an extended period. They are less prepared for this challenge, and we need to do more to assist them constructively. It is likely that various variables are at work here from brain and neurological functioning to insufficient awareness and ability to manage distractions effectively. Add to this the inability to manage and focus attention in the sea of the increased use of social media, all these factors are contributing to the problem.

GOT TO HERE.

³ Searching the word “multitasking” site:indeed.com and “multitasking” site:indeed.co.uk show a difference of 73,300 US compared to 6760 UK. This may suggest a difference in educational and employment emphasis. It could also be just a reflection of population differences.

1.4 Debating the Banning of Laptops During Lectures

Moving now from our increased understanding of why students multitask and how the brain functions we can further explore the impact these are having in the classroom. One of the standout factors aligned to multitasking in the classroom is social media usage. In 2005 Facebook and MySpace were launched, closely followed by Twitter and YouTube in 2006. The exponential shift happened a few years after the launch of smartphones including the iPhone in (2007), and the 2008 launch of the Android mobile operating system in 2008. It then took several years for mobile computing to appear in classrooms where more than half the class were in possession of a mobile. In 2010 only a few students had them, but the increase has become pronounced and new problems around attention and distraction arose. Smartphones are cheaper, faster and the operating systems less fraught with technical issues. By 2015 virtually every student had a smartphone in the classroom and often more than one device. This fast pace has put stresses on the higher education system and our ability to adapt our teaching methods at pace with these changes. Combine this with social media usage, and one has a perfect storm.

By the fall of 2016, social media usage amongst university freshman in the United States averaged over six hours a week, an increase of over 40.9% or 27.2% greater than in 2011 and 2014 (HERI 2017, p. 20). There were over 10 million participants in this survey. Being an election year in the United States may have had some impact on this data. However, if social media continues to increase at a similar rate, what changes are likely in the classroom? Will increased usage of social network sites (SNSs) amongst students increase distraction and attention levels in the classroom and if so how will we adapt our methods? It is essential to develop strategies to improve engagement in this changing environment as well as considering both cognitive and information systems models as part of that development. As professors and lecturers across the globe experiment with various approaches to control these relatively recent changes, we see both extreme and light touch reactions. One wing demonstrates only a modest understanding of the collision of human-to-human and human-computer interactions at play. For example, Seth Godin⁴ taking an oppositional stance towards Susan Dynarski, a professor at the University of Michigan. Dynarski published an op-ed in the New York Times stating that she has forbidden students from using laptops in her lectures (Dynarski 2017). Godin believes Dynarski has missed the point altogether. According to Godin, Dynarski is laying the blame in the wrong place by asking students to slow down their clock speed and listen attentively in addition to notetaking—all at the same rate. He argues this is unreasonable to expect this given the technological changes in recent years and lays some blame on universities not adapting quickly enough either. Godin states “the solution isn’t to ban the laptop from the lecture it’s time to ban the lecture from the classroom” (Godin 2017). He also believes the lecture should be digitally recorded so students can review it, as and when they choose to. However, the problem may not require institutions to do away with

⁴ Seth Godin is a well known entrepreneur, bestselling author, writer and marketing and leadership blogger.

the lecture hall, and it is worth considering the possibility of something in between these two somewhat extreme ends of the spectrum. Shorter lectures formed of no more than five to seven minutes followed by activities to discover information closely related to the presentation may be more motivational and engaging. The traditional 45–60-min lecture is still currently the norm but is unsustainable given the changing environment. There are a number of arguments against banning laptops, not least of whether such a ban would be compatible with an ethos of open education and how such a ban might be enforced. There is the question of potential discrimination against students with disabilities, or if some students were allowed laptops to support their disability, discrimination against students without disabilities. Furthermore for “Zers” a laptop or smartphone may be the most efficient way to take notes and to instantly look up additional information. Some research suggests that students who multitask using their laptop during lectures perform less well compared to those that do not (Sana et al. 2013). However, one must ask—if students had more advice on *how* to take notes optimally, would this study still be valid? The early days of email usage in the mid-nineties had a pretty steep learning curve and compared to numbers of technologies and applications we have now it seems an almost silly comparison, yet we all struggled with learning how to manage it. Academic staff misused and overused the medium while simultaneously bemoaning the extra workload. We may have to consider students similarly don’t know *how* to manage their devices optimally to improve their performance. Sana’s study above was only investigated with forty participants. A limited sample suggests a need for a more comprehensive study that also considers using an intervention method as a control group and then comparing the data similarly to a study undertaken at Ryerson University (Tassone et al. 2017, p. 1).

1.4.1 Note-Taking

The research on note-taking goes back to the 1960s where there was considerable debate about how and when to listen and take notes (Eisner and Rohde 1959). It is worth having a brief look at how note-taking fits into the multitasking debate. Many researchers believe that taking notes on a laptop will impair performance compared to those who take notes longhand (Mueller and Oppenheimer 2014, p. 1; Bellur et al. 2015, p. 65; Fried 2008, p. 47). The problem is not the technology or mandating rules to comply with it. The problem is more precisely that students need assistance managing the interplay of these issues. Generally, most studies tend to support a rule or discipline-based solution in the classroom more or less finding fault with the student, the technology or the social media networks and default towards asserting that students must follow “proper rules [...] and abide by these rules” (Anshari et al. 2017). This approach mainly describes the problem but misses the importance of considering a model sensitive to context, changing cognitive conditions and human-systems design persistently shaping behaviour and influencing human evolution.

1.5 Smartphone Dependencies

Dependency on smartphones and academic performance is another area aligned with variables contributing to multitasking. There has been a plethora of research over the past decade on this topic (Samaha and Hawi 2016; Junco 2012, pp. 505–514) is well-documented students were almost in a state of discomfort having to turn off or look away from monitors or devices during the formal part of a lecture. There are many issues at work here. Firstly, students have become used to large amounts of visual activity and stimulus with the average 19-year-old checking their phone every ten minutes. Secondly, most students have had a smartphone for at least five years or more and lived in a context where these technologies have been an inseparable part of their daily lives. The smartphone has become an object of instant gratification, a quick fix for boredom and has neurologically altered their brains and consequent behaviours. Often this is leading to a form of addiction (Terry et al. 2016, p. 245). We can now confirm this has changed our students' brains having grown-up in tandem with smartphones and mobile computing more (Loh and Kanai 2015, pp. 2–3). If we can accept this, then much of what has been discussed in this chapter should begin to make sense. With this in mind try to imagine what a student would be experiencing in the average university classroom. Imagine how frustrating it would be to sit for extended periods while the lecturer reads from slides. This approach is still occurs in many lecture halls in both the United States and the UK. The lecture format will likely not keep students engaged unless it is short (5–7 min), targeted and has a specific outcome followed up quickly by an information consolidation activity. So, we currently have a problem, and it is not with the student—we are missing opportunities to create engagement in the classroom.

1.6 The Survey

A survey on multitasking was carried out between March 21st–31st 2017 on a cohort of 60 undergraduate students taking a first-year, core, web development module. The students were asked to describe their multitasking habits during formal lectures. The study aimed to discover perceptions about multitasking behaviour.

A Likert scale was used for 22 questions. A 23rd question asked if they would like to share their thoughts. The Likert scale was especially useful for establishing some evidence of a possible correlation between high percentages of neutral answers and whether questions were either too broad or vaguely stated. (The detail of these results has not been included.) The highest neutral score was 42% for the question: *I believe multitasking during lectures is a smart thing to do.*

1.6.1 Intrinsic Questions

Four questions were similar for a reason. These were questions about whether participants would change their minds about multitasking. 60% were willing to change their minds if multitasking proved to them it could: lower or improve their grades (66%), harm their learning (60%) or improve their learning (48%). 55% believed they could get more done with 43% thinking it made them more efficient.

1.6.2 Extrinsic Questions

Just 58% of the students said they were using one or more devices to multitask during their formal lectures. This result is generally in line with other studies. The reason for this appeared to be that they felt they could get more done 55%, while 62% said they multitasked because lecturers were reading from slides, while 55% said their multitasking was due to boredom during the lecture. In some ways, this is encouraging as a change in teaching approach may result in more active or participatory learning. No students felt any pressure to multitask by their lecturers (0%).

1.6.3 Employability

In 2012 at the CASBS summit, Clifford Nass stated: “companies now create policies that force their employees to multitask”. In our study, just 11.7% thought multitasking would make them more employable. This result demonstrates an opportunity to raise awareness amongst students for employability purposes. Oddly, 40% said they believed multitasking to be an essential skill. There has been an increasing frequency ‘multitasking’ appearing in job posts for software developers. This response is interesting despite evidence multitasking skills are often sought by employers. However, there is a difference in emphasis between the United States, and the United Kingdom⁵ in this regard. Oddly respondents did not consider multitasking to be an employability factor as highlighted in some research (Burak 2012; Crenshaw 2008).

The survey shows some evidence that computer science students in the UK have varied views on whether multitasking during class lectures is positive or negative. Though one comment did not see the point of the survey or why their views about it would be interesting. This response suggests students need more information about

⁵ Searching the word “multitasking” site: indeed.com and “multitasking” site:indeed.co.uk show a difference of 73,300 US compared to 6760 UK. This may suggest a difference in educational and employment emphasis. It could also be just a reflection of population differences.

this for their continuous and focused information-seeking behaviours about multitasking. Similarly, lecturers may want to alter teaching methods to reflect the changed cohort as mentioned earlier. Students also appear to want the facts about multitasking as there seemed to be some slippage between what they believe and what may help them in their studies and professional life.

1.7 Conclusion

Early on in this research project, it became apparent that the study needed to expand beyond issues of education and therefore consider the recent advances in the cognitive neurosciences and cognitive psychology. It also became clear that media multitasking and its effects have been investigated exhaustively in many ways. “The problem of how the brain undertakes multiple tasks concurrently is one of the oldest in psychology and neuroscience” (Verghese et al. 2016). What has been offered in this chapter is the breadth and depth of the challenge ahead and to some extent behind us as mediators in the classroom. Further advancements and changing frontiers in the sciences are still being discovered and how much Gen Zers brains have been altered is becoming apparent. However, as Susan Greenfield asserts “the brain is exquisitely adaptable” (Greenfield 2015) and further research will likely bring enhancements possible for our ongoing adaptation concerning information foraging. It is also possible that with these advancements there will be more ‘supertaskers’ among us (Strayer and Watson 2012). Video games are an indication of this and have been shown to be highly beneficial to multitasking particularly with older participants (Mishra et al. 2016). These developments indicate not all aspects of multitasking mean poor performance as some researchers assert (Bellur et al. 2015, p.65). Changes are underway that will continue to test us as educators though, and students will require specific and targeted guidance about the risks and benefits of multitasking as they manage their courses, careers and lives. However, I would suggest that there is one conclusion we can certainly draw. Multitasking is prevalent, and it is here to stay. We can either choose to rile against it, or adapt our methods to accommodate it. Accommodation would seem to be the more productive approach and possibly the less stressful one. It might well be worth considering how best to incorporate the changes learning environment into our teaching.

References

- Anshari M et al (2017) Smartphones usage in the classrooms: Learning aid or interference? *Educ Inf Technol* 22(6):3063–3079
- Bellur S, Nowak KL, Hull KS (2015) Make it our time: In class multitaskers have lower academic performance. *Comput Hum Behav* 53(2015):63–70
- Burak LJ (2012) Multitasking in the university classroom. *Int J Sch Teach Learn* 6(2):1–13

- Carr N (2010) The shallows: What the Internet is doing to our brains. Available at: <https://www.hispacultur.org/book/380556044/download-the-shallows-what-the-internet-is-doing-to-our-brains-w-w-norton-company.pdf>
- Cassilhas RC, Tufik S, de Mello MT (2016) Physical exercise, neuroplasticity, spatial learning and memory. *Cell Mol Life Sci: CMLS* 73(5):975–983
- Crenshaw D (2008) The myth of multitasking: how doing it all gets nothing done. Jossey-Bass: A Wiley Imprint
- Dux PE et al (2009) Training improves multitasking performance by increasing the speed of information processing in human prefrontal cortex. *Neuron* 63(1):127–138
- Dynarski S (2017) Laptops are great. But not during a lecture or a meeting. *The New York Times*. Available at: <https://www.nytimes.com/2017/11/22/business/laptops-not-during-lecture-or-meeting.html>. Accessed 23 June 2018
- Eisner S, Rohde K (1959) Note taking during or after the lecture. *J Educ Psychol* 50(6):301–304
- Firat M (2013) Continuous partial attention as a problematic technology use: a case for educators. *J Educators Online* 10(2):1–20
- Fried CB (2008) In-class laptop use and its effects on student learning. *Comput Educ* 50(3):906–914
- Garner KG, Dux PE (2015) Training conquers multitasking costs by dividing task representations in the frontoparietal-subcortical system. In: *Proceedings of the national academy of sciences*, p 201511423
- Gazzaley A, Rosen L (2016) The distracted mind: ancient brains in a high-tech world. The MIT Press, Cambridge, Massachusetts
- Godin S (2017) No laptops in the lecture hall—Seth Godin—Medium. *Medium*. Available at: <https://medium.com/@thisissethblog/no-laptops-in-the-lecture-hall-1847b6d3315>. Accessed 23 June 2018
- Greenfield S (2015) Mind change: how digital technologies are leaving their mark on our brains 4677(July):384. <https://doi.org/10.1080/00664677.2015.1021442>
- Hatten ME, Lisberger SG (2013) Multitasking on the run. *eLife* 2:e00641
- HERI (2017) The American freshman: national norms fall 2016, higher education research institute graduate school of education & information studies University of California, Los Angeles. Available at: <https://www.heri.ucla.edu/monographs/TheAmericanFreshman2016.pdf>
- Junco R (2012) In-class multitasking and academic performance. *Comput Hum Behav* 28(6):2236–2243
- Kramer A, Wiegmann D, Kirlik A (2007) Attention: from theory to practice. Oxford University Press
- Krebs J (1977) Optimal foraging: theory and experiment. *Nature* 268:583–584
- Loh KK, Kanai R (2015) How has the internet reshaped human cognition? *The Neuroscientist: a review journal bringing neurobiology, neurology and psychiatry* (January):1073858415595005
- MacArthur RH, Pianka ER (1966) On optimal use of a patchy environment. *Am Nat* 100(916):603–609
- McCoy BR (2013) Digital distractions in the classroom: student classroom use of digital devices for non-class related purposes. *J Media Educ* 4(4):5–12
- Mishra J, Anguera JA, Gazzaley A (2016) Video games for neuro-cognitive optimization. *Neuron* 90(2):214–218
- Mueller Pa, Oppenheimer DM (2014) The pen is mightier than the keyboard: advantages of longhand over laptop note taking. *Psychol Sci* 25(6):1159–1168
- Ophir E, Nass C, Wagner AD (2009) Cognitive control in media multitaskers. *Proc Natl Acad Sci* 106(37):15583–15587
- Pirolli P (2007) Information foraging theory: adaptive interaction with information, Oxford. Oxford University Press, New York
- Pirolli P, Card S (1995) Information foraging in information access environments. *Conf Hum Factors Comput Syst Proc* 1:51–58

- Reisenwitz TH, Iyer R (2009) Differences in generation x and generation y: implications for the organization and marketers. *Market Manag J*. Available at: <http://www.mmaglobal.org/publications/MMJ/MMJ-Issues/2009-Fall/MMJ-2009-Fall-Vol19-Issue2-Complete.pdf#page=112>
- Rothbart MK, Posner MI (2015) The developing brain in a multitasking world. *Dev Rev* 35. <https://doi.org/10.1016/j.dr.2014.12.006>
- Samaha M, Hawi NS (2016) Relationships among smartphone addiction, stress, academic performance, and satisfaction with life. *Comput Hum Behav* 57. <https://doi.org/10.1016/j.chb.2015.12.045>
- Sana F, Weston T, Cepeda NJ (2013) Laptop multitasking hinders classroom learning for both users and nearby peers. *Comput Educ* 62:24–31
- Sanbonmatsu DM et al (2013) Who multi-tasks and Why? Multi-tasking ability, perceived multi-tasking ability, impulsivity, and sensation seeking. *PloS one* 8(1). <https://doi.org/10.1371/journal.pone.0054402>
- Statista 2017 UK: Number of smartphone users 2011–2018. Available at: <https://www.statista.com/statistics/270821/smartphone-user-in-the-united-kingdom-uk/>
- Strayer BDL, Watson JM (2012) Supertaskers and the multitasking brain. *Sci Am Mind* 23(1):22–29
- Tassone A et al (2017) Multitasking in the classroom: Testing an educational intervention as a method of reducing multitasking. *Active Learn Higher Educ*:1469787417740772
- Terry CA, Mishra P, Roseth CJ (2016) Preference for multitasking, technological dependency, student metacognition, and pervasive technology use: an experimental intervention. *Comput Hum Behav* 65:241–251
- U.S. Department of Transportation (2017) Driver electronic device use in 2016. Available at: <https://crashstats.nhtsa.dot.gov/Api/Public/ViewPublication/812426>
- Verghese A et al (2016) Prefrontal cortex structure predicts training-induced improvements in multitasking performance 36(9):2638–2645
- VitalSource/Wakefield (2015) Fifth annual vitalsource/wakefield survey finds college students want more—and better—classroom technology. Available at: <https://www.vitalsource.com/press/fifth-annual-vitalsource-wakefield-survey-finds-college-students-want-more-and-better-classroom-technology>
- Watson JM, Strayer DL (2010) Supertaskers: profiles in extraordinary multitasking ability. *Psychon Bull Rev* 17(4):479–485

Chapter 2

Active Learning in Large Lectures



Diane Kitchen

Abstract An increasingly diverse student body combined with pressures to demonstrate excellence in teaching and improve results presents challenges for lecturers. Active learning techniques have attracted interest and discussion amongst educationalists. This chapter investigates the challenges and gives a practical guide to techniques that have been used effectively in a large lecture situation with first year students.

Keywords Active learning · Large lectures · Computing education · Constructivism

2.1 Introduction

Changes in higher education over recent years have seen a rise in the number of students going to University and consequently a more diverse student body. This has led to a need to help students make a smooth transition between school and University and to adjust to different environments, different delivery styles and different expectations. With increased competition for students amongst Universities, and the new demands of the Teaching Excellence Framework (TEF), comes increased expectations on lecturers—we must ensure our students pass our modules, and moreover pass them with good marks. Excellent and effective teaching is seen as a key factor by Universities in attracting students to their courses. The University of Huddersfield mentions the quality of its teaching in four out of six of the factors that demonstrate its excellence on its ‘About us’ page for new and prospective students. (University of Huddersfield 2018). Active learning pedagogies was one of the themes identified by the Higher Education Academy (HEA) in its review of the written submissions Universities included as part of their TEF documentation, in support of the assessment criteria for Teaching quality, Learning environment and Student and outcomes learning gain. (HEA 2017). In the HEA report on TEF 2 the authors state that “Course

D. Kitchen (✉)

Department of Computer Science, The University of Huddersfield, Queensgate, Huddersfield, UK
e-mail: d.kitchen@hud.ac.uk

design comes out as a prevalent aspect for providers upgraded to a Gold award Features mentioned in some statements included: ... use of active learning...” (Moore et al. 2017)

Additional challenges to providing an active learning experience are that group sizes are often large, particularly for first year teaching, where lectures may be delivered to groups of 150 students or more. At Huddersfield with this diversity in the student body we have seen an increase in the number and level of student support initiatives. We have also seen a shift in the type of learning activities available, with more studio work, project work and group work. Research in approaches to learning advocates a more learner-focused approach in teaching, with active learning being much-discussed over recent years.

In this chapter we focus on the practical techniques adopted to overcome these challenges and foster an active learning environment, particularly in a large lecture situation. The remainder of this chapter is structured as follows: Sect. 2.2 discusses the challenges and motivation for this work in more detail; Sect. 2.3 describes active learning and reviews some of the literature; Sect. 2.4 describes the specific, practical techniques used to deliver active learning in a large Computing Science & Mathematics lecture; Sect. 2.5 reflects on outcomes and the use of these techniques and looks at possible further development and Sect. 2.6 gives a summary and presents our conclusions.

2.2 Challenges and Motivation

The specific context for this discussion is the delivery of a Computing Science & Mathematics module to first year students. This is a long-running module, which has existed in some form or another for many years, and which has been delivered by me for almost 10 years. It covers topics such as set theory, graphs and trees, propositional logic, sorting algorithms, Finite state automata, grammars and languages, regular expressions, binary search trees and tree traversal algorithms. It is typically taken by a very large group of around 100–170 students. Invariably, there will be a very wide range of abilities in each group. Some students won't have done any Maths since GCSE two or more years previously and may only have a grade C. Others will be very able students who have studied A-level Maths or Computing or both, and so they may have covered some of the topics already. There will also be a small number of mature students, who could have been out of formal education for a number of years, and who may be nervous about the subject. There will also be a number of international students, who may have had very different educational experiences. Again some of them will have covered similar material before, while others may not. A large mixed-background, mixed-ability group presents many challenges. How can we keep the attention of and give new challenges to students who are familiar with a topic, whilst not overwhelming and alienating students who view the material as difficult and possibly scary?

There are a number of barriers to understanding in such a group. The cultural backgrounds and life experiences of students can vary widely, so I always try to bear in mind that not all examples will be appropriate for such a mixed audience. For example, once when teaching Java classes and inheritance to a mixed group of students, using what seemed to be a simple Bank Account example, I was puzzled as to why one of the international students was struggling to understand the material. It turned out to be because he was unfamiliar with the meaning of terms like Current Account, Savings account etc., rather than the programming language concepts. Some students may have been taught just to accept and learn what they are told, and will not have been encouraged to ask questions. Also many students just assume that maths is difficult or that they can't do it, and often don't recognise the difference between the general maths they studied at school and the discrete maths relevant to computing. With such a large and diverse group, it is important to keep all the students interested and engaged, and to make sure that the teaching materials don't provide an unnecessary barrier to learning or exclude any students. Even small things can make a difference—such as not choosing exclusively male or white British names in examples. So a simple set theory example of a set of students = {Bob, Tom, Harry} could make a student feel excluded if the examples all follow a similar pattern.

2.3 Active Learning

In the past, traditional lectures were often viewed merely as a vehicle to convey a large amount of information to a lot of students at one time. Students were expected to be largely passive, just listening and taking notes, and perhaps occasionally asking or answering questions. Research has shown that audience attention in lectures begins to wane every 10–20 min. This traditional approach has been questioned by educationalists and researchers, with theories such as Constructivism (Bruner 1960; Piaget 1950, and others) advocating active learning.

Constructivist theories see learning as an active process, whereby students build on and adapt existing knowledge or knowledge structures in the mind known as 'schemata', so developing and deepening their understanding. Knowledge isn't something that exists in isolation, it requires a context and connections to what we already know, so when we learn we are not just adding new information, we also have to make sense of it and give it meaning in relation to existing knowledge. "Learning is thus an active process of individual transformation and changes in understanding" (Fry et al. 2015, p. 65). Acceptance of this theory entails a change in the way teachers think about teaching and how they do it. Constructivists argue that the teachers must encourage a deep approach to learning, with context and expectations being set by the teacher. In addition, some researchers emphasise the importance of learning outcomes, arguing that the curriculum should be constructively aligned (Biggs and Tang 2009) with the teaching environment and assessment, and that students should be made aware of and understand these learning outcomes.

The question now arises “what exactly is active learning?”. There are many different definitions in the literature. The term active learning originates in the work of Revans (1971). Prince (2004) offers a simple and useful explanation, defining “the core elements of active learning to be introducing activities into the traditional lecture and promoting student engagement”.

Bonwell and Eison (1991) define active learning as “instructional activities involving students in doing things and thinking about what they are doing.” According to Weltman (2007) the activities can range from the very active, such as physically making something to doing something like playing a game or to the less-involved such as watching a video, and then applying what has been learned from this in some way. A commonly used lower-level example of active learning could just be allowing time in a lecture for students to think about or discuss the material presented. The argument is that since students can only concentrate effectively for about 15 min, introducing pauses two or three times in a lecture, or having some change of activity or focus is beneficial and a simple way of making the learning experience more active. Other commonly mentioned methods include problem based learning, and cooperative and collaborative learning.

We should also ask, before adopting an active learning approach, what evidence there is for its effectiveness. Weltman (ibid.) cites various studies (Raelin and Coghlan 2006; Sarason and Banbury 2004; Sutherland and Bonwell 1996; Umble and Umble 2004) which have found supporting evidence for the effectiveness of active learning techniques. According to a review by Prince (2004) there is evidence in favour of active learning. He cites Bonwell and Eison (1991) who “conclude that it leads to better student attitudes and improvements in students’ thinking and writing”. His review includes the work of Felder et al. (2002) who recommend active learning as “one of the teaching methods that work”. However, Prince also points out that the support for active learning from the literature he reviews is not conclusive and the improvements may only be small. Prince also reviews empirical support for active learning and concludes that “introducing activity into lectures can significantly improve recall of information while extensive evidence supports the benefits of student engagement.” (Prince 2004, p. 4).

2.4 Techniques and Practices

As previously mentioned, the techniques, practices and examples given in this section are all taken from a first year Computing Science and Mathematics module. The material used has been build up over a number of years by colleagues who taught the module previously (Ron Simpson, Barbara Smith, Lee McCluskey and John Turner) and myself. A lot of the material is theoretical, technical and factual, with little room for individual interpretation. Nevertheless, students have to build up their own understanding and mental models. So the aim is to make the lectures a mixture of factual delivery combined with various active learning techniques, to encourage engagement and facilitate learning.

The active learning techniques which are used are: -

- a. Motivation with real applications
- b. Worked examples on a Visualiser
- c. Students work on further examples in the lecture
- d. Animations and applets
- e. Short videos
- f. Active participation and enactment

Examples of the use of all of these techniques will be given in the remainder of this section.

Often in large lectures students are largely passive—just listening and perhaps taking notes. From the start of the term students are encouraged to be active learners in these lectures, participating in exercises and even activities. They should not think of themselves as empty vessels passively waiting to be filled with information and knowledge, but as active participants in a process. So it is made clear from the first week that they will be expected to take an active part in lectures, to bring pen and paper each week and to ask and answer questions.

Some students will see Maths as difficult, or a topic that they have previously found challenging, and some won't have done any computing science at all. They often don't initially see the relevance of theoretical material to their studies—"what has maths got to do with computing?" they may ask. It is important to build their confidence and to give them some motivation for learning what they may perceive to be hard and unnecessary. Thus a good starting point for every lecture is an example of a practical application within the real world or from computing of the particular topic to be studied that week. It also helps give some context for what may be completely new material. This, in terms of constructivist theory, helps them to begin to build or adapt a knowledge structure in their minds. For example, few students will have studied graph theory and will not link it with computing. Some concrete examples of the application of graph theory that could be given would be for modelling a network configuration or modelling dynamic processes, while trees are commonly used data structures they will meet in computing. For Finite state machines (FSM) a computer game would make a good example—where the states of the machine represent the states of a game, something almost all student will be familiar with. The states represented could be "explore", "attack", "evade" and so on with transitions between states labelled "player nearby", "no player in sight", "player attacks", "player idle" etc.

Once the context, motivation and applications are clear, these examples can then be followed by some definitions of the terminology or concepts being introduced. Wherever possible each concept or item should be accompanied by an example or diagram—e.g., vertex, edge, graph, tree. As with all written examples the Visualiser is used for this. In logic a simple example of a proposition is given—maybe something intriguing or amusing—e.g., an example of a simple proposition could be: "My real name is Trillian" (from Hitchhiker's Guide to the Galaxy), or perhaps something more current, but the idea is to choose something that is familiar to them from the everyday world. The students can then be asked to think of and write down their own

example, perhaps something relating to them, and either share it with the person next to them or tell the whole group, when asked.

The general pattern followed is exposition, example, activity, so, as soon as possible the students are given a small example or problem to work out. For example, when teaching set theory, after the concept has been explained and an everyday example given, they can then be asked to raise their hand if they are in the set of students with brown hair. Then the set of students with blue eyes, then the intersection (i.e., if they are in both sets) and so on. Another everyday example used is something like the “Who am I game” where students are shown a collection of pictures of men and women, with and without hats, glasses and beards and asked to determine the members of various sets, and the results of applying simple set operators. The idea is that the theory is more likely to be retained if they’ve been physically involved by putting their hand up, or writing a simple sentence etc.

As the lecture progresses the examples and exercises become increasingly technical. After introduction of all basic definitions and concepts I do a simple worked example with the students. I prefer to use a Visualiser for this rather than writing on a white board, because it is clearer for the students to see, as it is projected on to the main screen, and also I can still look at and talk to them while doing the example, rather than turning my back to them.

Two or three problems of increasing difficulty may be given for the students to work on. For example, three sentences in English to be translated into propositional logic, or using inference rules to prove logical arguments. I always do at least 1 or 2 problems with them first and then let them have a go on their own. Alternatively, if the exercise is more difficult or time-consuming, such as a more complex FSM, then I would talk through it, but involving the students in its construction, asking them what to do next at each step—e.g., which edge or label should be added next, where it should go etc., so continually encouraging involvement and engagement, and discussing why a particular suggestion might or might not work. I might ask “would adding a particular transition as a loop allow the generation of invalid strings, which would mean we should add a new state instead”? It is important to keep checking for understanding—encouraging a culture of asking questions (I say I don’t expect them to understand everything immediately and so I’m worried if they’re not asking questions).

For the more able students, hopefully the lecture serves as a reminder or refresher if they have covered the material before, and also a way to help them reinforce or add to what they already know. More challenging questions can be included in the tutorial exercises. These students often are more confident in answering questions or volunteering information in the lecture, which can help weaker students who might be reluctant to speak up.

A lecture on recursion will now be used as an example which incorporates many of the active learning techniques mentioned at the start of this section. The learning outcome is that students should develop an understanding of the working of a recursive algorithm—which then leads on in following weeks to understanding different recursive algorithms used in sorting or tree traversal. It begins with an everyday example that should build on existing knowledge. The context is then widened with

examples from computing and maths to help motivate further learning. Eventually there is a physical enactment of the algorithm by the some of the student followed by a more formal worked example.

The example from everyday life to introduce the idea of recursion is walking across a moor in the fog. If we just try to find our way across the foggy moor by attempting to walk in a straight line in one direction, we can easily get lost and end up walking round in circles. However, if we use our compass to identify a small feature or landmark about 100 m ahead in the right direction, such as a rock or hillock, which we then walk to, and keep following this strategy, we have broken our problem down into a smaller one which we can solve easily (walking 100 m). We can keep applying this method to the smaller problem that remains, as we are now 100 m close to our destination. We can write this as an algorithm for crossing the rest of the moor:

- If you can see your destination, walk to it;
- If not,
 - choose a landmark in the right direction 100 m further on and walk to it;
 - cross the rest of the moor.

This is a recursive algorithm; we have defined crossing the rest of the moor in terms of... crossing the rest of the moor.

A useful way to change the focus of student attention is to use short animations or applets within a lecture to illustrate a particular concept. This lecture next moves on to another example where recursion can be used—the Tower of Hanoi. There is a dynamic application showing the Tower of Hanoi (Dynamic Drive). This puzzle has three pegs and a number of discs of graduated sizes, each with a hole in the middle so that they can be placed on a peg. The aim is to move the tower of discs from one peg to another. Although we could physically move them in one go, this is not allowed by the rules of the puzzle:

- We can only move one disc at a time.
- We can only put a disc on top of a larger disc, or on the base board.

This puzzle according to legend, originates in a monastery in Hanoi (Vietnam), where the monks are engaged in moving a tower of 64 discs from one peg to another; when they have finished, the legend says that the world will end.

The application allows the user to select the number of starting discs, and then to move them one at a time from one peg to another. It also shows the minimum number of moves and the number of moves made by the user. I usually run this starting with three discs, moving them from the start peg to the finish peg, and then increase the number of discs by one on each attempt, trying to keep to the minimum number of moves. This can be quite stressful for the lecturer, once the number of discs increases, but you may be rewarded with a round of applause if you can do a sufficiently impressive number of discs without making a mistake—six is a good number to aim for! The students can also have a go at this themselves after the lecture or in one of the tutorials, to help increase engagement with the topic.

The demonstration can then be followed by showing the pseudocode for a recursive algorithm to solve the problem.

So far we have moved from a simple everyday example, followed by an outline recursive algorithm, to a demonstration, followed by a more detailed recursive algorithm. The next step is to ask students to take part in animating a more mathematical example of a recursive algorithm. The problem to be solved is calculating the factorial of an integer—8!

I explain that I ask a student (A) to work out 8! For me. He's not keen on multiplying, so this seems like a lot of work, but he can see that 8! Is the same as $8 * 7!$ He thinks that multiplying something by 8 would be manageable, so he asks a friend to work out 7! For him. The friend (B) feels the same way about multiplying as A, so she asks C to work out 6! For her, and she knows that all she has to do is multiply C's answer by 7. They carry on in this way: C asks D for 5!, D asks E for 4!, E asks F for 3!, F asks G for 2!, and G asks H for 1!

But finally, 1! Doesn't need any multiplying, so H knows that the answer is just 1, and passes this back to G. Now G has to multiply that answer by 2, and passes the answer (2) to F, who multiplies that by 3, and passes her answer (6) to E. E multiplies 6 by 4 and passes 24 to D; D multiplies 24 by 5 and passes 120 to C, C multiplies 120 by 6 and passes 720 to B; B multiplies 720 by 7 and passes 5,040 to A, who finally realises that multiplying by 8 isn't necessarily that easy but does it anyway and gets 40,320 to pass back to me.

I have pre-prepared A4 cards with the relevant numbers and calculations written on them. i.e., "Calculate 8!", "Calculate 7! And so on. I ask for 8 student volunteers and ask them to stand at the front of the lecture theatre. These cards are given to the students and we begin the animation of the algorithm, with each student passing the relevant Calculate card to the next in turn. When we reach the base case of 1!, the last student then hands back a pre-prepared card with the answer—i.e., $1! = 1$. The next student in line takes this, and then hands on a card saying $2! = 2 * 1 = 2$, and so on until finally card with the answer is passed back to me.

Following this change of focus and the physical activity, the students are then shown and talked through some Java code to calculate the factorial of an integer. After explaining the code, an example (5!) is drawn using the Visualiser, showing each recursive call until the base case is reached, and then the unwinding of the recursion until the answer is returned. All worked examples done on the Visualiser are made available after the lecture via the University VLE, along with the lecture notes and tutorial exercises. Lecture capture software also enables students to watch all or selected parts of the lecture again, facilitating further consolidation and reflection.

There are many animations and videos available; some useful examples follow. When teaching sorting algorithms there is an interesting animation from Toptal (n.d.) that gives a visual representation of several different algorithms sorting a number of lines of different lengths in real time, so that students can easily see the differences in performance, given different initial conditions. Also the Hungarian folk dancing representation of Quick sort on YouTube can give a useful break and change in focus and activity, and inject some humour into the lecture.

One of the key techniques is allowing students time in the lectures to do short exercises to reinforce what has been taught. For example, in propositional logic, after explaining how to translate from natural language sentences into propositional calculus, they will be given three small problems to solve. The first one I will do with them on the Visualiser, and I will then ask them to do the second problem themselves on paper. Students can either work alone or in pairs, allowing an opportunity for reflection and collaborative work. After a few minutes we will talk through the solution, and then the process is repeated for the third problem. These simple exercises allow the students to reflect on and apply the material presented to them. It can also highlight any areas they have not understood, so they can be encouraged to ask further questions as a result of attempting the exercises.

All lectures have an associated tutorial, where students can reinforce their learning and apply it to graded exercises. There is also the opportunity for collaborative work if they choose. The aim is for this knowledge to be useful to and used by the student in the rest of their University course and in their working life.

2.5 Outcomes and Reflection

How do we assess whether these methods work? Prince (2004, p. 2) suggests that the outcomes that should be considered include “measures of factual knowledge, relevant skills and student attitudes, and ...student retention “

We can look at the outcomes of the assessment, which for this module is a 2-h examination. We can look at Lecture capture star ratings and module evaluations. We can listen to feedback in student panels. From my own work I have noted that those lectures made available via lecture capture that get a higher star rating are the ones I've discussed here—that is propositional logic, recursion and sorting algorithms, which have the highest level of active learning content. Anecdotally, students on the whole are happy with the teaching they receive in this module as evidenced by their comments in our Student Panel meetings.

While the outcomes mentioned have all been good, they don't tell us which particular aspects of the teaching are the most effective. They don't tell us what may in fact work better, and they don't tell us which concepts students find the most challenging.

From my own experience, it seems that some students respond better than others to the active learning techniques discussed. For example, when asked to work on examples in the lecture, some students will work enthusiastically on the problem, discussing it and sharing solutions with their neighbours. Others are less willing to join in—perhaps finding the problem daunting or difficult. Doing a couple of exercises with them first can help, and I also try to give them an easy way to start—so, for example, when drawing a FSM, I suggest they think of the shortest possible legal input, and draw a diagram for that first. This may be as far as they get, but hopefully they get some sense of achievement from this. The worked solution then shows them how to build on this.

Videos and animations are generally well-received, and seem to hold everyone's attention. Obviously they have to be carefully chosen to be both interesting and informative. For the student recursion exercise it is useful to mention it to a few students beforehand, in one of the tutorials perhaps and check they will be willing to volunteer on the day, otherwise it can take a little while to encourage students to stand at the front. However, once everything is in place, it engages the audience.

We might want to consider which students benefit the most. All students should benefit to some extent from being more engaged with what is going on—active learning should be more interesting than just sitting there. Students who don't benefit tend to be those who haven't engaged with the course or the module, or those who have pre-conceived ideas about theory and maths, and believe that they can't do it. Additionally, students who have been absent and not caught up or are finding the module difficult may find the in-lecture exercises too hard. Extra optional support sessions are provided on a weekly drop-in basis to help with this, but it is difficult to ensure that all the students who need them attend. There will always be some students whose reaction to something being difficult is to ignore it and stop attending. Some very able students may find the exercises too easy, but can be given additional advanced exercises to do in the tutorials or in their own time. Ideally for all students their mental knowledge structures should continue to develop as they build on what they have learned in the rest of their degree course. The literature provides some support for the effectiveness of active learning, although many reviewers have identified a need for further study. The conclusions given by Prince (2004, p. 7) are that there is "support for all forms of active learning examined ... students will remember more content if brief activities are introduced to the lecture".

What pitfalls are there in adopting an active learning approach? There is a risk that if the exercises are too difficult then the students will just use it as an opportunity to chat amongst themselves. The amount of time given to work on an exercise should be limited, and the lecturer should intervene and go through the example if it seems that students are struggling or not working effectively. It can be difficult to gauge what is the correct amount of time to allow to complete an exercise, as not all students work at the same pace. But by keeping a close eye on who is doing what, who is writing, and general noise levels, it can be done.

It is also important to set the right initial tone and atmosphere as far as possible. So normally students are expected to listen and not chat or disturb other students—but when doing an exercise, they can discuss things amongst themselves. However, once the lecturer begins going through the answer, then their full attention should be expected. Thus changes in activity are accompanied by changes in pace, tone, atmosphere and noise levels. Technology can also cause problems—so any links to videos or animations should be checked before the lecture, in case they no longer work.

It might be asked if content has to be reduced to incorporate active learning. Over the years that this module has been delivered, some topics have been removed through natural wastage, for example, because they no longer feed into specific second year modules—so the removal of a second year Formal Methods module resulted in material on relations and functions being omitted. Such changes in content allow for

an increase in the amount of active learning that can be included. Material can also be restructured and reworked to allow for the inclusion of active learning material—perhaps replacing a wordy explanation with a shorter one accompanied by a video or animation, for example.

Finally, if we want to extend the use of active learning techniques, then further development could include the use of a Student Response system for students to give answers to questions to assess their level of understanding. This might help make it clearer to the lecturer which specific aspects or topics students are struggling with. A system developed at Huddersfield enables teachers to initiate questions, and students to respond using their own mobile devices such as phones, laptops or tablets. It allows data to be collected and automatically stored for future retrieval (Meng and Lu 2011). The system can be used in activity-based or problem-based educational settings regardless of the size, age or knowledge background of the learning group. This could provide useful information as to where additional explanation or examples would be beneficial, and help better assess the usefulness of the active components, as well as actually providing an additional active component in the lecture.

2.6 Summary and Conclusion

This chapter has discussed the challenges of teaching large, diverse groups of students, it has described active learning, looked at the motivations for its use and reviewed some of the literature. We have presented the specific, practical techniques and examples used to deliver active learning in a large Computing Science and Mathematics lecture and then considered the outcomes and potential pitfalls of using these techniques along with possible further developments.

References

- Biggs J, Tang C (2009) Teaching for quality learning at university, 3rd edn. Open University Press, Maidenhead
- Bonwell CC, Eison JA (1991) Active learning: creating excitement in the classroom. ASHEERIC higher education report no. 1. George Washington University, Washington DC
- Bruner J (1960) The process of education Cambridge. Harvard University Press, MA
- Dynamic Drive (no date) The tower of Hanoi retrieved from <http://www.dynamicdrive.com/dynamicindex12/towerhanoi.htm>
- Felder R, Brent R, Stice J (2002) National effective teaching institute: workshop materials. In: American society for engineering education annual conference. Montreal, Quebec, Canada
- Fry H, Ketteridge S, Marshall S (2015) A handbook for teaching and learning in higher education: enhancing academic practice. Milton Park, Abingdon, Oxon, Routledge
- HEA (2017) TEF2 Infographic Retrieved from <https://www.heacademy.ac.uk/knowledge-hub/evaluating-teaching-excellence>

- Meng Z, Lu J (2011) Opportunities of interactive learning systems with evolutions in mobile devices: a case study. In: Proceedings of the 2011 international conference on internet computing ICOMP 2011. CSREA Press, pp 238–244. ISBN 1601321864
- Moore J, Higham L, Sanders J (ARC Network) in partnership with partnership with. In: Jones S, Candarli D, Mountford-Zimdars A (eds) Evidencing teaching excellence analysis of the teaching excellence framework (TEF2) provider submissions. Retrieved from <https://www.heacademy.ac.uk/knowledge-hub/evidencing-teaching-excellence>
- Piaget J (1950) The psychology of intelligence. Routledge and Kegan Paul, London
- Prince M (2004) Does Active Learning Work A review of the Research. *J Eng Educ* 93(3):223–231
- Raelin J, Coghlan D (2006) Developing managers as learners and researchers: using action learning and action research. *J Manag Educ* 30(5):670–689
- Revans RW (1971) Developing effective managers; a new approach to business education. Praeger Publishers
- Sarason Y, Banbury C (2004) Active learning facilitated by using a game-show format or who doesn't want to be a millionaire? *J Manag Educ* 28(4):509–518
- Sutherland TE, Bonwell CC (1996) Using active learning in college classes: a range of options for faculty in education (2008) *Academy of management learning & education* 7(1):139–142. Retrieved from <http://www.jstor.org/stable/40214510>
- Toptal (no date) Sorting algorithm animations. Retrieved from <https://www.toptal.com/developers/sorting-algorithms>
- Umble M, Umble EJ (2004) Using active learning to transform the monte hall problem into an invaluable classroom exercise. *Decis Sci J Innov Educ* 2(2):213–217
- University of Huddersfield (2018) About us. Retrieved from <https://www.hud.ac.uk/about/>
- Weltman D (2007) A comparison of traditional and active learning methods: an empirical investigation utilizing a linear mixed model, PhD Thesis, The University of Texas at Arlington
- YouTube Quick-sort with Hungarian (Küküllőmenti legényes) folk dance. Retrieved from https://www.youtube.com/watch?feature=player_embedded&v=ywWBy6J5gz8

Chapter 3

The Flipped Classroom



Michael O'Grady

Abstract A first year introductory internet and digital media module, taught as an in- and out-service across two Schools and covering a wide range of music courses, resulted in attendance and completion problems on a yearly basis. Students ranged from technical/programmers through to musicians and performers. Keeping them all engaged, motivated and appreciative of the learning materials was a challenge. The Flipped Classroom approach of front-loading weekly learning materials on to the VLE and providing more time for structured tutorial and computer lab work provided a distinct improvement in student marks and satisfaction scores. This Chapter assesses student marks against attendance and VLE engagement from this approach over one academic year. The learning materials provided a broad richness of weekly-themed resources including: screencasts; text; PowerPoint slides; screencasts of slideshows; scholarly articles; blog posts; web articles; and embedded YouTube videos. Students were encouraged to read broadly and assimilate as much as possible from the varying formats of presentation whilst having weekly structured tutorial content. The weekly content creation/linkages were designed to provide additional materials for those on the technical spectrum and additional content for the creatives. Student marks are compared against attendance and also against VLE engagement over the year. Whilst both indicate a benefit with increasing attendance and VLE engagement, there is a greater parity with the latter—attendance in class is less important than perusing the weekly resources. Students were able to perform to first class standards whilst engaging fully with the VLE and having a less-than 50% attendance in class. A broad section of students engaged with materials before the weekly lab sessions and some engaged with hardly any tutorial work at all. VLE access data also shows many students engaged to the early hours of the morning. The Flipped Classroom approach resulted in a better module results as well as an improved classroom experience: greater focus on the learning materials and tutorial exercises; much less time surfing and social networking in class; and more time in class for one-to-one assistance.

M. O'Grady (✉)

Department of Computer Science, University of Huddersfield, Huddersfield, UK

e-mail: m.ogradey@hud.ac.uk

© Springer Nature Switzerland AG 2023

J. Carter et al. (eds.), *Higher Education Computer Science*,

https://doi.org/10.1007/978-3-031-29386-3_3

29

Keywords Flipped classroom · VLE · Attendance · Engagement · Learning resources

3.1 Introduction

This Chapter discusses the introduction of a Flipped Classroom approach to teaching first year digital and internet technology content to music students during the academic year 2014/15. The module, entitled 'Introduction to Digital Media and the Internet' provides broad coverage to a wide range of music students across two Schools (Computing & Engineering as well as Music, Humanities & Media, C&E and MHM respectively). Some courses had this module as core and others as optional. Around 70 students took the module in 14/15, spread over four groups of 1.5 h sessions per week based in computer labs.

Student backgrounds and courses ranged from the computer-programming BSc Music Technology students who wanted technical detail and coding through to musical artists and performers on BA Popular Music production. The latter were interested in setting up websites, content, promotion, video for marketing and visually expressing their own or their band's music persona, whilst the former were unmotivated to anything 'arty' and performance related, wanting instead to learn code and technically manipulate audio.

The module, based in C&E, was an in-house service module to Engineering colleagues and an external service module for students studying music within MHM. The module was intended, by their various Course Leaders, to be 'all things to all students' in the area of digital/multimedia. However, the reality hadn't live up to the billing, mainly because of the diversity and expectations of students and in part from a demotion in its importance by having the two hours contact time reduced to 90 min. The lost 30 min being assigned to a more difficult and core module on each course. As a result, one of the recurring complaints was that students didn't have enough time in class for one-on-one support to understand and complete their weekly tasks.

This module had been running for over 10 years. The different module leaders taking their own slant on what was taught within the specification. Different versions of the module covered the following:

- technical website production (coding from scratch)—good for techies, bad for performers;
- marketing, visuals, video and DVD production (Photoshop and branding from scratch, lots of visual assets produced)—bad for techies, better for performers;
- making content managed websites to a professional standard, publishing of image and blog content and video creation—the author's approach, good for some, bad for others.

Having taught the module for several years with relatively low levels of student satisfaction (see below), the flipped classroom approach seemed intuitively like a good direction to take. The basic premise was to:

- provide a weekly structured diet of rich and broad content available for pre-reading before the class session;
- some content would overtly appeal to the creative/performance music students and some to the more technically-minded;
- provide an adaptive-release set of weekly tutorial tasks to be completed before or in the class. Initially invisible, students gain access by working through the weekly pre-reading and click the *Reviewed* adaptive-release button at the bottom;
- minimise the taught/instructional aspects of the 90 min weekly contact and maximise tutorial one-on-one assistance.

The flipped classroom approach was to provide an increased amount of content for each side of the technical/art divide. It required much in-class support and verbal direction (individually and class-wide), with constant prompting and reminders on approaches to take with assignment work. In addition, each student was provided an opportunity to extend their learning (within the direction of their course) with experimental work. Each of the two termly and equally-weighted assignments allowed up to 20% of marks to be assigned towards experimentation—areas of work and learning defined as anything above/beyond the specific course content.

The result of this work was a doubling of student satisfaction from below 50% to around 90% from using this approach. Additionally, the author had access to a relatively large amount of corroborative data from different student-monitoring systems. Although the biggest disappointment was data available via UniLearn, our branded version of the Blackboard™ VLE.

The results (see later) show some interesting findings regarding student behaviour and performance by comparing data from:

- the two equally weighted assignments and their average;
- attendance monitoring of students in timetabled sessions; and
- click-through engagement with different elements of the VLE.

Further data, not covered in this Chapter, related to engagement with the module-specific reading list (MyReading) resources; formal module evaluation questionnaire; and a custom questionnaire.

This Chapter therefore presents an overview of how the module was structured in the VLE, how it was taught, what the students thought, author's conclusions and backed up with a broad range of data.

3.2 The Module

The module provides a broad-brush approach to dealing with the internet, digital content and multimedia as well as providing an introduction to marketing and promotion of individuals/personas/bands/groups via websites, video and social media. The content was tailored specifically to students in the music industry.

The diverse student backgrounds and reduced content time mentioned above provided a challenging learning environment.

Approximately 85 to 90% of the cohort come from the more technically-orientated BSc-type courses within our School and 10–15% from MHM. The service arrangements were managed by Engineering colleagues who also managed MHM student attachment to the module. The decision to reduce weekly contact hours from 2 to 1.5 h was made so that other core modules (mainly programming for the technical students) could receive an additional 30 min weekly.

The main challenge was to keep all the students on all six courses challenged and engaged, however the courses ranged from BSc Music Technology and Software Development through to BA Music Technology and Popular Music. The latter group including students who were studying music and leading on instruments such as piano, brass, wind, harp etc.

Unfortunately, the module has been polarising student satisfaction for many years; it was not enjoyed by all students, seen as being unchallenging for some and too difficult and technical for others. For many, they developed digital life-skills to help promote a music career that wasn't yet fledged and most enjoyed the video element in term 2. So the challenge was to take roughly the same taught content (agreed by the various course leaders) and make it more engaging and useful to all the students.

3.3 Motivation and Justification of the Flipped Classroom Approach

During 2010 to 2013, the author was engaged in providing screencast lecture content and summative feedback for students and wanted more exposure to technologies that utilised video and voice recording, not least to gain enhanced broad digital literacy skills. As part of this interest, the author attended an internal Facilitating Online Learning course in 2013. It was designed for those wanting to undertake distance learning courses and gain exposure to cutting-edge webinar technology. The desire to attend was primarily to gain exposure to the webinar software (Adobe Connect) and to become more comfortable in front of a webcam for live, recorded virtual meetings and screencasts.

Whilst the webinar element was very empowering, the presentation of materials via a bespoke VLE module was itself presented using the flipped classroom approach and demonstrated an engaging learning environment. This approach entailed:

- Pre-reading of set materials **before** entering the virtual webinar meeting, including occasional completion of tasks (Wikis, Discussions etc.);
- Pre-reading included academic papers, blog articles and embedded video resources;
- Each week's learning materials centred around the practical use and engagement of the VLE and webinar software;
- All class sessions were hosted remotely on the webinar software;

- A set of webinar facilities was demonstrated: chat, polls, live voice and camera; and
- The facilitators also presented applications from their desktop through the webinar software (Word, PowerPoint etc.).

Over 20 years ago, King (1993) referred to the traditional taught lecture as the ‘Sage on the stage’ and commented that whilst some subjects work well with this approach, there was merit in learning stemming from more of a facilitation process, referred to as the ‘Guide on the side’. The flipped classroom approach identifies with the latter and is an instructional approach, combining a greater mix of delivery methodologies of learning resources, including student discussions, group working, collaboration on online tools and encouragement to reflect more. In effect, potentially covering all of Bloom’s Taxonomy (Gilboy et al. 2015).

In their work with HE students in Public Health, Galway et al. (2014) found that students generally liked the flipped approach to learning new content. Over 80% of students preferred the flipped classroom with blended online learning materials and problem-solving exercises in a classroom environment. Some students commented that they were more aware of subject-related news articles and more reflective about what the new material meant and how it related to learning in other areas/modules. Some of their students also commented that they felt obligated to do the pre-reading before arriving in the session because it is part of the course set-up, otherwise they tended not to engage with reading. Some students can initially have difficulty and/or resent what they perceive as too much reading/learning in the early weeks of such a course and this can impact negatively on student satisfaction (Critz and Knight 2013; Missildine et al. 2013), although this impact tends to lessen as they become used to the pre-loading of weekly work over the course of the academic year.

Significantly improved student satisfaction is also a key factor in the flipped classroom for High School students (Sergis et al. 2018), compared to traditional taught environments. Students gained a greater sense of accomplishment and incentives to participate in the learning process. Lower-performing students gained the highest levels of satisfaction and also benefitting from being more able to engage in deeply collaborative work, probably because of the wider window for improvement.

There were some personal challenges in following the course, many of which would be mirrored for the students:

- Time management of having to do the work before class. The system is less effective if students don’t do the learning;
- More reading and learning over short periods than the author was used to—it was quite a culture shock, something that would be more challenging to students;
- The possibility of having progress and non-progress monitored; and
- Increased contact with colleagues from other Schools—usually infrequent and unplanned but always pleasurable and advantageous.

In short, the flipped classroom requires much time and effort up-front, doesn’t necessarily work well if students don’t engage and is totally different to most learning takes place (...during and after the session).

3.4 Establishing the Flipped Classroom Approach

All content was placed on our Virtual Learning Environment (VLE). The University has a formal structure for the menu items in all modules and are as follows:

- Announcements (allowing *sticky* messages to remain at the top);
- Module Information (the module handbook);
- Staff Information;
- Learning Resources (the main weekly learning and tutorial area);
- Assessment; and
- Reading List (branded as *MyReading* internally).

There is now an additional menu item covering lecture Screen Capture which was introduced in 2015 and is not covered in this Chapter.

The Learning Resources area is arranged at the discretion of the module leader and is the main point of call in the teaching week. For this module, it was arranged weekly (teaching weeks rather than admin weeks) and released for viewing the previous week.

For each week created, the folder/container is placed at the top of the page and numbered with the teaching week (1 to 12 and 13 to 24, for terms 1 and 2 respectively) and titled appropriately. Students should always be aware of what week they are studying in and what weeks, if any, they have missed—the current week is always at the top.

Each week has three sections:

- An Introductory item with text and possibly containing an embedded descriptive screencast. An introductory screencast was created for several weeks at the beginning and soon dropped due to workload and propensity for it to become outdated the following year;
- A folder with an introduction to the learning content. The folder contains between 4 and 8 separate sequential bite-size items of content. The last item contains a button (*Reviewed*) that, once clicked, forces the Weekly Tutorial work to become visible and available to the individual student;
- Weekly Tutorial folder—a sequential series of tasks for students to undertake ideally before or during (and after) the weekly session.

All elements (folders, items and adaptive-release buttons) were activated for data recording to determine if a student clicked that particular link. This data was then used to assess VLE engagement and individual performance across the assignment work. Below is a closer look at the two main content folders.

3.5 Weekly Learning Content Folder

Each weekly content folder has a set of instructions and guidance prior to students entering. The weekly objectives, learning milestones, linkage with previous weeks' content and advice for assignments can all appear in this introduction text.

Items populating the folder are intended to be viewed sequentially from top to bottom by each student. Learning materials include:

- embedded graphics to demonstrate what type of item resource they were looking at (text, web, video);
- text within the item;
- text and links to uploaded files (Word, pdf, spreadsheet, slides);
- hyperlinks to one or more web resources;
- embedded video—either via the VLE video insert widget feature, or more reliably, via a direct YouTube (or similar) iFrame HTML *embed* code so that the video both displayed and played directly from the page; and
- Screencasts (generally uploaded to YouTube by the author and set to *Unlisted*) explaining some concepts, working through a PowerPoint slideshow or showing how a tutorial exercise is completed.

Some older slideshows (typically no more than four years) were included in the resources as additional reference items with specific guidance to currency, applicability and comparisons with current screencasts, as appropriate.

3.6 Weekly Tutorial Tasks Folder

Initially entitled 'The Learning Module' (a direct copy of the Facilitating Online Learning course format), the title was soon changed to Weekly Tutorial Tasks. This folder takes students through the weekly tutorial work in a step-by-step way with indexed pages presenting small elements of work; a sort of 'do this, do that' approach. There was no new learning content in this section although students were exposed to new techniques with software applications. Also, any screencasts in this folder were more specific about using tools and applications in relation to the weekly tasks and assignments.

After the introductory weeks of a new topic and associated software application(s), students sometimes had difficulty performing the weekly tasks. There is a cognitive disconnect between watching a live demo on the projection screen in class, then remembering the steps and doing the same for themselves on a PC. The Flipped Classroom approach was chosen in part to provide more time in class for this purpose, but non-attendees, less-able and less-motivated students can struggle to grasp the essentials by the end of a session. So, screencasts of the author working through the task solutions were created and made available a week or two later. Students would be reminded at the start of each relevant class that a previous week's solution was

available as a screencast at a certain location within the VLE. It is good to avoid this being a predictable resource to ensure students don’t rely on it as a means to avoid engagement and learning.

Using special ‘Catch-up’ or ‘Problems Solved’ screencasts were also created approximately once per term to cover common issues. These would appear as items in the weekly chronology of the Learning Resources page. The screencasts might show answers to common questions from class sessions, email questions and always bring the subject around to implications for the coming assignment.

Once activated, the initially invisible Weekly Tutorial Tasks folder is similar in appearance to the learning folder in that the student is expected to work their way down the stated activities. These are very pragmatic instructions such as Open an Adobe Application with the sample file, then make some additions, alterations and experimentation, save and view the work. The author has developed a shorthand for sequential instructions where the symbol ‘>’ is used to concatenate follow-on instructions. An example for making a hyperlink in MS Word is shown below:

1. ...*previous instructions*...;
2. Open a web page in a browser window that will be linked from your document;
3. Select the **URL > Ctrl + C**;
4. In Word, highlight the words(s) providing the **link > Ctrl + K > Ctrl + V > OK > Ctrl + S**;
5. ...*continue*...

There is a very focused chronology of weekly tasks expected of the students, particularly because the finished results of work can vary greatly if the order of work undertaken changes in some interfaces. Sometimes the tasks comprise quickly-produced bullet points with a fuller explanation provided in class or by embedded screencasts.

Some students would have undertaken the necessary learning and attempted the tutorial work in one or more sessions. At the other extreme, some would not have activated the weekly tutorial folder would be unlikely to have read the planned materials before class.

There was a short learning period of around three weeks before everyone was fully aware of the expectations placed on them regarding pre-reading and the practical work in sessions. One challenge was that this was the only 20 credit module in a 120 credit diet using this approach.

3.7 What the Results Show—the Data

There three main data sources accessed for this analysis:

- Overall module and individual assignment marks—two assignments, equal in weighting and duration plus the resulting averaged module mark;

- Attendance database—each student is required to swipe in to each timetabled session; and
- VLE engagement reports (the click data on items, folders and *Reviewed* buttons).

3.8 Module Mark

For several years, the module has been split in to two separate equally-weighted assignments spread across two terms of 12 weeks each. A term comprises 11 teaching weeks and a centrally placed Guidance Week (similar to a Reading Week). Term 1 dealt with the construction of a content management system (CMS) website to a semi-professional standard including customised layout menu, content, imagery, security, backups etc. Students could do as much technical or creative work as their course and interest dictated. The theme was at the student's discretion but they were encouraged to create something that served a module or their current- or future-self (performer or band website). This work is very chronological in the build-up of necessary skills for good website creation. The weekly learning is quite intensive, with much to learn, it's quite technically orientated and there are opportunities for technical students to engage in learning code.

In the second term students learn about planning for video creation, filming, screencasting and making videos. Again, this serves to aid their future-self, to aid placement job searches in year two or, as many discovered, a chance to have lots of fun. Many of the technical students were encouraged to produce videos along the lines of: 'A day in the life of a music producer' or 'How to engineer a [Hip-Hop] track'. This work was more creative and thoughtful for students than the website and for many, very enjoyable.

Figure 3.1 shows a bar chart of termly assessment and class average marks for the three years up to and including 2014/15. We have recently adopted an institutional target of 75% of students achieving a minimum 60% module marks (the lower boundary of a 2:1 Degree classification) in relation to TEF metrics and a steady growth can be seen over the three academic years from 58, 61 to 65% for the year in question. Additionally, the blue bars represent the technical website-build work (term 1) whilst the green represents term 2 video work. There seems to be a significant increase in term 1 marks from around 57/58 in the preceding years to 68%. This is attributed to the increased exposure to materials for both the technical and creatively/visually-minded students of the class. It is also partly reflected in the 20% of marks awarded for experimentation and pursuing of specialist knowledge within the assignment, as befits their course and interest.

The yearly cohorts are particularly high achieving (evidenced by UCAS points) and passionate group of students, some having the same subject background as our Web and Multimedia students, and can produce significantly better work than the latter group. However, whilst much of the work handed in is of a high quality, there is a persistent yearly element of students with low final marks due to missing elements of

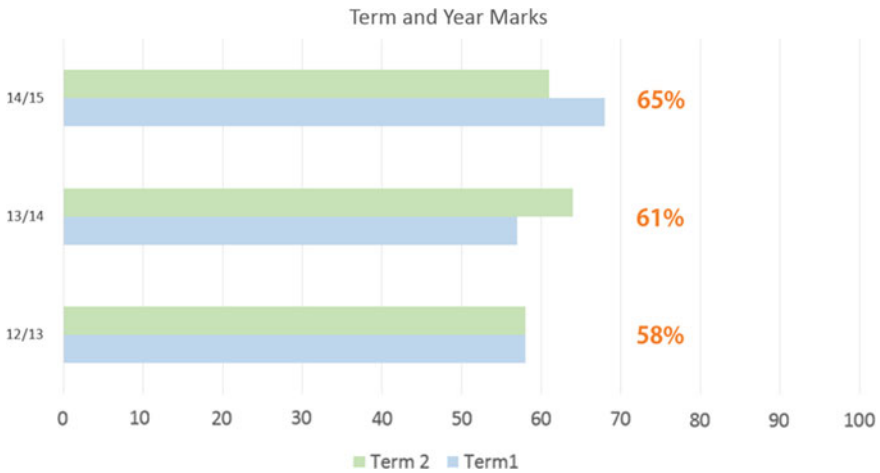


Fig. 3.1 Averages for the module for three years

work; they do great work but don't always complete all the elements. Consequently, excellent students scrape by with averages marks close to the 40% pass.

3.9 Attendance Versus Attainment

As an institution, we have had attendance monitoring in all timetabled sessions for several years and it is used to inform student risk algorithms for *support and guidance* interventions. Two key triggers are maintained: (a) students should ideally have an overall attendance record of 75% and (b) if attendance falls below 50% then the student is requested to attend a formal meeting. In reality, there are escalating interventions as attendance falls towards the lower value (invites to attend with support staff for example).

Attendance was very good in term 1 as seen in Fig. 3.2. This shows Attendance in sessions against Assignment Marks for individual students. The upper horizontal line lies at the desired minimum of 60% marks and the lower line lies at the 40% pass.

Figure 3.2 shows clear grouping in the top right corner with approximately a third of students falling in the 50 to 75% attendance band. Interestingly, there is a 50/50 split in the 50 to 75% attendance band of those above and below 60% marks indicating that attendance may not be that important when students have access to materials remotely, although it is clear that high attendance and high marks go together in the 75–100% attendance band. Here, 70% of students attain 60% marks or more. Additionally, there are 15 students with 100% marks for this component as a result of them doing more than could be expected in the time available. The prescriptive/list nature of all the weekly learning items provide a useful checklist for

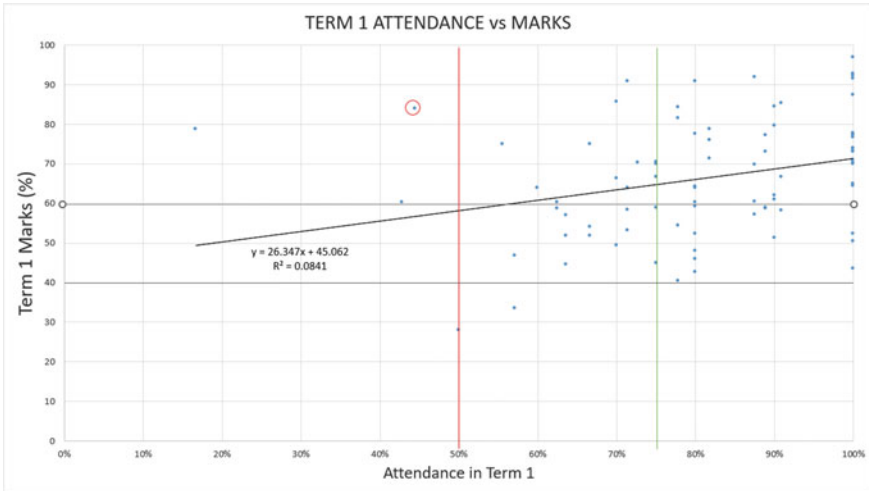


Fig. 3.2 Attendance versus marks for assignment 1

assignment work, especially when students undertake additional experimentation and it is easy for students to ensure they work in ‘one of each’ learning points in to their assignments.

The student result marked with a circle is one who will be referred to several times in this Chapter. This student was a particularly consistent high-achiever over both terms but who was a borderline case for a disciplinary attendance interview with the Dean. They managed their time exceptionally well and had high engagement with the VLE, as we’ll see later.

Term 2 (Fig. 3.3) shows a different picture for the cohort, partly because of: the less intensive learning requirements; a more creative assignment including design and documentation; a need to ‘get out of the building’ and film their footage; sometimes the need to managing actors; and the willingness of some students to ‘play the system’ (students who reduce both attendance and VLE engagement then cram their assignment work in to the last few weeks).

Figure 3.3 shows a broad spread of marks across the attendance range, notably with 10 having marks higher than 60% and attendance worse than 50%. There are 13 students who, having withdrawn or suspended, have zero marks. Interestingly, the 50/50 weighting allows students to have less than a 40% pass on one assignment element if the other mark compensates. The student with the ring is still attaining just enough to get a First having dropped their attendance to 45%. Clearly, the negative slope defies perceived wisdom, mainly because of the high-attending non-submitters.

Figure 3.3 shows a good set of attainment results, mainly due to the creative and enjoyed nature of this assignment, all but two fully-submitting achieving a pass grade or better.

Keeping all the non-submitters and averaging over the whole year, Fig. 3.4 shows a broad scatter around a relatively shallow incline, showing that grades improve with

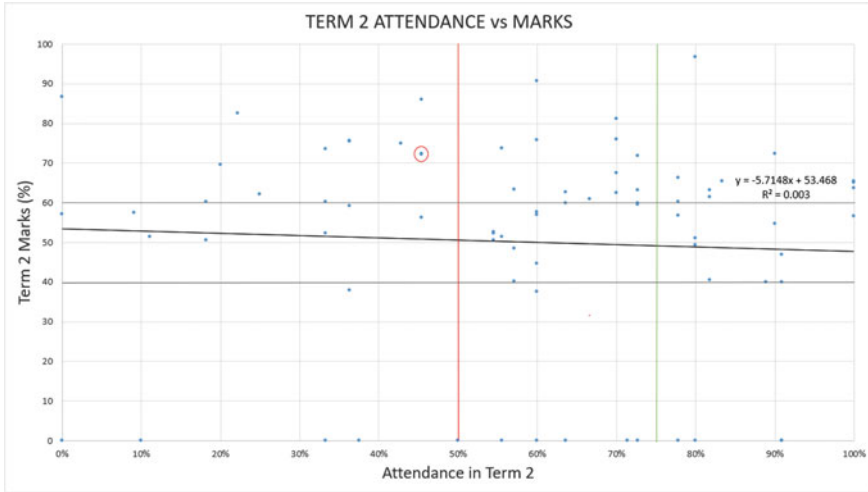


Fig. 3.3 Attendance versus marks for assignment 2

attendance. There are still four students who have good attendance and fail to make a 40% pass (bottom right corner).

The trend line indicates a 44.3% pass without attendance and only 65% with full attendance though the R^2 value is relatively small at 0.0491. The red-circle student continues to provide an exception to this result.

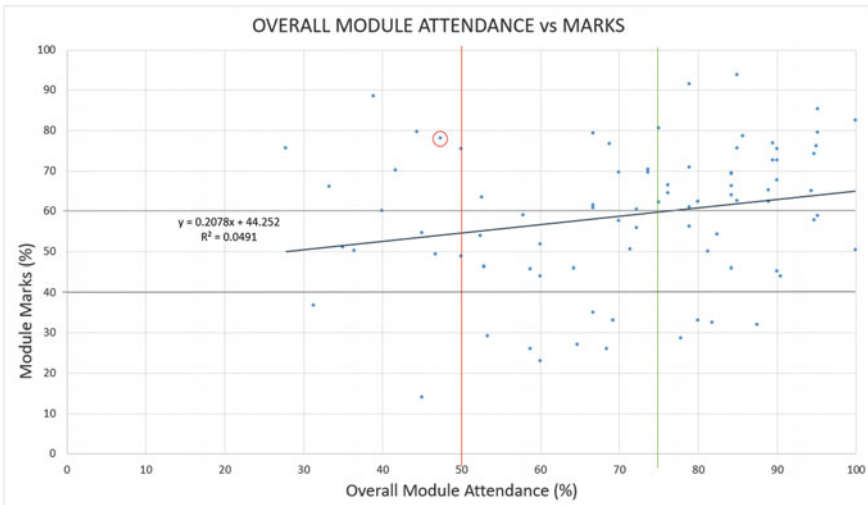


Fig. 3.4 Attendance versus final marks for the module

In conclusion, it is clear that in many cases, good attendance results in good marks, though not always. Weekly engagement with structured learning materials and progressive tutorial content would seem to indicate that good attainment marks result from good attendance and that there are clear exceptions, especially when materials are made available for flexible access and students are good at time management. The term 2 creative work was less successful at engaging students to attend, but again, continued flexible access to resources has resulted in good results in well-motivated students.

3.10 VLE Engagement

Some useful high-level information came out of UniLearn (our branded *Blackboard*TM VLE) in relation to overall student engagement. However, pursuit of this data revealed that the dataset for each module only lasted six months before it was overwritten and was not easily obtainable without firstly knowing that such data would be needed at the beginning of a term (which it wasn't). Thus each term's VLE engagement information came from two data sets; one taken after Christmas (term 1) and one in May (term 2) with much work done accessing rolled-back stored data.

Summary engagement data is available for daily click activity, hourly engagement, the number of *Reviewed* buttons clicked and the overall student engagement (total number of clicks). In all, over 220 elements were embedded in to the VLE module, most of which were set to record data (one or more student clicks).

The hourly and daily summative data for term 1 is shown in Fig. 3.5. Hourly engagement (left) shows peak engagement occurs in class times between 11:00 and 17:00 and that around 8% of the peak (around 500 out of 6000) uses occur in in the evenings in each hour up to about 01:00. The daily engagement (Fig. 3.5, right) shows high use mid-week with around a quarter of peak daily use coming on Fridays when the materials are released for the following week. Weekend use is low but there is around 50% peak use on Mondays when there were no classes. Clearly students are engaging flexibly with the VLE content.

Term 2 data for hourly and daily engagement summaries show a similar profile with much reduced values, with peaks of around 50% of term 1 engagement.

Interestingly, whilst the Thursday engagement reduces from around 6000 to 3500, (class time), the Monday values drop little, from 3200 to 2600, indicating that there is less class engagement (attendance) but students are still using Mondays to work on the module (Fig. 3.6).

The sequential daily engagement in term 1 (Fig. 3.7), where technical content is high, shows a relatively good engagement with a dip to around 45% of peak mid-term and then relatively stable usage at around 60% of peak towards the end-term and assignment hand in. The assignment is handed out in teaching week 2 with 900 click peaks in the weeks either side. There is a non-teaching Guidance Week in the middle of term then a resurgence back up to 600.

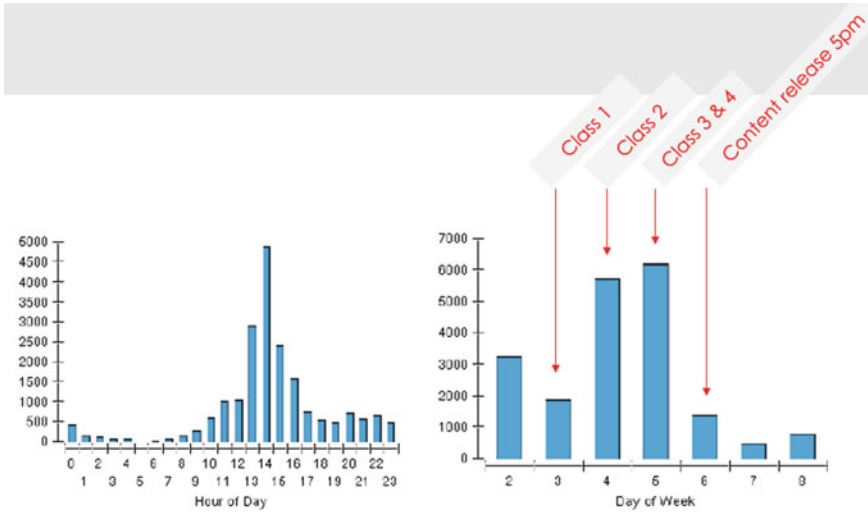


Fig. 3.5 Hourly and daily VLE Engagement during term 1

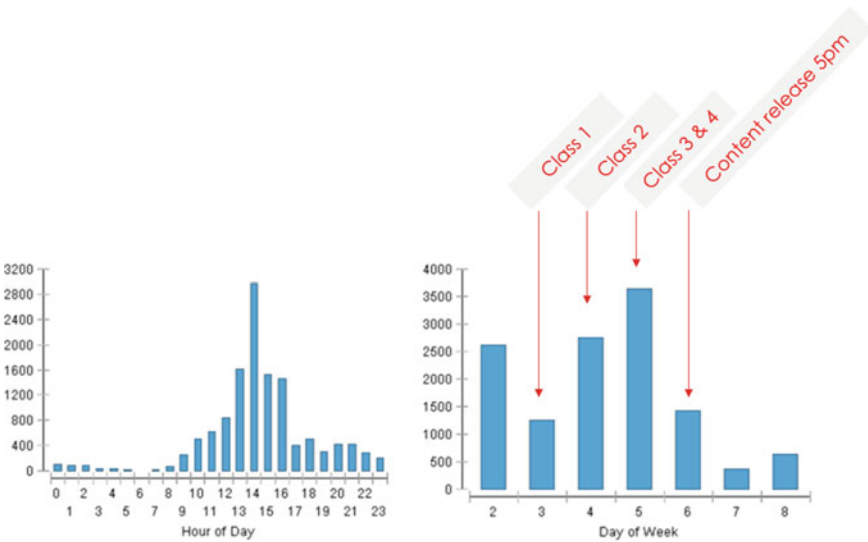


Fig. 3.6 Hourly and daily VLE Engagement during term 2

The Term 2 graph shows a more relaxed attitude to VLE engagement, running in parallel with the attendance (Fig. 3.8); peak engagement in the first four weeks then a rapid fall-off to the three week Easter holidays and a low resurgence in the last two weeks before assignment hand in.

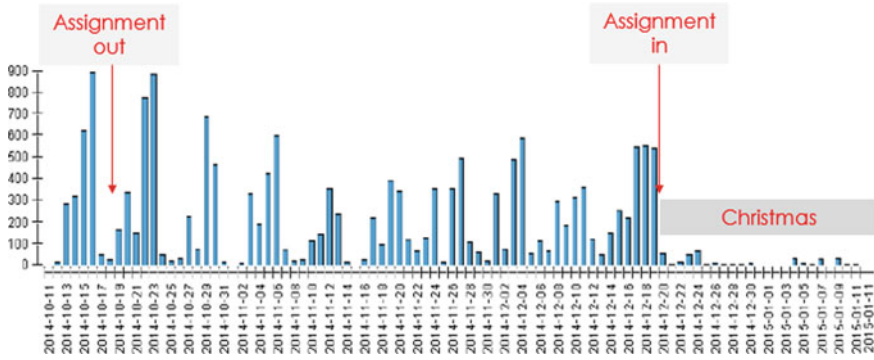


Fig. 3.7 VLE Engagement during term 1

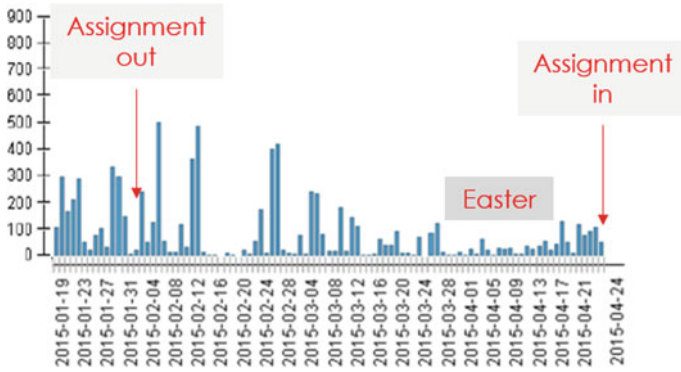


Fig. 3.8 VLE Engagement during term 2

3.11 VLE Engagement Versus Marks Data

Due to the difficulty obtaining data from Blackboard™, the results are quite crude, although quite revealing. The results below look at the resulting total *Reviewed* button clicks vs attainment marks for each student (Fig. 3.9) followed by the total number of engagement clicks vs attainment marks (Fig. 3.10). The differentiation is that the former highlights only the exposure of the Weekly Tutorial Tasks (by clicking the *Reviewed* button) and the latter addresses all clicks within the VLS for each student.

Figure 3.9 clearly shows that whilst there is broad spread across the chart, there is a correlation between the students’ journey in to the weekly tutorial folder and their marks. With 26 *Reviewed* buttons available, our red-circled student (poor attendance, high performance) can be seen as having the highest engagement in this area, seemingly, having looked at all the resources. This tends to support the idea that some students are working strategically through the resources in their own time whilst

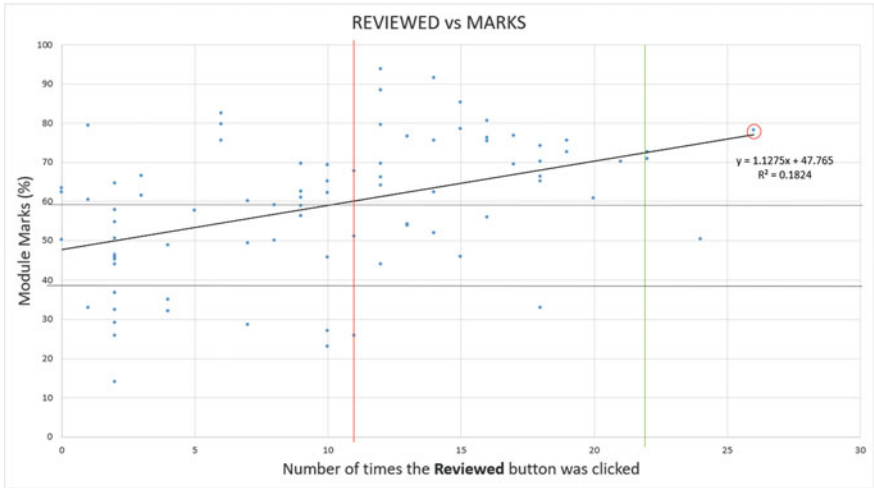


Fig. 3.9 Total Reviewed button clicks versus module marks

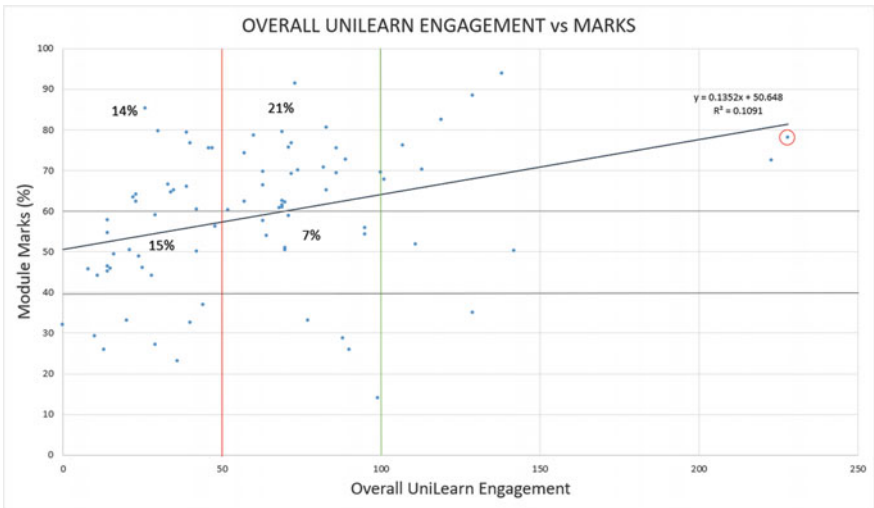


Fig. 3.10 Total engagement clicks versus module marks

minimising travel and contact time in class. This is an increasingly common occurrence with a larger proportion of the Huddersfield intake commuting from home due to spiralling costs.

Since the *Reviewed* button does not indicate whether a student worked through the resulting exposed tutorial materials, it does show that some attained reasonable marks whilst having entered no more than five weekly tutorial areas (seven students with greater than 60% marks and nine with marks between 40 to 59%).

A vertical line in Fig. 3.9 is placed at 22 Reviewed clicks, representing one click for each teaching week and nominally encompassing the bulk of clicks. A second line is placed nominally at a mid-point of 11 clicks. Analysis of the grades to the left of this line show that there are roughly equal numbers achieving 60% or more (18) with those between 40 and 60% (17) and a relatively large proportion not passing (12). Percentage results are 18%, 17% and 12% respectively. However, between 11 and 22 clicks, the resulting values are 27%, 6% and 1% showing a dramatic rise in success of those engaging with the content.

Whilst the R^2 Regression value of 0.1824 is high, the line and trend is persuasive, the intersect still resulting in a mark of almost 48%. The trendline slope is 1.1275 which is approximately five times steeper than that of overall attendance slope of 0.2078 (Fig. 3.4). Hence it can be argued that VLE engagement, particularly into areas that are actively opened by students, is of greater significance to student marks than simply attending classes.

Figure 3.8 shows the total engagement clicks within the VLE module per student against their module mark. Approximately 230 clicks were possible and once again, our red-circled student is leading the way on engagement.

Two vertical lines have been drawn nominally to act as a datum and were chosen partly to reflect the data grouping and be positioned on significant engagement numbers (50 and 100). Without the two data points at around 230 clicks, there seems to be a very broad scatter, which is not helped by the non-passing marks, typically from students who had not submitted assignment 2 (those who had suspended, withdrawn or repeating the module in the following year).

Figure 3.10 also shows that there are roughly equal numbers of students below the 50 click engagement line with marks 40 to 59% (15%) with those attaining 60% and above (14%). Although not investigated, the author's view is that this group included those with previous skills and experience of the learning materials who didn't need to access the tutorial materials.

Interestingly, the increased VLE engagement between 50 and 100 clicks sees three times more students attaining 60% or more to those between 40 and 59% (21% of students compared to 7% respectively). This is a good indicator that those who have a reasonable engagement with the learning materials attain better marks, not least by having a greater knowledge of what the assignment requires of them. Only 10% of the students (nine) engaged between 100 and 150 clicks with all but one passing and just over half (five) achieving marks of 70% or more.

It is from the above that overall engagement with the VLE is a more significant factor in increasing student performance than attendance in class and that flexible access to materials is a predominant factor in increasing student engagement and attainment.

3.12 Benefits of the Flipped Classroom Approach

There were some early gains with this approach, the first being the engagement of students in class. In previous years, the ratio of taught content to tutorial and support work was roughly 2:1 due to the reduced time in class. There was a tendency for students to become distracted and noisy unless firmly controlled and guided. The flipped classroom approach provided a sea-change; the technical and information-rich nature of term 1 materials resulted in students being eager and engaged with their learning and tasks, to a point where there would be utter silence for long periods. Some students would not want to be disturbed by enquiries to see if they needed any help.

The need to be very organised in advance of the day of materials release (typically Friday of the week before) is almost a reason to put this item in the *Issues* section (following). However, it provided excellent time management practice and was a benefit simply because the teaching week generally remained free of problems—all the hard work was completed the previous week.

Students like having a regimented presentation of materials which is in a predictable format and is consistently available. It's one less thing for them to worry over. Also, providing a one-page weekly timetable across all weeks and the consistent numbering, labelling and naming of the weekly learning items allowed quick access backwards at assignment time.

Students appreciated having the current week's material located at the top of the page. Scrolling down the page to find older materials is an intuitive process—recent learning is near the top and content from many weeks ago is further down the page.

Many students commented that they liked having the materials available in advance of the session, particularly over the weekend so they could take advantage of the 1.5 h session to do practical work. It would be advantageous to some of the cohort to have materials available for at least a week prior to the class session.

The Learning Resources area provided access to a wide range of material formats; students enjoyed the mix of web articles, digitally-accessed academic papers, blog posts, podcasts, videos and screencasts. The use of a variety of different *voices* (literally and metaphorically) helps students accept new and challenging theories, concepts and techniques. These resources do need careful choice regarding academic level, the potential for mixed messages and relevance. The change of *presenter voice* is important because different experts can not only tell the same story differently, but they approach the content with different backgrounds and motivations, often introducing associated and interesting asides. These are a good opportunity for increasing the engagement and extending wider reading around the subject. This reflects similar findings in.

The weekly tutorial work has historically comprised pdf instructions with graphics and screengrabs to allow students to build/create digital assets such as elements of a website, manipulate a digital image, to edit elements of a digital video etc. These are typically sequential step-by-step guides. They were updated yearly to reflect software

versions and interface updates, however, with the increased workload arising from the flipped classroom approach, the author decided early on to reference previous year's tutorials as legacy materials, with additional time taken in class to point out the relevance and changes exposed in the older material.

3.13 Issues with Flipped Classroom Approach

There is a need to upload a lot of relevant, appropriate and variable content into the weekly areas in the VLE in advance. It's a lot of work! This can seem relentless at first until the methodology of finding and presenting resources becomes familiar. There is a need to formalise the content structure and to make weekly work tasks *approachable* before the session and also to provide additional challenge within (and beyond) the session. The weekly content needs to be valuable to the students rather than a hasty collection of web links and a mix of format types helps break up the learning process.

Use of Adaptive Release for the weekly tutorial tasks (use of the *Reviewed* button) was not as important a facility with students as it might be with others (say staff on an internal training course). Students were not informed that it would be possible to analyse data on whether they click each *Reviewed* button or not. Telling them this information can often work as an implied threat and can be useful in driving up usage and exposure to the materials. However, students can *play the game* and scroll down the learning resources, click the *Reviewed* button and click on the exposed tutorial tasks simply for the purposes of achieving a better engagement score.

Whilst the benefits of using the flipped classroom are clear, the connection between engagement with the materials and summative marks is also clear, so anything that can be done to drive students in to the content is well worth the effort. Ideally, there needs to be some incentive for students to investigate all the materials; such as a weekly class discussion item, contribution to a Wiki or Discussion Board based on *deeper* materials, creating a logbook that forms part of the assessment strategy etc. This additional work was not done due to workload.

Students have no way of being reminded of what they have read or what sections they have worked through or simply visited. Data access via our version of Blackboard provided very rudimentary data that was difficult to use. Other VLEs may treat these issue better.

The system of presenting learning materials, work tasks and tutorial solutions on a VLE could easily be circumvented by students. Good students may do so to accelerate their learning whilst poorly performing students may dip in to see if it is of interest and to see the complexity of materials presented, or not dip in at all. There ideally needs to be a mechanism that encourages or requires students to open and engage with as much of the content as possible; this was a failing in this work and the obvious next-step.

The basic approach of expecting students to engage in pre-reading materials before class provided an additional challenge, which soon became apparent in the first weeks of this methodology. It was the only module (20 credits in 120 for the year) that required students to undertake pre-reading before the class. For many, this was a culture-shock and presented them with time management problems which could be easily avoided if desired. Pre-reading became a polarising activity with some relishing the access to deep and rich materials and others learning the predictability of the system to do the minimum required to attain their desired grade. The flipped classroom approach would be more successful if a greater number of modules were taught this way.

3.14 Personal Reflections

Moving from weekly PowerPoint presentations and tutorial sheets to the flipped classroom was a big workload commitment because of the many new resources needing to be found, evaluated and presented for student dissemination. Existing teaching materials became the focus around the generation of new weekly content but soon became relegated in importance for both myself and students, mainly because they aren't presented in class time. Students may or may not want to look through a series of slides, and it may be possible to collect that snippet of data if the VLE allows. However, one of my colleagues maintains that his students would prefer to watch a video of his presentation rather than watch him present live, so it would be worth testing this hypothesis by converting some slideshows to screencasts and embedding on a content page, then testing.

The process of finding interesting and relevant materials never stopped being fun; mainly because of the updating of my own personal knowledge and also because of the interest and difference it made to the students' learning experience. One of the challenges with working in the fast-moving technology area is the need to update materials which can quickly become outdated and this is something that needs careful attention each year. It may be useful to have a Current Trends section which resides above the weekly content and has current-interest links and information. Putting new content at the top of this folder provides an archive of previous years' content if it is sectionalised by academic years, making a virtue from additional updating.

All of the weekly preparation work was completed in the week before it was taught, so contact time was far more relaxed for both myself and students in that week. This allowed more time in class for signposting towards upcoming content and assignment work as well as reference previous weeks' content as questions arose. There was also more time to spend dealing with student support as well as suggesting and demonstrating improvements they may consider making towards assignment work. And finally (significantly), interactions were more fun and creative in class and it became easier and quicker to remember student names.

Additional evidence regarding module evaluation scores and comments, use of a data-aware reading list, additional survey information and the use of screencasting to

provide assignment feedback and marks justification are not covered in this Chapter but provide additional support of the above findings. Whilst there were significant improvements in student learning in this approach, it is anticipated that additional improvements could be gained by including more-engaging and interactive activities in to the weekly tutorial sessions creating a better blended learning experience (gamification, use of Wikis, Discussion Boards, polls etc.).

3.15 Conclusions

Whilst only providing a vehicle for presenting learning materials and weekly tutorial sheets, this flipped classroom approach has proved beneficial in: motivating students to engage with the materials; being studious and focussed in lab sessions; and raising metrics such as module marks. The approach in providing such materials was consistent over both terms, the first (technical) half produced high levels of attendance and VLE engagement whilst the second half (creative) suffered from reduced attendance and VLE engagement. Average marks in term 1 increased by about 10% whilst term 2 marks reduced by approximately 3%. It is concluded that creative content needs a greater level of activity planned in the weekly sessions to drive attendance and learning.

Students benefit from having a logical weekly presentation of materials in their VLE. Using a combination of teaching-week numbers and appropriate titles allowed students easy searchable access to a large bank of materials at any time in the year.

Around a quarter of the cohort accessed the weekly materials in the preceding week and many spread their engagement across the weekend and into the early hours of mornings.

VLE engagement seems to be a better indicator of student performance than attendance in class for this cohort. If rules and penalties allow students not to attend sessions, then some will take advantage and manage their time accordingly and others will decide not to engage with the learning.

The need to actively open the weekly tutorial materials by clicking the *Reviewed* button is easily done without having to undertake the learning. The fact that many did not take this shortcut indicated a distinct lack of engagement by some.

The provision of learning materials provides a rich tapestry to all types of learners with excellent opportunities for asides and extra reading for those interested in the topic. For staff, a great deal of learning and sifting, especially of the myriad web-based opinions, can provide an enjoyable and nourishing activity, not least to keep abreast of current thinking.

References

- Critz CM, Knight D (2013) Using the flipped classroom in graduate nursing education. *Nurse Educ* 38(5):210–213. <https://doi.org/10.1097/NNE.0b013e3182a0e56a>
- Galway LP, Corbett KK, Takaro TK, Tairyan K, Frank E (2014) A novel integration of online and flipped classroom instructional models in public health higher education. *BMC Med Educ* 14(1)
- Gilboy MB, Heinerichs S, Pazzaglia G (2015) Enhancing student engagement using the flipped classroom. *J Nutr Educ Behav* 47(1):109–114, Jan–Feb 2015
- King A (1993) From sage on the stage to guide on the side. *Coll Teach* 41(1993):30–35
- Missildine K, Fountain R, Summers L, Gosselin K (2013) 2013 Flipping the classroom to improve student performance and satisfaction. *J Nurs Educ* 52(10):597–599
- Sergis S, Sampson DG, Pelliccione L (2018) Investigating the impact of Flipped Classroom on students' learning experiences: a self-determination theory approach. *Comput Hum Behav* 78

Chapter 4

Applying Cognitive Theory to the Teaching of Programming: Metaphors, Robots and Problem-Based Learning



Clive Rosen

Abstract The age old problem of how to teach programming has gone around in circles for decades. What programming language should we use? Should we introduce object orientation from the outset? Is one approach better than another and why do some students have almost no difficulty learning to program whilst others never seem to get it? Within this wealth of discussion papers however, there is a paucity of theoretical argument upon which to base the competing suggestions. This is problematic because it impedes discussion as to why a particular approach should be better than another. This chapter suggests that cognitive theory, in particular Construct theory (Kelly in *The Psychology of Personal Constructs*, Routledge, London, 1991) offers a potential explanation as to why some students find learning to program easy and others struggle. This chapter suggests that the essential use of metaphors in teaching resonate with the cognitive model of some students but not with others and that cognitive noise plays a role in the learning process. Finally, it proposes that using robots can help reduce the cognitive noise, mitigate the cognitive gap and keep all students satisfied whatever their ability.

Keywords Teaching programming · Cognitive · Robots · Problem-based-learning

4.1 Introduction

Before I begin, let's be clear, not all twenty-first century students/pupils need to learn how to program in order to make good use of information technology.¹ After all, you don't need to understand the workings of an internal combustion engine or the physics of electric induction to drive a car. It is important to say this because some people do not have the mind set to learn to program and it is unrealistic to

C. Rosen (✉)

Passerelle Systems Ltd., 11 Edward Avenue, Newcastle Under Lyme ST5 2HB, Staffs, UK

¹ Whether there are other good reasons for learning to program as Soloman Solomon, C. J. and S. Papert (1976). A Case Study of a Young Child Doing Turtle Graphics in LOGO. Proceedings of the June 7–10, 1976, National Computer Conference and Exposition, Association for Computing Machinery: 1049–1056, numpages = 1048. and Papert argue, is open to debate.

expect them to. Others, on the other hand, take to programming like ducks to water. Every lecturer of introductory programming has experienced this phenomenon. Some students seem capable of writing a new operating system after 12 weeks of an introductory programming course without assistance from the instructor, whilst others fail to grasp the concept of a variable. Why is this this, and what, if anything can be done about it?

This is not a new question. It has perplexed teachers and tutors for a long time (Sheil 1981). A 2003 review paper (Robins et al. 2003) identified 91 publications on the subject of teaching computing. This exercise repeated 14 years later (Luxton-Reilly, Albluwi et al. 2018) found a further 2189 papers on the subject. Excluded from this study were more specific reviews of student misconceptions (Qian and Lehman 2017), teaching approaches (Vihavainen et al. 2014) and programming comprehensions (Schröter et al. 2017). These three studies found 133 papers, 226 papers (of which only 32 were studied in detail) and 540 papers (111 of which were excluded from the study).² Clearly this is a subject of considerable debate. However as Luxton-Reilly, Albluwi et al. (op. cit) point out, within this body of literature there is a paucity of general theory that could help to explain the difficulties students suffer in learning to program. This chapter is intended to address this deficiency and offer a theoretical perspective based on cognitive construct theory (Kelly 1991). As with all theory, construct theory offers potential insight into the problem but must be interpreted in the light of the specific situation (in this case learning to program). Hopefully this approach can help to explain why large numbers of students have difficulty learning to program and where, as educators, we might make some impact.

4.2 A Cognitive Perspective

Four barriers to learning to program can be identified from a cognitive perspective, but it is helpful to distinguish fundamental cognition problems from the cognitive noise that compounds the learning difficulties. Although cognitive noise is important, it can be treated as a second order issue. In the case of programming there are (at least) two fundamental problems and two issues that could be regarded as noise. The two fundamental problems are:

1. The abstract nature of programming concepts that divides the programming world from previously understood concepts in the physical world. This results in students failing to grasp a sufficient understanding of the tools available to them.
2. The lack of a schema that students can adapt and apply to solving problems using programming constructs. So even when students understand the tools they have, they don't know how to use them.

² There may well be some papers that appear in more than one of these sets.

The two cognitive noise issues are:

1. The need to learn how to use whatever development environment that is being used.
2. The pedantic nature of language syntax and the peccadilloes of particular compilers.

Cognitive noise creates confusion in the minds of student and compounds their difficulties, but do not constitute fundamental thinking misconceptions. If they can be dealt with separately (offline so to speak) as a distinct and separate task, the more visceral difficulties can more easily be addressed. The noise issues act as distractions, create obstacles and frustrations in the minds of students that constitute the straws breaking the camel's back. The weight of these straws can be minimised by choosing simple development environments, unfussy compilers that provide meaningful error messages and clearly distinguishing between runtime and compile time. However, the essential cognitive problems remain and it is these two difficulties that are addressed in the rest of this chapter.

Construct theory suggests that people build a cognitive model of the world from infancy by contrasting new concepts, objects and experiences with previously existing ones thus growing a more and more complex, sophisticated and unique understanding of how the world works. It is making connections between these "constructs" that enable people to learn and adapt. Existing constructs are continuously challenged by new observations. Existing constructs are used to predict what might happen in any given situation. If the prediction is determined to be lacking in some way the construct model is modified in the light of the new experience.³

The problem for programming students is that there are no existing constructs to relate to when learning to program. This is because programming constructs are abstract (mathematical) concepts and artificial in the sense that they do not relate to real world constructs. Whereas maths is taught from infancy and, initially at least, can be related to physical objects, programming is not universally required and has no foundation in the real world. Students approaching programming for the first time therefore lack the necessary cognitive models required for competency (Winslow 1996).

For some, the lack of pre-existing constructs does not present a great problem. They find connections that make sense to them and are able to build a revised working model of the world that include the new concepts without much difficulty. These are the people who find programming easy. Often, it is these people that go on to teach programming which is a problem because they do not understand why other

³ This process can involve a level of cognitive angst as the person's model of the world needs to be remoulded. As a result, there is a tendency to evade the mental pain implicit in reshaping the cognitive model as people become more wedded to their existing model. Inflexible cognitive models are sustained through a number of methods, the most common of which is rationalising the experience out of existence (it was something else that happened), denying that the observation happened in the first place or simply walking away from the experience and saying "I'm not interested in it". This process is the subject matter of human psychology and far too large and controversial to enter into here.

people cannot see the seemingly obvious. They may therefore lack empathy with those that cannot make the connections. This can lead to staff working more with the successful students and paying less attention to those who are struggling. It is however the struggling students, the ones who don't make the connections, who need help to progress.

Metaphors can be used to try and bridge the gap between concept and construct, but in programming, the metaphors break down quite quickly and fail to offer a sufficiently comprehensive understanding of the constructs (Sanford et al. 2014). So when a student learner tries to apply a programming construct and it does not behave in the way they had theorised that it should, it leaves them with an incorrect cognitive model and no way to fix it without help. Often the instructor is at a loss to understand what it is that the student doesn't understand. The student can't explain what they don't understand because their model of how they thought something should work has just been blown apart. Reusing the same metaphors the tutor has previously used does not help because the student's interpretation of that metaphor is either incorrect, or the point at which the metaphor breaks down has been passed. So, whilst for some the metaphor makes perfect sense, for others it just exacerbates their confusion. For example, many teachers use the metaphor of a box to explain a variable. This may work for some because they appreciate that although boxes do not empty themselves, the concept relates to a storage container for a value, not a thing. Others have no idea what is going in the box or how the content of the box can be changed (Kohn 2017). Sometimes the metaphor simply doesn't relate to the student. My own experience of understanding variables was eased by the metaphor of music cassettes. I was used to recording, and over-recording LPs onto magnetic tape, and I recognised that this was exactly the same process for variables, only using numbers rather than music. This metaphor would be pretty much meaningless to today's students who probably have never seen a music cassette in the age of downloads and smart speakers. Perhaps a closer match could be made with creating and erasing contact list entries? The point is that the student needs to both be able to relate to the metaphor and understand the aspects of the metaphor that relate to the new domain and which do not.

Two compounding complications can exacerbate the situation for the confused student at this point. The first occurs when the lecturer marches on assuming that their preferred metaphor has explained the concept. If the student has failed to identify with this particular metaphor they are simply left behind. The second issue is that in computing, one concept builds on the understanding of previous concepts. You cannot understand arrays, repetition or parameter passing unless you understand how a variable works. Once left behind it is a long way back to fix the problem. Try solving a Sudoku puzzle once you've made an earlier mistake (to employ a metaphor which would be meaningless to anyone who doesn't solve Sudoku puzzles.) Instructors who never had a problem understanding a concept find it difficult to understand why a student cannot grasp what they think is a trivial idea. Time pressure to get through the course material and lack of empathy for the struggling student drives the lecturer to plough on regardless piling one abstract concept on top of another. The cognitive load on the student becomes unbearable. At the same time, their resilience is undermined by the whiz kid students who are ready to rewrite the internet and are

becoming impatient at the laggardly pace of the course. The lecturer's dilemma is how to reconcile these two groups. One possible solution is discussed later.

As suggested above, students face a second order difficulty; that of how to use the tools available to them to solve a programming problem. This arises from the apparent disconnect between the problem and the tools. It's hard to plant a garden if all you have is a hammer. Again, construct theory offers a possible explanation. As people build up their model of the world, they also develop the necessary social tools to navigate their way through it and how objects can support them like a child playing with a cell phone. As both computing problems and the tools available are new, student programmers lack the knowledge and experience of how to construct a plan to solve the problem. Using example programs works for some students because they can make the connection between how the example program works and the task requirements. Others will amend the example almost by rote without understanding the principles hoping that by doing so enlightenment will emerge. If it does not, this group of students will be unable to generalise from the example to implement solutions to a different problem. They have failed to make the cognitive connection between the tools and the problem because they don't fully understand how the tools work. No number of example programs is likely to close this gap. A common teaching approach is problem decomposition. Clearly this can be helpful, but is limited if the student is still struggling to understand the tool itself. A more productive approach is to apply the tools in ways in which the student is more familiar so they can build a more secure cognitive model of how the tools work. This was the theory behind Papert (1993) and Solomon (Solomon and Papert 1976; Soloway 1986) proposal to use Turtle graphics and Logo as long ago as the 1980s. Robots are a similar, but updated approach. Whereas Turtle provided a visual analogue to promote comprehension, robots offer both a visual and a physical artefact which reduces the cognitive gap between the purely virtual world of the program and the real, physical world.

Before exploring alternative approaches however, it is necessary to return briefly to the question of cognitive noise. The concept of the ratio of signal to noise has long been recognised in cognitive science. This concept suggests that extraneous events, distractions or thoughts can interfere with the transmission and or the reception of the signal (the main message.)

Noise involves irregularities and failures of correspondence between the sensory signal and target properties in the world. Noise arises from at least two sources. Externally, it arises from poor environmental conditions, as when you're viewing objects in fog. Internally, noise arises from variability in neural activation, as when neurons fire differently on two occasions, despite the same relevant initial conditions. (Vance and Stokes 2017)

Here we need only be concerned with external sources of noise. The objective of an introductory programming course is to learn to program. It is not to learn a development environment. This may be a necessary step on the way, but it is not the primary objective. Similarly, it may be necessary to learn the programming language syntax and how to interpret the sometimes obscure error messages the compiler generates, but again these are a means to an end. Having to gain this knowledge is

therefore a distraction that distorts the comprehension of the main learning objective. A good example is the difference between “ = ” and “ = = ” in some programming languages, an easy mistake for anyone to make, but one that can result in total incomprehension for new programmers and one that can be difficult to correct.

The simpler the development environment, the less noise it generates, the sooner it is possible to get onto the fundamental principles of programming. Similarly, the obscurities introduced by various programming languages and their compilers seem more often designed to confuse students. We should not burden students with the need to differentiate $X = Y$ and $X = = Y$, or add unnecessary requirements for semicolons or matching brackets. These things just add to the cognitive load and should be minimised as much as possible. Development environments and programming languages should be chosen with this in mind. Experienced programmers may benefit from little short cuts and big ideas such as inheritance. For new programmers they add to the signal noise. Reducing the cognitive noise is important as it will reduce the level of cognitive distraction, but it will not address the two fundamental cognitive difficulties students face, understanding the tools and how to deploy them.

In the Luxton-Reilly et al. (op cit) review, the authors categorised 17 papers addressing theory whilst 196 papers explored topics such as literacy, behaviour and ability. This appears to be putting the cart before the horse. Developing a robust theory of a subject provides a benchmark against which empirical evidence can be tested. Without theory or with poorly articulated theory it is difficult to evaluate one methodology against another; it is difficult to distinguish significant factors from insignificant ones; and it is difficult to transpose between the general and the specific and from the specific back to a more robust generalisation. Enhancing literacy, interpreting behaviour and analysing ability may offer insights into programming competence, but without a theory as to why they might signal something of relevance, they cannot point the way towards meaningful intervention. Code literacy or debugging skills might be of little relevance to a student’s capability to write a program (Pears et al. 2005; Malmi et al. 2014). Generating robust theory must be a first step towards effective teaching. Cognitive theory seems to provide a strong candidate for this role.

4.3 A Potential Way Forward

Identifying the problem is a start as it helps to focus on the important factors. If, as I have argued above the real problems are a) a disconnect between the student’s existing mental model and how to associate a model that encompasses programming concepts and b) lacking a schema for using the tools to solve a problem, then we need mechanisms that address these difficulties.

I have, in this chapter, used the concept of programming tools. This is, of course a metaphor, but a metaphor with which students are familiar. It is a metaphor that can be readily extended to simplify a model of programming to which students can relate (see below). As stated above, metaphors are a useful mechanism for reducing the cognitive gap between “a source domain and a target domain” (Sanford et al. 2014).

We do however need to tread carefully because as Sanford et al. note, metaphors only partially explain the target domain concept and require interpretation, so care must be taken to ensure the student has composed a sound mental model from the metaphor.

The more the gap between the target domain and the source domain can be minimised, the more likely the gap can be closed. This was the essential idea behind Logo (Abelson et al. 1974) and Turtle Graphics (Papert 1993). However the availability of relatively cheap robots whether Raspberry Pi, Arduino or Lego Mindstorms[®] provides greater opportunities and greater flexibility. Robots provide five distinct benefits.

1. They represent a physical analogue to the normally virtual existence of a computer program so that when an instruction is passed to a robot, a student can see the resulting action. (This offers a small improvement over Turtle Graphics as although both the Turtle engine and the robot are visual, robots are also physical. This reduces the cognitive gap between the abstract and the physical a little bit more.)
2. They provide a ready metaphor for program sequence as it is clear that robots obey one instruction at a time. (Other metaphors can be used if needed to reinforce the concept of a program as a sequence of statements, for example building flat pack furniture from instructions or following a recipe.)
3. Using robots differentiate compilation from run time thus reducing the cognitive noise of compilers. Students can readily recognise that the (pseudo) English of the instruction sequence needs to be translated into a language that the robot understands, and the concept of downloading information into a remote device is now commonplace so does not present any cognitive hurdle.
4. Using robots disassociates the program design phase from the run phase enabling students to concentrate on the programming concepts such as sequence, variable use, conditionals and repetition.
5. Robots provide the opportunity, when combined with problem-based learning, of satisfying the curiosity of capable students as well as meeting the needs of those students who are struggling. The tasks set can be tailored to match the progress the student has achieved. Advanced students can be challenged to use different sensors and be set complex tasks whereas those student who are finding a particular concept difficult can be given tasks that emphasis that particular concept.

Yousoof et al. (2006) suggest that cognitive load can be reduced by code walk-throughs showing what is happening to register and variable values as each instruction is obeyed. Robots add value to this process because they can be seen to be acting on the instructions. The complexity of the task can be increased incrementally. So, for example, a robot could be made to walk forward 10 paces. The next step would be perhaps to walk forward, turn round and walk back 10 paces—1. This might be followed by adding a fixed repetition and then perhaps a conditional repetition using a proximity sensor and so on. At each step the student can see the change to the code and the resulting effect.

Fig. 4.1 Tools and materials used in programming

Tools

Conditionals / Selection ~ If / Then / Else

Repetition ~ Do, While, For

Materials

Data ~ Constants, Variables

Operators ~ + - * / =

It is essential to keep the coding constructs simple allowing the student to progress at their own pace. There is a tendency to overcomplicate programming constructs too quickly. This can give students the impression that programming is hard, planting doubt in their mind regarding their capability to master coding. I would argue that the difficulty of programming is that there are so few fundamental programming concepts that students have a tendency to overcomplicate things in their mind. The set of concepts required are only sequence, a couple of tools and a couple of types of material. The tools and materials are identified in Fig. 4.1:

Robots deal naturally with sequence and little else needs to be said regarding this, but what constitutes “an instruction” needs to be dealt with is a little more detail. “An instruction is either a “data assignment” or a “program control statement”. “Assignments” can then be identified using the materials from Fig. 4.1 and “program control statements” using the tools in that figure. At this point the concept of variables needs to be introduced and the two tools, “Repetition” and “Selection” explained. Once this has been done, students have all the information they need to write a program and solve a problem. Traditionally, the teaching of programming has tended to present each of these concepts separately ad seriatim. It is suggested here that students are provided with the whole toolkit at once so that they can see the whole picture. The subtle differences between tools that do the same job, but differently are treated as noise that can be addressed later.

Treating programming concepts as either tools or materials is clearly a metaphorical allusion to building physical entities in order to relate these concepts to tangible objects. Karlsen et al. (2022) identify a number of metaphors in common use. They also report that teaching staff believe that students find using more than one metaphor for a concept confusing. However, if a student fails to connect with the metaphor being used, they may well find that a different metaphor might make more sense to them. However, all metaphors should come with a health warning.

4.4 Problem-Based Learning (PBL)

Teaching of programming has always taken a problem based approach as programming is a problem solving activity. Often the problem solution has taken precedence over the real objective of ensuring that the basic programming principles are well understood and cognitively grounded by the student. Barrows and Tamblyn (1980) define PBL as a pedagogic system that.

use of realistic problems as the starting point of self-directed, small-group-based learning guided by a tutor who acts as a process guide rather than a point of knowledge transfer (ibid.).

There are several models of PBL (Servant-Miklos, Norman et al. 2019) but the basic principle is that the instructor sets tasks for the students and supports them undertake an investigative approaches to achieving the task rather than didactically imparting information. There are a wide range of both proponents and detractors of this method (Gonzalez 2019), but where resources allow, PBL does appear to well suit the teaching of introductory programming and, in particular the deployment of robots. Once a student has solved a problem for themselves, it is more likely that they will have a firmer grip on the cognitive process they have worked through. Rather than simply going through the motions, developing a flawed cognitive model of the process by rote learning, they will be able to apply the principles and apply them to similar problems. Initially this may be a slow process and require a great deal of support from the tutor, but as the student builds a more stable model, they are likely to grow in confidence and self-belief rather than have this confidence undermined. Students reported their preference for PBL over traditional lectures (Caceffo et al. 2018) but the same study also reported that it took longer for staff to prepare material. In a study by Chang et al. (2022), improved achievement of learning outcomes was reported particularly among weaker performing students. However Souza, Bittencourt et al. (2019) demonstrates the importance of setting the problem at the right level for students to achieve success.

The jury is still out with regard to problem-based learning. The problem is that there are just too many ways to measure success and too many ways in which PBL can be implemented. However the combination of robotics and PBL in the teaching of introductory programming does seem to provide a compatible match when the skill and sensitivity of the tutor to her/his/their tutees' ability and confidence are also compatible. Verifiable empirical evidence is lacking, but given that current, traditional methods of teaching programming do not seem to be altogether successful and that programming robots in a problem-based learning environment is at least fun, it cannot be such a bad thing to give it a go.

4.5 Conclusion

In this chapter I have argued that construct theory provides a coherent theoretical platform for explaining the difficulties some students have in grasping the principles of computer programming whilst others race ahead with very little difficulty. I have suggested that a distinction needs to be made between fundamental cognitive constructs and cognitive noise that distorts the primary cognitive signals and that therefore whatever can be done to reduce cognitive noise and boost the signal should be done. Metaphors can both boost signal and interfere with it and should therefore be used carefully, checking that students have taken the intended meaning from the metaphor and not misinterpreted it. Simple robots can, in effect, act as a physical analogue of program sequence and also as a resource for demonstrating programming concepts. They have the additional benefit of reducing signal noise by providing a physical disconnect between compilation and run time. Furthermore, when combined with a problem-based learning approach they can, if not solve the problem of diverse student capability within a class, at least help to alleviate it.

Teaching introductory programming will always be challenging, but having a theoretical understanding of the problem is essential if we are going to help students overcome the barriers they face and reduce dropout rates from courses. In this chapter I have offered one incomplete and sketchy theoretical model that has facilitated my teaching. Resourcing can be a problem and transitioning from the traditional lecture based course to a lab based PBL approach raises logistical issues, but seeing students get robots running / rolling round the room certainly has its rewards, and, in my experience, students benefit from the approach.

References

- Abelson H, Lee R, Goodman N (1974) LOGO Manual, M.I.T.
- Barrows HS, Tamblyn RM (1980) Problem-based learning: an approach to medical education. Springer Pub Co, New York
- Caceffo R, Gama G, Azevedo R (2018) Exploring active learning approaches to computer science classes. In: 49th ACM SIGCSE technical symposium on computer science education (SIGCSE), Baltimore, MD
- Chang YH, Yan YC, Lu YT (2022) Effects of combining different collaborative learning strategies with problem-based learning in a flipped classroom on program language learning. *Sustainability* **14**(9)
- Gonzalez L (2019) The problem-based learning model. In: Wang S, Lin L, Hartsell T, Zhan H, Beedle J (eds) 2019 eighth international conference on educational innovation through technology (Eitt), pp 180–183
- Karlsen A-CB, Akerbæk TM, Stigberg SK (2022) Towards a metaphor-based tangible toolkit for learning programming concepts. Springer International Publishing, Cham
- Kelly GA (1991) The psychology of personal constructs. Routledge, London
- Kohn T (2017) Variable evaluation: an exploration of novice programmers' understanding and common misconceptions. In: Proceedings of the 2017 ACM SIGCSE technical symposium on

- computer science education, association for computing machinery, pp 345–350, numpages = 346
- Luxton-Reilly AS, Albluwi I, Becker BA, Giannakos M, Kumar AN, Ott L, James P, Scott MJ, Sheard J, Szabo C (2018) Introductory programming: a systematic literature review. In: 23rd annual ACM conference on innovation and technology in computer science education (ITiCSE 2018 companion). New York USA, ACM, pp 55–106
- Malmi L, Sheard J, Simon RB, Helminen J, Kinnunen PI, Korhonen A, Myller N, Sorva J, Taherkhani A (2014) Theoretical underpinnings of computing education research: what is the evidence? In: Proceedings of the tenth annual conference on international computing education research, association for computing machinery, pp 27–34, numpages = 28
- Papert S (1993) *Mindstorms: children, computers, and powerful ideas*. Basic Books, New York
- Pears A, Seidman S, Eney C, Kinnunen PI, Malmi L (2005) Constructing a core literature for computing education research. *SIGCSE Bull* 37(4):152–161, numpages = 110
- Qian Y, Lehman J (2017) Student’ misconceptions and other difficulties in introductory programming: a literature review. *ACM Trans Comput Educ* 18(1):1–24
- Robins A, Rountree J, Rountree N (2003) Learning and teaching programming: a review and discussion. *Comput Sci Educ* 13(2):137–172
- Sanford JP, Tietz A, Farooq S, Guyer S, Shapiro R. B (2014) Metaphors we teach by. In: Proceedings of the 45th ACM technical symposium on computer science education, association for computing machinery, pp 585–590, numpages = 586
- Schröter I, Krüger J, Siegmund J, Leich T (2017) Comprehending studies on program comprehension. *IEEE*, pp 308–311
- Servant-Miklos VFC, Norman GR, Schmidt HG (2019) *A short intellectual history of problem-based learning*. Wiley
- Sheil B (1981) The psychological study of programming. *ACM Comput Surv* 13(1):101–120
- Solomon CJ, Papert S (1976) A case study of a young child doing turtle graphics in LOGO. In: Proceedings of the June 7–10, 1976, national computer conference and exposition, association for computing machinery, pp 1049–1056, numpages = 1048
- Soloway E (1986) Learning to program = learning to construct mechanisms and explanations. *Commun ACM* 29(9):850–858, numpages = 859
- Souza SM, Bittencourt RA, (2019) Motivation and engagement with PBL in an introductory programming course. In: 49th IEEE frontiers in education conference (FIE). IEEE, Univ Cincinnati, Cincinnati, OH
- Vance J, Stokes D (2017) Noise, uncertainty, and interest: predictive coding and cognitive penetration. *Conscious Cogn* 47(S1):86–98
- Vihavainen A, Airaksinen J, Watson C (2014) A systematic review of approaches for teaching introductory programming and their influence on success. *ACM*:19–26
- Winslow LE (1996) Programming pedagogy—a psychological overview. *SIGCSE Bulletin* 28(3):17–22
- Yousoof M, Sapiyan M, Kamaluddin K (2006) Reducing cognitive load in learning computer programming. In: Proceedings of world academy of science, engineering and technology. Vienna, Austria, world academy of science, engineering and technology. 12:259–262

Chapter 5

Distance Learning: Lessons Learned from a UK Masters Programme



Jenny Carter and Francisco Chiclana

Abstract The MSc Intelligent Systems (IS) and the MSc Intelligent Systems and Robotics (ISR) at De Montfort University are Masters courses that are delivered both on-site and by distance learning. The courses have been running successfully on-site for over 10 years. Designing and delivering courses as distance learning presents a challenge, particularly where the content includes technical and practical elements. For this work we look back at some of the techniques that have been adopted and consider their success or otherwise at enabling us to overcome these challenges. We reflect on some previous studies that have been undertaken over the years of running the courses. Finally, the lessons learned from these successful programmes are considered with a view to generalising the approach and more specifically how we have applied our experiences to the development of new distance courses in Data Analytics and Artificial Intelligence at Huddersfield University.

Keywords Distance learning · Intelligent systems · Robotics · Postgraduate

5.1 Introduction

The MSc Intelligent Systems (IS) and the MSc Intelligent Systems and Robotics (ISR) programmes at De Montfort University (DMU) are Masters level courses that are delivered both on-site and by distance learning (DL). The courses are delivered mainly by the members of the Centre for Computational Intelligence (CCI) at DMU. Their development enabled the teaching team to capitalise on the research taking place within the CCI and therefore on the strengths of the staff delivering the modules. Usually there are around 60 students enrolled on the course at any time and at various stages in their programme of study.

J. Carter (✉)
University of Huddersfield, Huddersfield, UK
e-mail: j.carter@hud.ac.uk

F. Chiclana
De Montfort University, Leicester, UK
e-mail: chiclana@dmu.ac.uk

| | |
|---|--|
| Pre-programme | Induction Unit |
| Semester 1 (year 1) | Research Methods (RM) Fuzzy Logic (FL) |
| Semester 2 (year 1) | Artificial Neural Networks (ANN) Computational Intelligence Optimisation (CIO) |
| Semester 2 (year 2) | Artificial Intelligence Programming (AIP) Mobile Robots (MR) |
| Semester 2 (year 2) | Applied Computational Intelligence (ACI) Intelligent Mobile Robots (MSc ISR) or Data Mining (MSc IS) |
| Year 3 (typically in year 3 or sometimes on successful completion of 4 taught modules) | MSc Independent Project |

Fig. 5.1 The course structure

Each MSc consists of 8 taught modules and an independent project which is equivalent to 4 modules. Each module is worth 15 credits. The MSc ISR includes two mobile robots modules whilst MSc IS replaces one of these with a Data Mining module as an alternative application area for those less interested in pursuing mobile robotics work. In semester one, a Research Methods module is delivered to ensure that students develop the necessary skills to carry out literature searches, write project proposals and reports. The Applied Computational Intelligence module (ACI) gives students the opportunity to select a relevant area of interest to them and to explore it in greater depth. In this paper we discuss the approaches to delivering the courses to both distance and on-site students. Figure 5.1 shows the pattern of delivery of the modules for distance students; for full time on site students, the pattern is 4 modules in semester 1, 4 in semester 2 and the project over the summer. We also investigate the motivation of the students for embarking on such a programme with a particular focus on how they consider it could affect their employability.

The remainder of the paper is structured as follows: Sect. 5.2 discusses approaches to learning on the MSc programmes and how this fits with recognised approaches from the associated literature; Sect. 5.3 considers the lessons learned from ten years of running such a course; Sect. 5.4 considers how we might generalise from the experience of running the course, thus enabling further courses to be developed in a similar way; Sect. 5.5 summarises the work.

5.2 Approaches to Learning

This was the first distance learning (DL) programme to be developed within the faculty that made use of the virtual learning environment (VLE). It was therefore

necessary to find guidance from organisations such as the QAA and from pedagogic literature in order to develop good quality teaching materials. This has been reported in more detail in Carter et al. (2015); Carter and Pettit (2014) but is summarised here to provide background. Many of the modules employ experiential learning as described by Kolb in (1984), this is evident in particular through the assessment styles where students are encouraged to explore ideas and try out different approaches to problem solving. This is also supported by JISC (2010). All module materials are placed on the VLE which at the time of writing is Blackboard and the course leader ensures a consistent appearance on all modules.

The Quality Assurance Agency (QAA) for Higher Education (Quality Code 2014a, b) publishes codes of practise for the developers of HE courses. These support developers at both undergraduate and postgraduate levels. They also provide specific guidance on flexible and distributed learning. These are updated as technology and ideas develop and are the best place to start when working on a new course. For the MScs, we drew heavily on these to guide the development. One of the important pieces of advice that was more specifically for distance students is to make extensive use of formative feedback.

A particularly effective way of providing formative feedback has been the use of the discussion board facility within Blackboard. The way that this is employed is that on all modules except for Applied Computation Intelligence, there is a discussion board activity worth 10%. The marks are awarded for relevant and timely contributions. It is organised in such way that most weeks a short activity is provided and the students are expected to post something in response. These responses can be posted in the same week or within a week or two. The idea is that this is something comparable to looking over the shoulders of on-site students during a laboratory session or a tutorial activity session and giving some feedback, particularly where there is misunderstanding evident. For example, on the Fuzzy Logic module, the activity for the first week is to post a brief explanation of why a fuzzy membership grade is not to be confused with a probability (a common misunderstanding) and to give an illustrative example. The tutor would look through all of the posts—spending only a short time, probably no more than an hour a week for all posts—and if all is well they may simply write ‘good’ ‘fine’ on each. However, if there is a misunderstanding they can intercept and give more detailed feedback. What we have found is that the marks for the discussion board activity (awarded at the end of the module) provide the incentive for engagement and that the students then get used to it and start to post responses to each others’ posts and hence develop working relationships. It also serves as way of staff becoming familiar with how the students write and carry out their work and it can help to flag other potential issues.

We also believe it is important to provide constructive and timely feedback on any assessed summative work (the university has a 3 weeks turn around policy). Marking through the use of electronic means made this much quicker than previously requiring hard copies and is now the norm on all of the modules as is the case in almost all courses now in many universities. Making sure that feedback is of good quality is another important consideration as advised in Clark and Meyer (2008); Nicol and Milligan (2006; Wiggins 2001) and we aim to achieve this by ensuring that feedback

is explanatory and where possible provided at the place in the piece of assessed work where the errors took place. Research has shown (Clark and Meyer 2008) that this improves learning and it is an easy approach to implement. We previously reported on this in Carter et al. (2011). We take on board further advice from the QAA and try to ensure that we do not over-assess our students.

In (Goodyear et al. 2008) we meet the idea of networked learning. This is learning in which communications technology is used to promote connections between learners and their tutors and between learners and their peers. Our discussion board described earlier encourages this for each module and some module leaders have other mechanisms built into their modules (for example the neural networks module uses some peer assessment). Others use blogs, wikis and the course has a Facebook group. Blogs are often used to monitor progress with project type work (e.g., in the ACI module and for some when completing the Independent Project). Wikis are used for organisational purposes e.g., booking time slots for presentations or viva voces but also as a way of sharing findings. For example posters or presentation slides (sometimes with sound) that have been developed by students can be uploaded to a Wiki for all students to see. We leave decisions about the choice of communications technology to staff, which results in a variety of methods. Many use Skype for supervision meetings and for presentations and viva voce sessions though others choose to use the facilities within the VLE.

In more recent years, DMU adopted an approach across the university known as Universal Design for Learning (UDL). UDL aims to promote the development of learning materials that suits all learning types and was introduced partly as a response to the Government withdrawal of the Disabled Support Allowance for students with special educational needs; one such example would be those students who may need a note-taker due to dyslexia. UDL draws on the idea that natural learner variability is the rule rather than the exception. The introduction of this required all modules to undertake an initial audit to check key basic requirements of learning materials and assessments met UDL criteria. Due to the careful development of the DL materials for the MSc courses, almost all modules met these criteria completely at the first pass through this audit. A drive such as the promotion of UDL compliance, as well as serving learner variability, also promotes quality and appropriate provision of DL learning materials.

The next section provides a discussion of the lessons learned from designing and delivering such a course.

5.3 Lessons Learned

In order to find out more about the students' perspective on the course, a study (Carter et al. 2015) was carried out where the perceptions of the students were gathered and analysed. The findings of this previous work showed that most of the DL students did not feel isolated in their studies which was pleasing as this has been an important aim with the design of the course delivery. Another interesting observation from the

work was that many of the students already in employment (those studying by DL) were doing the course either to enhance their careers in terms of new potential jobs but many were studying to improve the way they carried out the job they were already doing. A noticeable proportion of students were studying with a view to doing further academic research as a result of completing the course.

From a technical perspective, some modules posed particular practical problems when designing the delivery and these evolved over the years. For example, the Mobile Robots module initially involved physically posting a Lego robot to all enrolled students. This became impractical as the numbers grew and the locations of the students became more widespread. We also found that when simplified this by sending directly to each student from an Amazon order, that Amazon does not deliver to all countries, so this was only a temporary approach. More recently, simulation software has been used for both robotics modules (previously simulation software was only used for the Intelligent Mobile Robots module), the software being open source making it much more accessible for all students. Advantages are that the students can develop more sophisticated robot simulations that they could not necessarily implement with a lego robot. The approach to the robotics module is discussed further in Carter and Coupland (2010). Some students who decide to take robotics further, e.g., for the Applied Computational Intelligence module or for their project, purchased their own robot and applied the techniques practically. The delivery of the robotics module has previously been reported in Carter and Coupland (2014).

Most modules are now able to use open source software though some still use specialist software that requires licenses for the students. In these cases, the Faculty pays for licenses for students and they are able to download and install the software on their own computers.

Another module that has proved to be particularly successful is Applied Computational Intelligence. This is a 15 credit module that takes place prior to the 60 credit project module. The assessment of ACI requires the students to investigate an area of interest to them and to develop an AI solution drawing on the techniques that they have learned in the other modules. The module is effectively a mini-project and it enables the students to practise for the 60 credit project module that they do towards the end of their course. The result is that some students extend the work they did for ACI into their project work. In some cases, the ideas have been extended further into work for a PhD. The submission of assignments for ACI required the students to write up their work as a report using an IEEE conference template. Where students produced very good work and where there was an element of novelty, these were then worked into papers for publication at conferences. This has been very popular with the students as it gives them confidence and validity in their work. It also gives them first hand training at becoming a researcher and presenting at conferences, networking with other presenters and so on.

In order to find interesting ways of encouraging both DL and on-site students to meet, we built in optional activities. Firstly, all DL students were invited to visit whenever they wanted and were welcome to attend time-tabled sessions if they chose to come to Leicester. Quite a few students did this from time to time and one of our most frequent attendees lived in Canada but came to visit DMU when his business

venture brought him to Europe. We organised two other visits most years and these were partially funded by the university. These were a robotics competition known as the Robot Challenge. Originally this was always held in Vienna and we usually took around 6 students—combination of DL and on-site each year. The Robot Challenge has grown and is now in a different location each year in the same way as a conference. We allowed students taking the ACI module to work as a team on an entry to the Robot Challenge should they wish to. As the assignment for ACI was individual, they took responsibility for different parts of the project and wrote up individual reports for submission. This worked well and there was usually at least one group, sometimes more, that opted to do this.

The other activity that we did regularly was to attend the BCS Specialist Group on Artificial Intelligence (SGAI conference held each year in December at Peterhouse College Cambridge. This was also a popular experience for both DL and on-site students. Occasionally one of the students would submit a and present a conference paper; otherwise they attended, joined in the workshops and benefitted from the networking possibilities.

5.4 Recommendations and Further Developments

The lessons learned from the development and running of this course can be summarised as a set of guidelines that can be applied to new courses. In this section we provide the guidelines and consider how these can be used to develop a new course at The University of Huddersfield: the MSc in Data Analytics.

All Universities have their own QA processes which are generally very similar so it is assumed that the guidelines will be taken in this context. These usually include input from marketing teams about the predicted popularity of the course.

- a. Content relevant to industry and research – at the forefront of the discipline
- b. Committed team with subject expertise and interest
- c. Thorough materials available and prepared well in advance, making use of such valuable resources and advice as provided by the QAA and JISC (QAA Quality Code 2014a, b; JISC 2010).
- d. Support for flexibility of style of delivery—can help with motivation/commitment from staff
- e. Support for flexibility of communication means for students—keep up to date—good advice in¹
- f. Support flexibility of pace for study (e.g. number of modules per year)
- g. Discussion board—helps to motivate and connect including DL and on-site
- h. Share materials and delivery pattern for both on-site and DL as they can support each other
- i. Don't have commitment to attend but create optional opportunities e.g. conference, competitions

¹ <https://www.jisc.ac.uk/guides/technology-and-tools-for-online-learning>.

- j. Encourage publications
- k. To keep good students interested allow for creativity in the design of assignments—many like to relate to their employment or interests or aspirations.
- l. Find interesting, subject related and inexpensive ways of enabling students to meet.

At Huddersfield University an Institute for Data Science opened recently and it provided the perfect staff base (as did the CCI at DMU) to develop a course that is in this same mould. We therefore developed a Data Analytics MSc that started to be delivered from September 2019 to both on-site and DL students. Similarly, the existing research group, PARK (Planning, Autonomy and Representation of Knowledge) provided the knowledge and research basis for a new MSc in Artificial Intelligence and similarly, a further MSc now exists, delivered in both modes—this is in Cyber Security.

It is interesting to note that the experience of developing and delivering these courses made the difficulties presented by COVID much more manageable, as many staff were already familiar with delivering modules online—mainly asynchronously but in some cases with synchronous classes too.

5.5 Conclusions

In this paper we have described the MSc in Intelligent Systems and MSc Intelligent Systems and Robotics. As courses that run both on-site and by DL they are often used as an example in our own institution, resulting in the development of further taught PG courses delivered using the same pattern.

Delivering courses at a distance is a topical area. With the many available mechanisms for interacting with learners electronically there are a number of choices to be made regarding the approach to take. In this paper we have described some of the approaches taken to the delivery of the learning materials and our approaches to assessment and feedback. These approaches have been seen as valuable assets that support the development of e-learning both in terms of trying and testing appropriate technologies but also addressing pedagogic issues that arise when delivering distance and e-materials.

Embracing UDL as a university (DMU) reinforced the approaches taken in the development of the learning materials for the MScs. It drove the development of good practice generally and benefited the development of the DL provisions, ensuring that good quality materials to suit a wide audience were developed.

The courses at DMU have shown that this kind of delivery is successful and sustainable. It has changed in the mechanisms for delivery as technology has evolved and the content is reviewed frequently to maintain currency of the content. The lessons learned over the years have provided a template for other courses at DMU and as noted earlier, has inspired the development of the development and both on-site and DL delivery of MSc courses at the University of Huddersfield.

We have provided a checklist for others who wish to develop courses that are delivered in a similar way and believe that wider approaches to learning—distance, on-line, blended—are the way forward and suit the wide variety of learners that wish to study at this level.

References

- Carter J, Coupland S (2010) Teaching robotics at the postgraduate level: delivering for on site and distance learning. In: Proceedings of the international conference on robotics in education (RIE2010)
- Carter J, Coupland S (2014) Robotics for distance learning: a case study from a UK masters programme, HEA 2014. In: EM Annual Conference 2014
- Carter J, Pettit I (2014) E-learning for employability: a case study from a UK master's programme. In: Motta G, Wu B (eds) Software engineering education for a global E-service economy. Progress in IS. Springer, Cham
- Carter J, Matthews S, Coupland S (2011) Teaching robotics at the postgraduate level: assessment and feedback for on site and distance learning. In: Proceedings of the international conference on robotics in education (RIE2011)
- Carter J, Chiclana F Al-Omari M (2015) E-Learning for distance students: a case study from a UK masters programme, HEA 2015. In: The European conference on technology in the classroom 2015
- Clark R, Meyer RE (2008) E-learning and the science of instruction. Wiley
- Goodyear P, Banks S, Hodgson V, McConnell D (2008) Advances in research on networked learning. Springer, pp 1–10
- JISC (2010) [Online] Designing interactive assessments to promote independent learning. http://www.jisc.ac.uk/media/documents/programmes/elearning/digiassess_interactiveassessments.pdf
- Kolb D (1984) Experiential learning: experience as the source of learning and development. Prentice-Hall, New Jersey
- Nicol DJ, Milligan C (2006) Rethinking technology-supported assessment in terms of the seven principles of good feedback practice. In: Bryan C, Clegg K (eds) Innovative assessment in higher education. Taylor and Francis Group Ltd, London
- QAA Quality Code, Part B (2014a) Assuring and enhancing academic quality, Chapter B3. <http://www.qaa.ac.uk/assuring-standards-and-quality/the-quality-code/quality-code-part-b>
- QAA Quality Code, Part B (2014b) Assuring and enhancing academic quality, Chapter B6. [Online] <http://www.qaa.ac.uk/assuring-standards-and-quality/the-quality-code/quality-code-part-b>
- UDL. <https://www.cast.org/impact/universal-design-for-learning-udl>. accessed 30/03/23
- Wiggins G (2001) Educative assessment. Jossey-Bass, San Francisco

Chapter 6

Academic Integrity for Computer Science Instructors



Thomas Lancaster

Abstract For Computer Science instructors, upholding academic integrity requires approaching teaching and assessment in a way that encourages students to follow progressive principles such as honesty, trust, fairness, respect and responsibility. At the same time, instructors also have to take steps to minimise the risk of academic misconduct by students, whether intentional or accident. This means that instructors need to be aware of behaviours such as plagiarism, collusion, contract cheating, the use of machine generated answers, examination cheating and research fraud. Instructors also need to put measures into place to design out opportunities for students to engage in unacceptable behaviours. This chapter explores academic integrity from the perspective of the knowledge needed by a Computer Science instructor. This is a changeable feast, as new methods to subvert academic integrity are always emerging, particularly in Computer Science where students are already technically sophisticated and understand how to use artificial intelligence-based solutions that can solve many standard assessment tasks. As such, the chapter recommends that although detecting breaches of academic integrity may be necessary, instructors should lead by example, work in partnership with students and develop a curriculum with a pro-active focus on academic integrity.

Keywords Academic integrity · Academic misconduct · Plagiarism · Contract cheating · Artificial intelligence

6.1 Background

The first version of this chapter (Lancaster 2018) was written and published prior to the Covid-19 pandemic. This updated version has been written after a period that saw teaching, learning and assessment shift rapidly online. During the pandemic, students sometimes felt that they lacked the support they would have had in a face-to-face setting. Many people in the academic community were under pressure, not

T. Lancaster (✉)
Department of Computing, Imperial College London, London, UK
e-mail: thomas@thomaslancaster.co.uk

least students themselves. Such pressure could sometimes lead to students using unauthorised sources of support, a behaviour that could be referred to as cheating.

This revised and updated chapter retains much of the text and many of the ideas from the original version, but it has been updated to reflect wider societal changes. When reading it, instructors are encouraged to consider the challenges faced by students which may lead to academic misconduct, as well as the threats to academic integrity posed by artificial intelligence systems. These are the very systems that computer science students themselves may be working to develop.

6.2 Introduction

Academic integrity describes the framework of ethical understanding that everyone involved in the educational process must strive to work within. This chapter considers academic integrity primarily from a teaching and learning standpoint, looking at how instructors can ensure that students take full advantage of the learning opportunities available to them, engage productively with the higher education community and prepare themselves for a professional career. Academic integrity is often presented alongside a necessary parallel, yet less positive message that says students must not cheat. As such, the research and teaching practice on academic integrity lies closely intertwined with that related to student plagiarism.

For the Computer Science instructor, communicating and teaching the principles behind academic integrity offers challenge. Industry practices require students to gain experience collaborating and working in teams. Software development often involves the reuse of existing code bases. To ready for industry, students need to demonstrate that they are skilled at using online resources to find solutions to the problems they come across during the programming process. They may also have to show they can use the latest artificial intelligence-based software that generate code for them, rather than develop this from scratch in a way that a traditional educational assessment may have required. Students therefore need to understand the difference between acceptable and unacceptable conduct in a variety of educational and work-like situations.

The Computer Science instructor too must operate within the same academic integrity constraints as their students. This means that they must lead by example, taking every opportunity to design learning resources that remove opportunities for students to accidentally breach academic integrity norms. They also need to acknowledge their own sources.

This chapter addresses academic integrity for the Computer Science discipline by providing practical ideas and considerations. After a brief introduction to academic integrity terminology and challenges, the chapter focuses on three specific problems: (1) introducing the importance of academic integrity to students; (2) engaging students with assessment and reducing the temptation for them to cheat; and (3) detecting when academic integrity has been breached. Detection, although somewhat counter to presenting academic integrity as a set of positive virtues, remains

necessary since it is not fair for students to gain qualifications they do not deserve at the expense of others who are putting effort into their studies.

This chapter is intended to be of use to all Computer Science instructors of all experience levels looking to ensure academic integrity is upheld within their teaching. It is also hoped that the chapter will be of use beyond the Computer Science discipline, particularly for other practical and industry led subjects.

No attempt is made in this chapter to provide a source for every idea. It includes, for example, many now standard observations and recommendations on assessment design. These have been widely mentioned in the literature, including in other papers by the author and it would be difficult to credit these to a single definitive source. Interested parties who would be interested in current research and future research opportunities within the field may be interested in coverage by Lancaster et al. (2019). This chapter is intended to collate the relevant practical information in a succinct form suitable as a primer for the modern Computer Science instructor.

6.3 Academic Integrity in Computer Science

The Exemplary Academic Integrity Project (2013) defines academic integrity as:

acting with the values of honesty, trust, fairness, respect and responsibility in learning, teaching and research. It is important for students, teachers, researchers and professional staff to act in an honest way, be responsible for their actions, and show fairness in every part of their work. All students and staff should be an example to others of how to act with integrity in their study and work. Academic integrity is important for an individual's and a school's reputation.

The definition is useful as it reverses the traditional way in which academic integrity is presented, which is simply by telling a student what they may not do. The definition provides a set of values for students to follow during their academic life. It further notes that these values apply equally to everyone involved in education, which also includes instructors.

Despite an attempt to focus on the positive, any practical chapter on academic integrity would be incomplete without indicating the unacceptable behaviours that students have been observed using. An overarching term for such behaviours is academic misconduct, although informally they are often referred to as cheating. Several such methods most relevant for which knowledge is most useful for Computer Science instructors are summarised in Table 6.1.

The definitions in Table 6.1 are not intended to offer a complete list of the ways in which academic misconduct can be breached. Such a list can never be complete so long as new methods of teaching and assessment are developed and these are accompanied by revised methods of cheating. Academic misconduct can include any activity intended to give a student an unfair advantage over other people. Other examples include students using social engineering techniques to gain access to early copies of exam or hacking into computer systems to change their marks. There are

Table 6.1 Types of academic integrity breaches of interest to the computer science instructor

| Academic integrity breach | Description |
|---------------------------|---|
| Plagiarism | The process where a student uses the words or ideas of another without acknowledgement. A particular issue in Computer Science is <i>source code plagiarism</i> , where program code is copied or reused in an unacceptable manner |
| Collusion | Where two or more students have worked closely together during the production of assessed work, going beyond acceptable levels of collaboration. When collusion is taken to the level where the submitted work of two students is identical or very similar, this also represents a form of plagiarism |
| Contract cheating | The behaviour where a student uses a third party to complete assessed work for them, or attempts to do so (Clarke and Lancaster 2006). This involves an exchange of money or benefits for the parties involved. Contract cheating was originally observed primarily relating to the outsourcing of programming and technical Computer Science assignments, but also covers written work, such as reports. Some contract cheating is facilitated through <i>essay mills</i> , companies that are primarily online and which offer to produce original bespoke assessments for students |
| Machine generated answers | This covers the situation where technology is used to generate answers in the style that would be produced by a human, but in a manner unauthorised for assessment. This may involve the use of artificial intelligence and machine learning based software. Systems can construct realistic looking text, source code and images |
| Exam cheating | Where a student attempts to gain an unfair advantage in an in-person or online examination. For in-person exams, this may include referring to concealed notes or communicating in secret with an individual outside an exam hall. For online exams, this could include posting questions on online sites to receive answers (Lancaster and Cotarlan 2021). A related behaviour is <i>impersonation</i> , where a third-party switches places with a candidate and attempts to evade detection, either physically or online |
| Research fraud | The process where research results and conclusions are returned that are not backed up by verifiable evidence. This can include deliberate errors in the research methodology, such as mistreating or ignoring data. This can also include not acting within a defined ethical framework. A specific example is <i>data fabrication</i> , where reliable data is not collected but instead produced in a way that benefits the student in achieving the result or conclusion they would like to see. In Computer Science, this may perhaps be an issue during the Project stage of a student's course |

also areas where instructor opinions of what constitutes academic misconduct may vary, such as where a student uses chemical enhancements, such as study drugs, designed to allow them to concentrate for longer and hence gain a higher mark than their peers.

There are also areas where academic integrity can be breached, but this may not be deliberate. For example, a student could be judged to have plagiarised as a result of poor study skills. They could have committed research fraud through misunderstanding research methodologies or incorrectly processing data. This is why providing education and support for students is important, as is ensuring they know how to access such support when necessary. Addressing this once an academic integrity breach has taken place is too late. This is because it is not always possible to tell if this was intentional or accidental.

A suitable understanding of what plagiarism is and why this is considered a problem will help instructors trying to extend their knowledge to other forms for academic misconduct. Many commentators have linked a growth in academic misconduct with the prevalence of Internet aided plagiarism (Austin and Brown 1999). Students entering higher education have grown up surrounded with the Internet for all of their life and so their concept of the value and ownership of information may be different to that of their instructors.

Computer Science instructors also have to be aware of the types of academic misconduct that appear to be becoming more visible. For example, research into contract cheating has regularly seen assessments from across the Computer Science spectrum outsourced (Jenkins and Helmore 2006; Lancaster and Clarke 2007). Technical assignments, particularly those which do not require English language proficiency, can be contracted out to a global marketplace, often for a cost that is economically feasible for students. The range of assessments that can be outsourced contains activities contributing strongly to a student's final degree classification. This includes major work such as a final year project, assessments completed in stages and tasks where students are intended to provide their own personal reflection.

The use of examinations may remove some of the academic integrity challenges seen with coursework, but examinations themselves can be susceptible to contract cheating. Students have been observed paying or using a third party to provide them with undocumented help in an exam situation (Lancaster and Clarke 2017; Lancaster and Cotarlan 2021). With the continual move to Computer Science programmes being delivered online, instructors need to ensure that assessment is designed to promote learning and to remove vulnerabilities to academic integrity.

6.4 Teaching Academic Integrity Principles

Students need to be equipped with the necessary tools for success as part of their Computer Science course. This includes helping students to develop an understanding of how they can progress through their studies whilst always holding academic

integrity in their mind. For the instructor, this requires supporting students to avoid accidentally breaching academic integrity principles.

It is unlikely to be sufficient for an instructor to assume that students arriving at university all share a common understanding of what academic integrity is, why it is important or how they can act with integrity. This can particularly be the case where students arrive from different cultures, educational upbringings and backgrounds.

This means that teaching academic integrity is necessary. Such teaching needs to cover two main considerations.

First, students need to understand what academic integrity means, why this is essential to their studies and how academic integrity values relate to their everyday life and future careers.

Second, students need to be shown the practical techniques they will require during their course. These are the techniques necessary for them to avoid academic integrity breaches and to show that they respect the resources that they are working with.

As with many subjects, academic integrity is not a subject that can just be taught once and forgotten about. One model that can be used is to introduce students to the appropriate concepts early on in their course, integrating opportunities for students to receive formative feedback on how well they have engaged with the material. These concepts can then be developed and reinforced throughout the course, with students provided with information specific to the subject they are working on and their level to study.

For example, when arriving at university, students may need to be shown the practical skills of finding and referencing textual information very early on. This teaching would be supported by exploring why valuing the ownership of information is important and the possible consequences that taking information from a source without acknowledgement would lead to in their everyday life. In a career situation, this could lead to an employee being dismissed, Intellectual property breaches could also lead to litigation for the company they work for.

Discussions about other types of academic misconduct and how this can be breached can occur at appropriate points during the student journey. For instance, correct examination conduct and the skills needed to be successful with exams can be explored early on in their course before students take low risk tests. In Computer Science, presentation of the concept of examination success can also be related to the professional examinations, such as vendor certifications. These examinations are ones that students may be expected to take beyond their time at university and throughout their career.

Introducing a discussion on contract cheating can be more difficult. Some instructors may believe that all students already know this is wrong, but such a view does not account for differences in student cultural upbringing. There are cultures where students are taught that using the words of experts is not only allowable, it is expected. There are also students who already know that this is wrong and are still contract cheating. The counter view held by some students, that information belongs to them once they have paid for or acquired it, needs to be openly explored.

Challenging conversations can be framed as part of a wider discussion regarding what level of external help is permissible. This is an important distinction when attempting to build Computer Science student team working skills ready for industry, but still ensuring that students receive credit for their own individual effort and contributions.

Many academic integrity discussions also need to happen at the level of individual subjects. A student learning introductory programming skills does need to demonstrate that they have mastered basic coding concepts, even though the level of exercises set may be trivial to many experienced programmers. This means that taking a code fragment from an existing online codebase to solve a simple exercise regarding manipulating strings, for example, would not be considered appropriate at such an early stage of the student's academic journey. Nor would typing such a request into an artificial intelligence system to generate the source code needed. Once a student is required to demonstrate that they are ready for a professional programming career, the expectations on them may be different. In such a case, it may be considered both acceptable and desirable for a student to demonstrate that they can find and reuse existing code fragments.

There may be further complications when a student needs a resource to complete an assessment, but that resource itself is not a key area that is being examined. An example may be on a web development assessment, where a student wishes to improve the look of their website by using existing artwork, but it is their coding skills that are being assessed. In such a case, students need to be made aware of the instructor's expectations of how they demonstrate academic integrity.

For Computer Science, Simon et al. (2016) recommend that exactly what students are allowed to do and what they may not do is spelled out for them on the assignment brief. This may include detailing which individuals and services they can and cannot get help from, which resources they may and may not use and what aspects of the assessment they can and cannot solicit help with. It may also be worth agreeing such a list with students as part of the assessment development regime, thus also helping the student cohort to feel some ownership of the decisions made.

One message that is worth communicating to students is that academic misconduct is not a victimless crime. A student who used dishonest tactics to get ahead could end up getting better marks than a student that worked hard. A cheating student could even get a job that they don't deserve, taking this away from a more honest student. That is why it's important for instructors and students to work in close partnership and develop a shared understanding of why academic integrity is important for all parties operating within the higher education landscape.

6.5 Assessing with Academic Integrity

Both students and instructors have a role to play in ensuring that academic integrity is maintained through the assessment process. As this chapter has already discussed, students need to understand the benefits of completing their own work and commit

to doing this to the best of their ability. Students also need to be equipped with the academic skills necessary for their own success.

Instructors have further responsibilities during the assessment process, an area which students often consider the most important part of their educational experience. Instructors need to develop assessments that reduce the opportunity for students to accidentally breach academic integrity. They should also try and remove the temptations that may mean that some students would choose to cheat if they thought that they could get away with it. This can be accomplished by rethinking and restructuring the assessment process, or by reducing any benefits that attempting to cheat may bring.

Student interest in assessment can be improved by making this more engaging. Mechanisms for doing this will vary between student cohorts, so some partnership working with students is necessary here. It may, for example, be that a particular student group prefers examinations over coursework, group assessments over individual assessments or practical tasks over written ones. It may be possible to co-design assessments with students so that they feel some ownership of the assessments tasks that are set.

There may also be cases where students feel that assessments are simply hurdles that they need to jump. Working with students can help to share the message that these are measures of progress. Demonstrating to students that formative assessment is useful can also pay dividends, particularly in the areas of Computer Science where students need to build up skills gradually, such as when learning computer programming.

Assessments can be developed so that students cannot gain a passing mark by using repeatedly cheating using the same techniques. A simple way of doing this would be to set two different types of assessment within a subject, such as both a coursework and a test. In such a case, the students would be required to demonstrate their competency in both items. In this particular circumstance, a student who had plagiarised their coursework and not been caught would still have to find a different way to cheat during the supervised test element. As well as engaging the students using multiple assessment modalities, this also reduces the benefit of cheating, since the required learning would still need to be completed for the second element of assessment.

A supervised component of coursework assessment can also be useful, particularly where instructors are worried about contract cheating (Lancaster and Clarke 2016). This can include both written tests and practical tests, but also supervised coursework sessions, presentations, spoken viva voce examinations, industry style coding interviews and product demonstrations. There are many opportunities to innovate here.

Employability initiatives are also worthy of consideration. For instance, more authentic forms of teaching and assessment can be used, such as a simulated work environment where students work on assigned projects during office hours to develop and document a solution to a problem posed by an external client. Such assessments can also be of value for students when they looking to present a professional portfolio and to showcase on their Curriculum Vitae that they have acquired a wide range of new skills during their course.

Instructors do need to be wary about the need to balance several trade-offs when developing assessments. They have to balance encouraging academic integrity, reducing the value of academic misconduct, meeting industry requirements and their own ability to manage the assessment process within their wider workload commitments. As an example, an individual spoken examination, or *viva voce*, may be a very good method of authenticating that a student knows and understand a subject. Despite the advantage, the time commitment necessary to do this fairly for all students in a large Computer Science class may mean that a spoken exam is simply not practical. Even if the time required is not the issue, there are still issues of fairness to consider, such as how the instructor ensures that students who have their *viva* early in the process are not at a disadvantage over students scheduled later on. The latter group could be said to have both more time to prepare and the potential to quiz earlier students over the questions that they are likely to be asked. There can also be challenges posed to the consistency of processes when a large team of assessors is used.

Compromises are possible and recommended. It may be that instructors work together to ensure a varied diet of assessments across the whole year of a course, with extra academic integrity measures put into place during those assessments that underpin later learning. For instance, the crucial assessments in the first year of a traditional Computer Science course might be considered to be those for the core programming and mathematics subjects, an understanding of which is needed for success in later years. It is also possible to set an end-of-year *viva*, bringing together the skills from across a variety of subjects and helping to check that students have an understanding of how these distinct areas all integrate together. Such a *viva* can also be resourced using multiple instructors to ensure that this process is manageable. Major modules, such as a final year student project that often represents the culmination of a degree, can also have extra academic integrity checkpoints put into place.

Another option for assessment that combines several recommendations of good practice is for an instructor to structure this using multiple linked components. Here is a simple example regarding how this could be used for a computer programming module, but similar approaches are also possible for other subjects. First, the student completes a coursework assessment as usual, where they develop a solution to a programming problem. They submit their code for that coursework. At a scheduled time, they attend a practical examination where they are required to make changes to the code they already submitted. This approach requires students to be familiar with their code, reducing the usefulness of them having outsourced its production or used other cheating methods. Where students chose not to engage with academic integrity during the coursework process they would still have to master the programming skills required for the practical exam. The student should realise that there is little benefit to them not putting in the effort during the coursework phase, hence encouraging them to concentrate properly on the assessment.

Most of the examples in this part of the chapter have focused on coursework assessment, but examinations are also susceptible to cheating. Even an activity like a *viva voce* examination can be cheated on if a student has advance access to a set of standard questions, or if someone outside the exam room can feed the student answers

through a concealed earpiece. Some examination practices used by instructors are themselves questionable, for instance where the questions have been taken from previous papers that students have access to, or match those provided in a sample paper. This would make the exam merely test of memory skills and not ability. Hence, instructors need to be continually aware of opportunities where their own practice can lead to the manifestation of academic misconduct.

Academic integrity in examinations can be further preserved by carefully developed examination processes. This includes specifying the equipment that students are allowed to use, supplying equipment they cannot have tampered with such as pens and calculators, monitoring and recording screen activity during practical examinations, and ensuring that all examination questions are original and go through a robust quality checking and moderation process. If nothing else, all examinations need to be invigilated and students alert to the fact that their actions are being watched, thus removing the benefit from any opportunistic attempts by students to cheat.

6.6 Detecting Breaches of Academic Integrity

Instructors need to be aware that detection tools and punitive measures are not a single solution to academic integrity challenges, but they can be used to support other approaches. Tools can also provide instructors with some assurance where they believe that academic integrity has been breached. Many software tools already exist for the detection of breaches of academic integrity and the choice of the most suitable tool has to be closely matched to different situations. But software cannot be considered a complete or infallible solution. It may always be possible for students to cheat in ways that tools do not detect. Many web pages, videos and social media posts exist where students share the latest ideas to defeat detection technology.

Tools for detecting traditional forms of plagiarism are available and this is a field that is well-established. These tools are available to identify similarity in written text, within source code submissions in many programming languages and for specific technical environments such as spreadsheet assignments or database assignments. These tools flag student submissions that warrant further manual investigation by an instructor. For instance, this may include a written student submission where sections of the work match internet sources. The results from the tools do not directly indicate plagiarism, so human judgement is required. There are many acceptable reasons why a tool may give such a false hit, for instance two textual documents may appear to match each other, but the cause may turn out to be the use of a correctly cited quotation that was common to both documents.

Although plagiarism detection tools are well-established, tools designed to detect contract cheating or generated solutions are not so readily available. Indications are that students are turning to third party writers, programmers and other systems precisely because the original work they supply is unlikely to be detected using current plagiarism detection technology.

The best advice available at present is for instructors to approach assessing coursework with a keen eye and questioning mind, with the view that that someone other than the student may have produced this item of coursework. For an instructor who knows a student, their abilities or their writing style, such simple checks as looking at the properties of the document they submitted can be useful. These may show suspicious file creation dates or author names.

Several techniques that could be used to support the automated detection of contract cheating have also shown promise. Checks that the writing style of a student remains consistent from one assessment to the next are possible. Some work has been made to automate such checks (Juola 2017). Similar techniques could be used to decide if student programming style remains consistent from one exercise to the next. It has been suggested that contextual information could be extracted from student submissions to help develop artificial intelligence systems to detect contract cheating (Lancaster and Clarke 2014). Student mark profiles can also be analysed to see where student performance shows inconsistencies of the type that may suggest that the student has received external help (Clare et al. 2017). Forensic examination of files submitted by students is also possible (Johnson et al. 2022).

The use of tools to detect academic misconduct is an important part of any complete academic integrity process. Using tools protects the value of the results and overall qualifications awarded to the students who are working with academic integrity. Their use shows that academic integrity is being treated seriously. The existence of such tools and evidence of their use also provides a deterrent effect to students who may otherwise have considered cheating, but have changed that view due to the risk that they may be caught.

If an investigation suggests that a student has breached academic integrity regulations, it is important that a fair, transparent and consistent approach is followed. This means that all students in this situation are considered using an identical process and treated in the same way. Doing this ensures that the wider higher education setting also complies with the underlying principles of academic integrity. Where a fair process does identify that a student has breached academic integrity, this also has to lead to a suitable outcome, such as a fairly and consistently applied penalty for the student. There may also be the need for arrangements to support the student in future to be put into place.

The same technology used to detect plagiarism and potentially penalise students can also be used in a positive manner. For example, plagiarism detection software can be introduced in a formative way to allow students to find out if they are accidentally demonstrating poor academic practice (Halgamuge 2017). As this is formative, the process can then be used to put support into place for students, rather than attempting to penalise them. For instance, where a student draft report is run through plagiarism detection software and found to contain copied text, the student could be provided with support on how they should correctly acknowledge and reference information from an external source. Such support is useful for students to ensure that they do not accidentally breach academic processes when in a summative situation.

6.7 Conclusions

An understanding of academic integrity, including how academic integrity can be promoted to students and how academic integrity should be continually maintained, is important knowledge needed by all current Computer Science instructors. This chapter has motivated the need for academic integrity in Computer Science education and has provided an overview of ideas within the field.

When students cheat, many will have considered that they are taking a risk in return for a potential reward. For such a student, cheating has to offer them some sort of benefit, such as passing an assessment point or obtaining a better mark than they otherwise would have done. That benefit has to outweigh the chance of them being caught and the likely knowledge of what will happen to them if they are found to have breached academic integrity.

Some students cheat because they do not know any better. That risk can be mitigated against by ensuring that a programme of academic integrity tuition and support is a core component of study for all students.

Other students cheat because they fear failure and believe that their academic misconduct is unlikely to be detected. That risk to integrity can be reduced through careful instructional design and considered assessment processes. Sometimes, this fear can be due to wider pressures that the student is facing that are preventing them from studying. Access to appropriate pastoral support in an understanding academic environment is also needed.

There are students who cheat because they do not see the value in what they are being taught or the assessments they are being asked to do. This risk can also be mitigated against by developing engaging teaching and incorporating real world examples.

In Computer Science, there are also students who have the ability to complete assessment for themselves, but just want to show that they can beat the system. This is an example of the long-standing hacker mentality already evident in the Computer Science population. There is no single solution to prevent this, but here simply ensuring that students are kept engaged and interested may offer the key.

A further complication lies in the development of improved artificial intelligence-based systems designed to replace tasks usually completed by students. Many Computer Scientists are working on such tasks as part of their research. Students may themselves complete artificial intelligence and machine learning modules, work on capstone projects in such areas, and progress to working further on such systems in research and commercial roles. An academic who develops a system to better write text in the style of a human is also contributing to systems that can be misused by students if they so desire. There is a difficult balance to strike between developing systems to ensure that integrity of academic processes, but also working with technology that is being used in the world outside of academia.

Academic integrity within the Computer Science discipline can also be supported by using a wide variety of assessment types, including authentic assessment and work

simulations. Not every career path has the same view of integrity, even though documents such as professional body authorised codes of conduct offer some general principles. Within Computer Science education, there are areas where the best solution towards ensuring academic integrity is not clear, such as where the border between unacceptable collusion and expected workplace collaboration is concerned.

The opportunities for Computer Science academics to develop processes, definitions and solutions for the preservation of academic honesty also offers opportunities for these academics. Many academic integrity challenges, such as those linking employability and collaboration, are likely to be more prevalent in Computer Science than in other academic disciplines. This also means that the prospect exists for the Computer Science specialist to take a leadership role in the academic integrity field.

One of the most positive movements towards making academic integrity a cornerstone of the establishment has been students and staff working together in partnership (Lancaster 2022). There are many examples of students raising awareness, conducting academic integrity research projects and co-designing assessments, really helping academic integrity to be seen as something of importance to the whole academic community. Academic integrity needs to be more than just an add-on to Computer Science. Instructors working in this field need to be operating with students in mind and with the principles of academic integrity at the heart of the discussion.

References

- Austin M, Brown L (1999) Internet plagiarism: developing strategies to curb student academic dishonesty. *Internet High Edu* 2(1):21–33
- Clare J, Walker S, Hobson J (2017) Can we detect contract cheating using existing assessment data? Applying crime prevention theory to an academic integrity issue. *Int J Edu Integrity* 13(1)
- Clarke R, Lancaster T (2006) Eliminating the successor to plagiarism? Identifying the usage of contract cheating sites. In: 2nd Plagiarism: prevention, practice and policy conference 2006, Newcastle, United Kingdom
- Exemplary Academic Integrity Project (2013) Resources on academic integrity. <https://lo.unisa.edu.au/course/view.php?id=6751§ion=6>. Accessed 18 Apr 2018
- Halgamuge M (2017) The use and analysis of anti-plagiarism software: Turnitin tool for formative assessment and feedback. *Comput Appl Eng Edu* 25(6):895–909
- Jenkins T, Helmore S (2006) Coursework for cash: the threat from online plagiarism. In: Proceedings of 7th annual higher education academy conference in information and computer sciences. Dublin, Ireland
- Johnson C, Davies R, Reddy M (2022) Using digital forensics in higher education to detect academic misconduct. *Int J Educ Integr* 18:12. <https://doi.org/10.1007/s40979-022-00104-1>
- Jones M, Sheridan L (2015) Back translation: an emerging sophisticated cyber strategy to subvert advances in ‘digital age’ plagiarism detection and prevention. *Assess Eval High Educ* 40(5):712–724
- Juola (2017) Detecting contract cheating via stylometric methods. In: *Plagiarism across Europe and beyond 2017*, Brno, Czech Republic
- Lancaster T (2018) Academic integrity for computer science instructors. In: Carter J, O’Grady M, Rosen C (eds) *Higher education computer science*. Springer, Cham. https://doi.org/10.1007/978-3-319-98590-9_5

- Lancaster T (2022) Addressing contract cheating through staff-student partnerships. In: Eaton SE, Curtis GJ, Stoesz BM, Clare J, Rundle K, Seeland J (eds) *Contract cheating in higher education*. Palgrave Macmillan, Cham. https://doi.org/10.1007/978-3-031-12680-2_15
- Lancaster T, Clarke R (2007) Assessing contract cheating through auction sites—a computing perspective. In: 8th Annual higher education academy conference in information and computer sciences, Southampton, United Kingdom.
- Lancaster T, Clarke R (2014) An initial analysis of the contextual information available within auction posts on contract cheating agency websites. In: *Proceedings of international workshop on informatics for intelligent context-aware enterprise systems (ICAES 2014)*, University of Victoria, Victoria, Canada
- Lancaster T, Clarke R (2016) Contract cheating—the outsourcing of assessed student work. In: Bretag T (ed) *Handbook of academic integrity*. Springer
- Lancaster T, Clarke R (2017) Rethinking assessment by examination in the age of contract cheating, plagiarism across Europe and beyond 2017. Czech Republic, Brno
- Lancaster T, Cotarlan C (2021) Contract cheating by STEM students through a file sharing website: a Covid-19 pandemic perspective. *Int J Educ Integr* 17:3. <https://doi.org/10.1007/s40979-021-00070-0>
- Lancaster T, Robins A, Fincher S (2019) Assessment and plagiarism. In: Fincher S, Robins A (eds) *The Cambridge handbook of computing education research (Cambridge handbooks in psychology, pp 414–444)*. Cambridge University Press, Cambridge. <https://doi.org/10.1017/9781108654555.015>
- Simon, SJ, Morgan M, Petersen A, Settle A, Sinclair J, Cross G, Riedesel C (2016) Negotiating the maze of academic integrity in computing education. In: *Proceedings of the 2016 ITiCSE working group reports*, pp 57–80.

Chapter 7

Contextualisation in Data Science



Marcello Trovati

Abstract In this chapter, contextualisation is discussed with particular emphasis to the delivery of data science courses and programmes. Data science has seen a significant increase in popularity due to its applications to business and academic related fields. Data is being continuously created and therefore, it is crucial to create methods, frameworks and implementations to extract, assess and utilise actionable information for data. Data is, and will be, the new currency. As a consequence, data science must be embedded in the curricula from an early age. The objective of contextualisation includes the integration of learning with various topics and disciplines, to provide *meaning*. VR/AR technologies can further enhance contextualisation within data science, due to their ability to engage and interact with users.

Keywords Data science · Contextualisation · Pedagogy · Artificial intelligence · VR/AR technology

7.1 Introduction

Data Science has become one of the most popular scientific areas, whose applications include physics, mathematics, biology, as well as qualitative disciplines such as humanities, sociology, psychology, just to name a few. The central and most crucial concept in data science is *information*. In a world where data is continuously created, information has become even more important. Before we proceed any further, it is essential to emphasise the difference between data and information. Data is meaningless, in the literal sense. It might be related to sensor readings, temperature, wind direction, or even letters on a piece of paper. Information, on the other hand, consists of the relationships between and the connectivity of the data. Letter are connected to create words, which generate sentences and so ultimately, convey meaning.

It follows from the above paragraph that data science cannot be taught in isolation from the context in which it is applied as data has no inherent meaning. The tools

M. Trovati (✉)

Department of Computer Science, TechHub, Edge Hill University, Room THF-14, Ormiskirk L39 4QP, UK

e-mail: marcello.trovati@edgehill.ac.uk

and techniques used in data science have therefore evolved and been enriched by other domains and influenced by the multidisciplinary nature of the subject area. When teaching data science as a subject, it is therefore crucial to enable students to appreciate this collective endeavour. The obvious question is how can this be achieved?

7.2 The Challenges of Data Science Education

DeClue (2009), Lewis and Stoyanovich (2022) discuss the main challenges which data science education needs to address. They are grouped into four categories:

- Issues related to the inherent interdisciplinary structure of data science
- Cognition challenges stemming from the high cognitive skills necessary for learning data science
- Challenges related to the necessary skills to become a professional data scientist
- The challenges posed by the implementation side of a data science programme.

The authors explicitly describe all the challenges related to data science curricula based on such categories.

1. Content

- a. Need to design a balanced programme integrating the essential interdisciplinary nature and components of data science
- b. Data science has become embedded in many domains. As a consequence, the integration of the appropriate amount of knowledge of these other domains in the data science curriculum can be problematic.
- c. Interdisciplinarity may lead to ethical issues, which need to be acknowledged.

2. Cognition

- a. Data science lies in the intersection of computer science, mathematics, and statistics, which typically require an appropriate educational background and demand a high cognitive workload. Specific pedagogical strategies need to be put in place to widen the appeal of data science and encourage participation.

3. Skills

- a. A professional data scientist has to exhibit numerous skills, including non-technical skills. What are the essential non-technical skills that need to be developed and how can they be integrated throughout the educational programmes?
- b. Research skills are also necessary, which require a greater level of academic maturity and independence.

4. Implementation

- a. Interdisciplinary-based learning can be challenging both for students and instructors

- b. The technical skills required by the data scientist are only half of the equation. Understanding the problem domain and being able to appropriately apply the skills to create new insights into that domain is the other side of the coin. Data science programmes therefore need to be designed therefore so that they are accessible to a wider population so that students who lack extensive mathematical and computer science knowledge might still be able to engage with and participate in them.
- c. Data science should be introduced at different levels and not only as part of HE curricula. However, clear and well structured teacher training initiatives related to data science delivery have to be deployed

The above issues are still being debated, but some initial insights can be shared across the academic communities. One aspect that is becoming evident is the need for *contextualisation*. This is the topic of the rest of this chapter.

7.3 Contextualisation in Data Science

Data science curricula are ideally suited to take advantage of a number of new technologies and opportunities that enable programmes to be tailored to the interest of students. These include:-

- The Alice environment, robotics, virtual and augmented reality
- Data producing research
- Industrial collaborations
- Real world data

Furthermore, students are introduced to that incorporate the multidisciplinary nature of computation and its use. Real-world problems have become an integral part of curricula. Industrial partners actively contribute and can help design some teaching modules, as well as support their delivery.

Barnes (2002), defined contextualism as the interpretation of events within their corresponding contexts, which provides a pragmatic and dynamic approach to learning. In particular, such contextualisation is rooted in the realisation that topics and fields are interconnected, motivated by numerous instances of multidisciplinary applications, theoretical frameworks and foundations. Context can also be defined as the circumstances in which a conceptual entity is found, and more importantly, how the concept interfaces with the environment. This is a fundamental aspect, as it further emphasises the role of multi-disciplinarity.

DeClue (2009) discuss of the relevance of context to the learning process, as it provides meaning and a motivational approach to learning. In data science, often technical terminology, their principles and properties are seen as the primary objectives in the learning process. However, such concepts have little meaning to students. For example, the formalism of mathematics and theoretical computer science is based on a structured sequence of definitions, propositions, theorems, and corresponding

proofs. Undoubtedly, this is how such disciplines are *constructed*, not how they were introduced. Any advance in data science and computer science is seldom a linear process. Scientists often solved problems in a highly creative way, where intuitive thinking played a major role. However, this aspect is usually swapped for an emphasis on a more structured, and sadly more sterile, teaching delivery approach.

The ‘Calculus Reform’ movement between the 1980s and 1990s attempted to modify the pedagogical approach to focus on real-world problems (Smith 1994). Despite the fact that data science is integrated and embedded within the real world, there still appears to be a high level of resistance to a context-based teaching delivery. Traditional computer science courses focus on principles rather than on their meaning, neglecting the fact that contextualisation and real-world application can foster a deep-rooted motivation for a successful learning environment. STEM subjects are becoming increasingly important to our society, the whole economy and global well-being. Therefore, attracting and retaining students in STEM subjects is an important pedagogic consideration that needs to be encouraged.

Cooper and Cunningham (2010) propose a set of points to identify potential improvements in teaching delivery. These include:

- Using consistent and coherent examples from up-to-date and research/industry informed areas.
- Organising examples so that they can illustrate and introduce the core concepts and principles to augment the *meaning*.
- Ensuring the theoretical concepts are well linked with examples and applications so that they can provide a generalisation and, if appropriate, abstraction.
- Organising industry and/or external speakers to discuss how such examples and applications are relevant to their respective areas.

Introductory courses in data science can be easily adapted to capture the contextual approach described above. In fact, the use of robotics, programming in a supported learning platform, as well as augmented reality demonstrations and simulations should be relatively simple to implement due to the general nature of such courses. Examples include Alice to create animation and video games to explore introductory programming, whereas Lego Mind- storms enable the understanding of physical systems which can be controlled via simple GUIs. With a little bit of imagination, they can also provide some data for later analysis.

Contextualisation can also be successfully applied to more advanced data science modules. It is usually assumed that students attending such classes exhibit a higher level of motivation. Even though this may be the case, unmotivated groups do occur. Contextualisation has clear advantages in motivating as well as avoiding any decline in motivation. However, depending on the topic, such contextualisation has to focus on the relevant resources. For example, for theoretical computer science and data science modules, regular talks, and seminars on their applications, as well as introducing more sophisticated theory can provide a direct and effective contextualisation. In particular, multidisciplinary contextualisation, that is a well-designed to embrace multiple disciplines, reinforces the message that STEM disciplines are (and must be) connected and can flourish cross-collaborative environment are established.

7.4 Is Contextualisation Aligned with the Educational Framework?

The Technological Pedagogic Context Knowledge (TPCK) (Cooper and Cunningham 2010) is a framework aiming to integrate technology with suitable pedagogic approaches and technical content into the teaching delivery. One of the objectives of the TPCK is to ensure that not only is technology used in classrooms, but the most appropriate tools are identified and employed. For example, a particular IDE might be appropriate for advanced students but not for introductory classes. Furthermore, environmental factors (size of classes, their location, etc.), which might affect the deployment of such technology must be considered and weighed up.

Problem-based learning is a pedagogical approach, which is closely related to contextualisation, as it offers open-ended, student-driven problems to address the different learning outcomes of a course. Similarly, problem-based learning is aligned with cognitive constructivism, which argues that the learning experience is enhanced by relating educational concepts to real-world examples, which in turns leads to the conceptualisation of content (Duffy and Jonnasen 1992). By actively engaging with the content in an analytical manner, this approach can lead to improved critical-thinking abilities. Problem-based learning has been extensively used and implemented as part of the TPCK (Bonwell and Eison 1991). This further demonstrate the alignment of conceptualisation with the TPCK.

Contextualisation and problem-based learning have proved effective where ‘meaning-making’ is preferred over fact collection and assessment such as in biomedical disciplines (Barrows 1996). When decision-making and fast-thinking skills need to be developed and fostered, being able to semantically connect theory with action offers a particularly useful learning framework. It is important to emphasise that contextualisation is not just project-based learning. Project work in data science is a well-established practice, which has several and clearly observed advantages. However, solving specific problems based on the content knowledge might miss some important features of contextualisation. In particular, when project work is industry and business informed, it can be too narrowly focused. This is especially evident when such projects lack cohesion and a clear hierarchical structure spanning over different disciplines. This makes the process of curriculum development very complex, as it requires wide-reaching expertise (Duch et al. 2001) and balancing the various influences of the above factors. Furthermore, if such problems are not understandable to students, then this defeats the objective of a context based approach.

Considering that most scientific topics are essentially based on problem-solving, project-based working is considered a natural extension (Shackelford et al. 2006). However, this can also pose a challenge, as a wider contextualisation might not be that easy to implement in practice.

Another important aspect of contextualisation is the fact that it supports independent learning. In particular, it facilitates the discovery of state-of-the-art technologies within suitable problem-solving tools. In, data science curriculum interim

revision (Cassel et al. 2008, Lunt et al. 2008), a specific set of skills have been identified, namely a systems-level perspective, an understanding of the close connections between theory and practice, significant project experience, fostering of rigorous thinking, adaptability and analytical skills. Contextualisation clearly promotes and supports such features.

7.5 Applying Contextualisation at the Curricular Level

There are many observed instances where contextualisation is either applied to a single module or structured in a fragmented way. Despite this, any level of contextualisation can encourage students to work through small initial problems to gradually complex areas while increasing learning independence. Furthermore, this also leads to students' conscious ownership of their learning and a more agile problem-solving approach.

However, contextualisation is most effective when it spans across all the modules to ensure a consistent and meaningful learning experience. In fact, in order to take full advantage of contextualisation, students need to experience the process of discovering with possible solutions whilst assessing their suitability, across the whole curriculum. Furthermore, some students can find the perceived lack of formal structure inherent in PBL and lack of familiarity with the approach somewhat overwhelming (Kay et al. 2000) especially if contextualisation is not widely adopted across all the curriculum. Discontinuity in the pedagogic approach can result in student confusion over what is expected of them. Traditional teaching methods can be seen as less demanding of the students and hence be favoured by them over PBL until they learn to appreciate (if they do) the greater degree of freedom associated with PBL.

Since problem-solving is crucial to all academic fields, including especially computer science and data science, it is essential to ensure a consistent approach to contextualisation. This is also why contextualisation must span across all the teaching provision at departmental and/or institutional level. Moreover, it has been observed that problem-solving skills are better attained, consolidated, and enhanced over a prolonged period of time, preferably over the whole of the student's learning experience.

Contextualisation is credited with a positive impact on self-regulated learning. Time management, objective awareness, as well as knowledge sharing with their peers are all enhanced when students are offered opportunities to practice and subsequently strengthen the necessary skills of managing the student's own learning (Pintrich 1995). Self-regulated learning needs to be established incrementally to minimise students' frustration and anxiety while supporting their self-discovery, not only in introductory, but also during higher level courses (Kay et al. 2000).

Contextualisation is likely to require a significant commitment by the instructors. For example, mentoring students can be very time consuming. As students are encouraged to embrace contextualisation-based learning, as well as developing suitable solution strategies, additional resources and support need to be considered to

ensure a smooth instructor-student interaction, whilst optimising the learning experience. Student led learning does not result in fewer demands on instructor time and can often require more support for the student, particularly in the early stages. This additional workload needs to be recognised, appreciated, and supported at an institutional level.

As students progress through their chosen pathway, the nature of the contextualised problems, which are deployed, will need to increase in complexity (with a greater level of independence expected) aiming to enable students to independently resolve real-world problems.

7.6 Examples of Contextualisation

Depending on the educator's background, expertise and research interests, there might be various examples of contextualisation. More specifically, at the author's institution the main contextualisation themes focus on data science and industry 4.0. Both areas have been drawing considerable interest from the academic and industry communities, which has led to an increasing number of publicly available case studies and data repositories. Furthermore, due to the considerable number of interdisciplinary areas and topics that fall under the definition of industry 4.0, there is plenty of scope to include different disciplines, applications and theoretical frameworks. Some notable examples include sensor data from large manufacturing companies, which can be used to identify faults and production inconsistencies. Sensor data need to be assessed in real-time to minimise any potential disruption, and as such, specific AI and data science tools and techniques (mainly focusing on time series analysis) are introduced to address the main challenges posed by this type of analysis. Other examples include utilising the data collected by social media platforms. Data extracted from Twitter, Facebook, YouTube and other suitable social media are used to illustrate various data science techniques. Data patterns extracted from historical data sources can be used to identify opinions, topics and areas of interest. This has the added benefit of demonstrating the way social media platforms use data and how this data can be manipulated. Moreover, these data sources can also be utilised to identify potential bots spreading misinformation as well as potential cyber-attacks. Finally, general unstructured medical data (such as text and images) are used to identify comorbidities, treatments and specific diseases.

7.7 Digital Learning

The impact of the COVID-19 global pandemic has been (and still is) enormous, affecting social infrastructure and well-being, significant cultural disruption, as well as deep and long lasting economic issues. Furthermore, it is clear that it has mostly affected the vulnerable section of society (Van Lancker and Parolin 2020, Von Braun

et al. 2020). It has also had a negative impact, across all levels of the education system as the lack of face-to-face interactions has presented many challenges for learning, teaching and assessment. # Face-to-face learning delivers a qualitative higher value than online teaching (Paechter and Maier 2010). However, the general impact of COVID-19 at work and in educational settings has opened up new opportunities. Whilst several research studies have demonstrated the benefits of digital learning and teaching delivery across various areas and topics, the widespread digitalisation, utilisation and evaluation of such approaches is still being debated.

Scientific topics tend to be more suitable to be delivered in a digital format and approach. From a contextual point of view, data science has offered a massive input into the mitigation and assessment of COVID-19, from computational modelling, AI, machine learning as well as data science (Dwivedi et al. 2021). Remote teaching has evolved dramatically. From specific types of modules and settings, remote teaching has become the norm. The urgency to adapt has not yet been followed by pedagogical research and evaluation, but it has nevertheless created and demonstrated a clear pathway towards an increased digitalisation and more bespoke teaching delivery. Digital technology allows the identification and development of teaching material which are more flexible and portable. Advancement of the different teaching platforms has facilitated a more proactive interaction between students and instructors via chat, video, or audio communication.

In Crick et al. (2020) analysis of technology-based teaching delivery is discussed, showing that subsequently to the move to online teaching data science instructors felt they were much more prepared due to their overall knowledge of, and familiarity with, the relevant technologies, compared with those in other disciplines. Furthermore, they were much more likely to be supportive of online teaching. However, a major concern revolves around the nature of the different modules. For example, more collaborative or practical topics such as robotics and group projects can be more challenging and require more careful planning. A number of academics and practitioners argue that the efficiency and suitability of online teaching may be overestimated by educational institutions. This may be particularly relevant for the HE sector which has moved towards online teaching, with limited workforce development and assessment of effective online pedagogy. Another point to consider is the potential strain on an institution's educational digital infrastructure as well as the potential disparities among students who may not have access to specialised and sometimes expensive equipment or software.

Having said that, most of the challenges and opportunities created by COVID-19 in educational settings have shown that the recent digital and technological advance can and should be fully utilised in any curriculum. Virtual and augmented reality kits for example have become much more affordable and available. Perhaps unexpectedly the increased importance of technology in an educational context has created opportunities for promoting contextualisation as well as enhancing students' digital and data skills.

7.8 Virtual Reality in Data Science Education

As discussed above, virtual, and augmented reality (VR/AR) technologies have become much more affordable and accessible over the last decades. This technology has the potential to provide an engaging teaching environment and opportunities to develop new methods in the field of digital learning. VR/AR can be highly effective in providing an interactive learning environment, especially for STEM subjects.

As discussed in Pirker et al. (2020), learning outcomes within computer science and data science education can benefit from this sense of presence, despite being virtual. More specifically, VR/AR environments facilitate and support.

- Enhanced spatial knowledge representation;
- Tasks that would not be practical or even impossible, in the real world;
- Enhanced motivation and engagement;
- Improved contextualisation via simulated real-world scenarios;

VR and AR systems within learning settings are also referred to as immersive virtual educational environments. These have been demonstrated enhance student learning, by providing first-order experience, natural semantics, autonomy, and presence. These aspects all contribute to learning with virtual educational environments. The relationship between users and Avatars provides a new dimension to computer-assisted learning. However, such relationships can only be viewed in dependence on the different educational contexts, content, as well as their main objectives (Tassos and Natsis 2011).

Freina and Ott (2015), discuss aspects associated with VR/AR that are embedded in contextualisation such as access to scenarios not readily accessible in natural environments such as historical events, remote areas which cannot be reached or too dangerous. Ethically problematic areas can be successfully address via VR/AR, including surgery procedures, interaction with patients, etc. Bricken (1991) identifies further advantages such as the experiential nature of VR/AR, which mimics our natural ability to process information, share common experiences, with the potential to tailor the experience to users. Furthermore, despite long-standing challenges, such as technology usability, its cost, and users' reticence, the overall perception of VR/AR is changing as kits are becoming more available and accessible.

VR/AR within data science curricula offer both opportunities and challenges. It provides the opportunity for systematic investigation of hypothetical scenarios, exploring objectives, manipulating target audience, and examining weaknesses and potential mitigation actions. The limitations revolve around UI usability and the availability beyond STEM subjects. However, the state-of-the-art VR/AR technology partially mimics real peer-to-peer communication, teamwork, and problem-solving sharing, Thereby enhancing soft skills, as well as the experience of working on large projects as part of an interdisciplinary team are yet to be fully implemented.

Table 7.1 provides a concise overview of the main findings identified in (Pirker et al. 2020).

Table 7.1 The analysis and insights identified (Pirker et al. 2020)

| Data science and/or computer science concept | Learning objectives | VR/AR concept |
|--|---|--|
| Object oriented programming (OOP) | Users are allowed to design and deploy a programming language class, assess control flow in a program, and use object-oriented encapsulation mechanisms such as private members | Visualisation for understanding |
| Coding via programming languages | Users can modify parameters, variables, methods and function, as well as new object. Users can interactively solve complex tasks with loops and conditional statements via suitably designed scripting languages. Complex algorithms can be visualised and interacted with (e.g. Bubble Sort algorithm) to understand the different procedures. Specific non-intuitive concepts such as inheritance, polymorphism can be visualised to enable an enhanced cognitive understanding. Robotic components can be integrated and visualised with specific algorithmic components | playful concepts, engagement, joy, collaborative learning |
| Computational thinking | Logical loops and conditions can be contextualised and visualised in a modular way | Embodied activity, natural embodied thinking and cognition, critical thinking via physical actions |
| System development | Users are allowed to spatially organise learning concepts and share documents in virtual collaborative spaces | Spatial freedom, creative interactions, innovation |
| Security concepts | Cybersecurity concepts can be visualised and appreciated, such as firewall. White hat hacking techniques can be deployed easily and interactively. IT security terms can be visualised and contextualised | Playful interactions, educational entertainment, engaging students |
| Theoretical computer science | Theoretical algorithms, and their properties can be visualised and understood by interacting with the different states | Visualisation of theoretical algorithms, their complexity and properties via metaphorical learning |

(continued)

Table 7.1 (continued)

| Data science and/or computer science concept | Learning objectives | VR/AR concept |
|---|--|----------------|
| Innovation, invention skills, project- oriented working | Product-oriented tasks can be implemented and observed (including their mutual connections) in contextualised learning setting | VR/AR projects |

7.9 Conclusion

Data science has become increasingly prominent in the majority of qualitative and quantitative sciences and disciplines. As a consequence, data science needs to be embedded in the curricula from an early age. Data scientists are very much in demand and many businesses as well as academic institutions are promoting the need to train more data scientist to address the national and international shortage. However, there are many challenges which are due to the interdisciplinary nature of data science. These include selective academic backgrounds, high cognitive workload and the need to swiftly switch between theoretical and practical approaches. A new trend that is becoming significant is contextualisation, which aims to embed learning in various topics and disciplines. Due to the nature of data science, this approach is a natural choice, especially when complemented by VR/AR technologies. More evaluation and systematic research are needed. However, the initial observations and results suggest the potential of contextualisation, not only within the data science curriculum, but to wider set of programmes and modules.

References

Barnes DJ (2002) Teaching introductory java through LEGO MINDSTORMS models. In: SIGCSE 02 proceedings, Cincinnati Kentucky USA, ACM

Barrows HS (1996) Problem-based learning in medicine and beyond: a brief overview. In: Wilkerson L, Gijsselaers W (eds) Bringing problem-based learning to higher education: theory and practice. New directions for teaching and learning series. Jossey-Bass, San Francisco, pp 3–12

Bonwell C, Eison J (1991) Active learning: creating excitement in the classroom. ASHE-ERIC Higher Education Reports, Washington

Bricken M (1991) Virtual reality learning environments: potentials and challenges. ACM Siggraph Comput Graph 25(3):178–184

Cassel LNR et al (2008) Computer science curriculum 2008: an interim revision of CS 2001: 108

Cooper S, Cunningham S (2010) Teaching computer science in context. ACM Inroads 1(1):5–8

Crick T et al (2020) The impact of COVID-19 and “emergency remote teaching” on the UK computer science education community. In: Maguire J, Cutts Q (eds) UKICER ’20: United Kingdom and Ireland computing education research conference. ACM, Glasgow, pp 31–37

DeClue T (2009) A theory of attrition in computer science education which explores the effect of learning theory, gender, and context. Consortium Comput Sci Coll (CCSC). (Southwest Baptist University, Bolivar, Missouri USA)

- Duch B et al (2001) The power of problem-based learning: a practical “how to” for teaching undergraduate courses in any discipline. Stylus Publishing, Sterling
- Duffy T, Jonnasen D (1992) Constructivism and the technology of instruction: a conversation. Lawrence Erlbaum, Philadelphia
- Dwivedi YK et al (2021) Artificial Intelligence (AI): multidisciplinary perspectives on emerging challenges, opportunities, and agenda for research, practice and policy. *Int J Inf Manage* **57**:1–47
- Freina L, Ott M (2015) A literature review on immersive virtual reality in education: state of the art and perspectives. *Int Sci Conf Elearn Softw Edu* 1(133):10–1007
- Kay J et al (2000) Problem-based learning for foundation computer science courses. *Comput Sci Educ* 10(2):109–128
- Lewis A, Stoyanovich J (2022) Teaching responsible data science: charting new pedagogical territory. *Int J Artif Intell Edu* 32(3):783–807
- Lunt B et al (2008) Curriculum guidelines for undergraduate degree programs in information technology. Association for Computing Machinery, New York, p 139
- Paechter M, Maier B (2010) Online or face-to-face? Students’ experiences and preferences in e-learning. *Internet High Edu* 13(4):292–297
- Pintrich PR (1995) Understanding self-regulated learning. *New Dir Teach Learn* 1995:3–12. <https://doi.org/10.1002/tl.37219956304>
- Pirker J et al (2020) Virtual reality in computer science education: a systematic review. In: VRST ‘20: 26th ACM symposium on virtual reality software and technology; RJ Teather, Joslin C, Stuerzlinger W et al (2020) Virtual Event ACM: 1–8
- Shackelford R et al (2006) Computing curricula 2005: the overview report. In: SIGCSE ‘06: proceedings of the 37th SIGCSE technical symposium on computer science education, Houston Texas USA, ACM
- Smith D (1994) Trends in calculus reform. *MAA Notes*
- Tassos AM, Natsis A (2011) Educational virtual environments: a ten-year review of empirical research (1999–2009). *Comput Educ* 56(3):769–780
- Van Lancker W, Parolin Z (2020) COVID-19, school closures, and child poverty: a social crisis in the making. *Lancet Public Health* 5(5):243–244
- Von Braun J et al (2020) The moment to see the poor. *Science* 368(6488):214

Part II
Teaching Examples and Practice

Chapter 8

Using Graphics to Inspire Failing Students



Dave Collins

Abstract The chapter summarises recent challenges faced by teachers of first year undergraduate programming and their causes. It proceeds to describe a pragmatic means of addressing such problems through the provision of parallel second programming module provision. The approach provides a means of motivating weaker students, introducing remedial prerequisite knowledge and avoids sacrificing employability and other perceived goals of early *high-flyers* in the subject.

Keywords First programming languages · Motivating computer science students · The processing language · Failure in programming · Learning edge momentum theory · Novice programmers

8.1 Introduction

A range of factors have made the teaching of computer programming in UK universities ever more problematic in recent decades. General grade inflation at A Level (UK university entrance qualifications) has meant that the skill and experience level of entrants is difficult to predict. Changes made to both the nature and assessment of such school qualifications have led to greater selectivity within subjects making it difficult to rely upon adequate coverage of the individual topics within a discipline—for example, trigonometry or statistics within mathematics. The expansion of the university system and the concomitant competition between institutions has paradoxically often led to the direct or indirect (through the university application clearing system) reduction in entry qualifications. The same competition has led to league tables which prioritise student satisfaction and ‘added value’ which has resulted in strong pressures to improve retention and grant higher award levels.

Whilst the above factors have been of influence, computer science courses continue to experience problems with widespread first-year retention difficulties and low pass rates. These ultimately manifest as low numbers of upper class degree

D. Collins (✉)
University of Keele, Keele, UK
e-mail: daicollins@yahoo.co.uk; d.j.collins@keele.ac.uk

awards (the second worst across UK universities) and make CS the worst subject for students either leaving with no award or with a lower award than they had originally targeted (Woodfield 2014).

In attempts to deal with these problems, many universities have experimented with different programming languages and paradigms for their first-year courses. Although the jury is probably still out regarding the success of most such endeavours, it is worthy of note that a recent comprehensive survey of first year programming language adoption in the UK revealed that such experimentation (alternatives to Java and C) is more common in lower tariff (lower entry qualification requirement) universities where the need to consider alternatives is likely to be more acute (Murphy et al. 2017). In my institution, faced with very poor first year performance levels and consequent retention problems, we also sought a remedy. In common with similarly ranked universities, at least a third of our students were not significantly benefitting from their first-year programming experience based upon assessment of their end of year performance. Some studies suggest that the figure could be even higher, given a failure of first year university assessments to accurately measure student programming knowledge and ability (Ford and Venema 2010).

Naturally, we regarded this as a serious problem and engaged in an exhaustive review incorporating internal focus group interviews and extensive analysis of student performance statistics. The overall conclusion reached was that we were not attracting the calibre of students that would benefit from the reasonably mainstream pattern of programming skill development that was embedded in our CS curriculum. That much was evident, but there were considerable differences of opinion regarding an appropriate solution to the problem. The main obstacles were a certain degree of inertia, a desire to meet expectations of more able students and the ultimate goal of maximising employability skills.

Streaming students based upon their initial aptitude for programming was a possible route forward. However, this presented several challenges: could we identify such students? could we resource parallel streams?; could we guarantee the efficacy of an alternative approach for failing students? What we were sure of was that by the end of the first programming module (Let's call it CS1), we could identify the failing group and what the future might hold for them without some form of intervention.

Through the focus group discussions, we identified three major barriers to learning present in the failing section of the cohort: (i) Expectations and motivation, (ii) Prior knowledge and experience and (iii) The nature of the programming learning process. I will consider each of these in turn.

(i) Expectations and Motivation

For many students, the CS1 course was not what they were expecting to encounter in their CS degree. Their knowledge of the subject was heavily influenced by their experiences in school and the portrayal of the subject in popular culture. At the time, school experiences were extremely varied but most likely to be focussed on IT solution packages such as spreadsheets and 'databases'. The adoption of the new computing curriculum within schools has changed this somewhat, but the delivery of the curriculum appears to be extremely varied at

present. In popular culture, computing could be an exciting subject capable of addressing (or causing) almost any societal problem. Hacking, social networks, artificial intelligence, robotics and gaming were common themes expressed by students. In contrast, CS1 exercises tended to be mathematically focussed and rather dry. Many studies, for example (Jenkins 2001), suggest that only a minority of CS students are *intrinsically* motivated by the subject of computer programming, and it is difficult to see how the use of such exemplars would be likely to improve upon this situation.

(ii) Prior Knowledge and Experience

There was a time when most universities could rely upon a common level of skills and experience in their UG cohort. For example, a GCE O Level qualification in mathematics guaranteed a fundamental knowledge of trigonometry and calculus—but attained to varying levels. This was no longer the case and many of the exemplars used in CS1 contained concepts and terminology which were not universally shared by the cohort. The provision of a discrete mathematics module had been abandoned some time earlier when it became apparent that failing students on the CS1 module also tended to fail this compensatory module.

(iii) The nature of the Programming Learning Process

There are now a wide range of theories to account for the unusual distribution of performance produced by year one computer programming modules. Of these, the author finds the *learning edge momentum* theory to be the most compelling (Robins 2010). In this theory, success in acquiring one concept makes learning other closely linked concepts easier, and failure makes it harder. The tightly integrated nature of the concepts comprising a programming language, drives students towards extreme outcomes. Coupled with this, early failure to master concepts is de-motivational and further exacerbates the problem. Failing students typically disengage early in a programming module reporting that they were unable to grasp the introductory concepts. Failing early often means failing completely.

We needed a prompt and pragmatic solution to the above problems. The essential approach was to agree an alternative to CS2 (Programming II—Data Structures and Algorithms) for the failing section of the CS1 cohort that would attempt to both meet the broad learning objectives of CS2 and address the three barriers to learning described above. CS2 at that stage occurred in the second semester of the first year of studies. Institutional constraints required that such a module (an alternative to CS2), would also be available to some *successful* CS1 students and that completion of the new module might allow access to CS3 (Advanced Programming) in year two. CS1, CS2 and CS3 were all taught using the Java programming language. I trust that the reader appreciates the low probability of a satisfactory outcome!

The choice of language for the new module was astoundingly straightforward. We needed to build upon previous experience in Java and allow for possible progression to subsequent Java based modules (CS3). The only language that met these criteria and also offered some other pedagogic possibilities was the Processing language

developed at MIT by Chris Rea and Daniel Shiffman (Reas et al. 2007). Processing is a graphics oriented language designed for non-science students to create visual art. The language comes with a simple development environment (subsequently much improved but no more complex) and a large set of example programs (termed *sketches* in the Processing terminology). The language is essentially implemented as a Java library which entails that it could be imported into a conventional Java program within a conventional Java IDE. This afforded the possibility of returning to ‘mainstream’ Java toward the end of the new module without requiring any major conceptual leaps. By now, I am sure that many readers are familiar with the Processing language and environment so subsequent discussion will relate to how it was used to address our specific problems. An excellent exposition of the possibilities of the language is given in Daniel Shiffman’s book (Shiffman 2016).

8.2 The Learning Objectives of Our CS1 and CS2 Modules

Before progressing further, I present the aims and learning objectives of our CS1 and CS2 programming modules:

8.2.1 CS1

Aim: To introduce computer programming concepts using a generic (non-context specific) computer language and to develop problem-solving skills in the framework of computer programming.

And Objectives:

- Demonstrate an understanding of the basic concepts of computer programming
- Evaluate the suitability of computer language data and control structures to achieve basic problem-solving
- Demonstrate an understanding of the basic software engineering principles
- Show practical experience of those basic concepts.

8.2.2 CS2

Aim: To develop new programming skills as part of an exploration of several important data structures and algorithms used in Computer Science.

And Objectives:

- Write a program that demonstrates important features of computer programming using an object-oriented programming language;

- Describe, explain and evaluate the principles and operation of several data structures that are widely used in computer science;
- Use a programming language to operate, test and evaluate one or more of the widely used computer science data structures.
- Select class, data and control structures for program-based problem-solving.

In common with many other universities these aims and objectives are quite broad and allow implementation using a wide range of languages and, to a lesser extent, paradigms (object-oriented has a specific mention). The nature of *Processing* is such that it affords much the same opportunities as Java but the reduction in verbosity and bureaucracy coupled with graphics and animation support allows more creative problems to be addressed and expressed early in student's encounter with the language.

The challenge became one of designing a curriculum keeping the afore-mentioned three barriers to learning in mind. We sought to (i) attempt to provide a level playing field for students in terms of prior knowledge: (ii) motivate students by selection of exemplars that would stimulate interest: (iii) ensure that students mastered concepts incrementally, thus preventing early disengagement and failure. In the pages that follow we describe some of the main aspects of the module using tables to summarise how we addressed the identified barriers. The module progresses using exemplar programs, generally incomplete, as the context in which topics are taught and explored by students. During the practical sessions associated with each topic, completion of the associated individual exercises is confirmed and recorded by post-graduate demonstrators.

8.3 The Module

1. Drawing and Graphic Transformations

In order to introduce the *Processing* language and provide some revision of basic programming principles, the first topic provides an introduction to the procedural production of graphic images using primitive drawing shapes. Students are also introduced to graphical transforms and coordinate systems. Exercises include the reproduction of example images using repetitions of translations, rotations and scaling. Associated skills and concepts are illustrated in Table 8.1.

2. Animated Clock

The second major exemplar involves the animation of a real-time analog clock. Students are initially provided with a program demonstrating radial motion which also interactively depicts Pythagoras's theorem, trigonometric relationships and the expression of angles in degrees and in radians (the majority of our students being ignorant of the latter upon entry to the module). Some basic initial exercises are provided which are intended to give students with the opportunity to explore these

Table 8.1 Table of skills and concepts for Exercise 1

| | | |
|---------------------------------|-----------------------------|--|
| Additional knowledge and skills | CS programming concepts | Motivational material |
| Coordinate systems | Control structures | Students are asked to produce multiple flower and plant shapes with different morphologies and at different scales |
| Graphical transformations | Functions and decomposition | |
| Colour representation | Variables and scope | |
| Alpha transparency | Constants | |
| | Formal and actual arguments | |

Table 8.2 Table of skills and concepts for Exercise 2

| | | |
|-------------------------------------|-----------------------------------|---|
| Additional knowledge and skills | CS programming concepts | Motivational material |
| Trigonometry | Library function calls | The production of a functional program with visual design selected by the student |
| frame-based animation | Continuous versus discrete events | |
| Time zones and time representations | | |

concepts. As a final exercise, students are required to EFFICIENTLY simulate the action of an analog clock. Associated skills and concepts are illustrated in Table 8.2.

3. The Game of Life

For this example, we provide students with a simple but visually captivating version of Conway's game of life (Gardner 1970) is shown in Fig. 8.1. This provides a stimulating introduction to the concepts behind finite state machines. Exercises involve modification of survival and reproduction rules, changes to the grid size and resolution and modification of the 'neighbourhood' definition. The exercise provides the opportunity to master the use of 2/3 dimensional arrays and reasonably complex selection constructs. Students are expected to modify the code to permit different rules and neighbourhood definitions to be 'plugged in'. Associated skills and concepts are illustrated in Table 8.3.

8.4 Card Games

Students are provided with graphics for presentation of playing cards and are asked to design algorithms for dealing and shuffling. We present some sorting algorithms and Durstenfeld's algorithm for shuffling (Durstenfeld 1964). This provides an opportunity to introduce the topic of complexity and we discuss the performance of shuffling algorithms in this context. The card decks provide an excellent medium for exploring stacks and queues and associated ADTs. Associated skills and concepts are illustrated in Table 8.4.

Fig. 8.1 Conway’s game of life implemented in processing with fade-out effects for dying cells

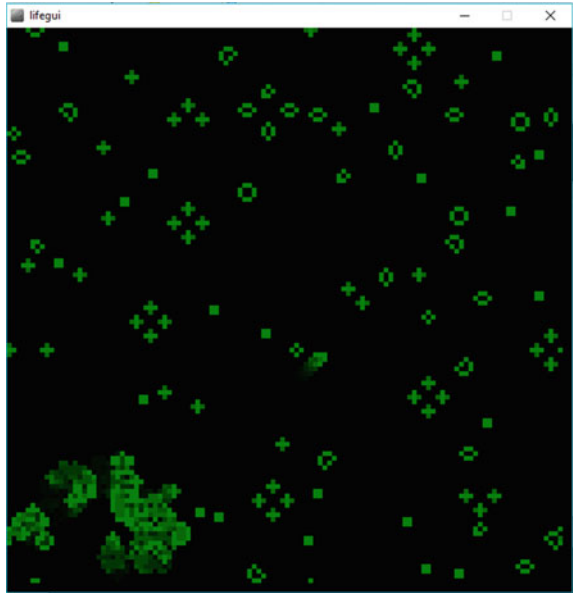


Table 8.3 Table of skills and concepts for Exercise 3

| Additional knowledge and skills | CS programming concepts | Motivational material |
|---|-------------------------------------|--|
| State machines | 2/3 dimensional arrays | Students are generally fascinated by the complex emergent behaviours that derive from simple rules |
| Turing, Von-Neumann and computing history | Cellular automata | |
| Modular arithmetic | Selection control constructs | |
| | Procedural abstraction | |
| | Tracing, reading and debugging code | |

Table 8.4 Table of skills and concepts for Exercise 4

| Additional knowledge and skills | CS programming concepts | Motivational material |
|---------------------------------|--|---|
| Numeric distributions | Sorting and shuffling algorithms | Card games require a range of algorithms and data types. The relevance and purpose is immediately apparent to students. Students are invited to devise their own shuffling algorithms initially which provides a natural vehicle for exploring complexity |
| Statistical concepts | Algorithmic and computational complexity | |
| Randomness | Arraylists | |
| | Random number implementations | |
| | Abstract data types, stacks and queues | |

Table 8.5 Table of skills and concepts for Exercise 5

| Additional knowledge and skills | CS programming concepts | Motivational material |
|---------------------------------|-------------------------|---|
| Seamless textures | File handling | Weaker students in particular are delighted to be able to produce a program that approaches the quality of a published (but retro) game |
| Easing in animation | Object arraylists | |
| Frame buffering | Asset management | |
| Multiple animation timelines | | |

8.5 Sprite Based Animation

Animation in Processing is essentially frame-based. In this section of the module we introduce the use of sprite sequences on scrolling backgrounds. Sprites may be controlled via the keyboard or with the use of a gaming input device. Associated skills and concepts are illustrated in Table 8.5.

8.6 Lunar Lander

Students are provided with a background lunar image and a transparent GIF representing the landing craft. They are also provided with an example program depicting objects falling under the influence of gravity that uses a skeleton implementation of a Vector class. They are then expected to complete the Lunar Lander Program using the keyboard to deliver 2 dimensional thrust to counter gravity and gracefully land the craft at a randomly selected location. This requires implementation of new methods for the Vector class. Associated skills and concepts are illustrated in Table 8.6.

Table 8.6 Table of skills and concepts for Exercise 6

| Additional knowledge and skills | CS programming concepts | Motivational material |
|---------------------------------|-------------------------------------|--|
| Forces and Newtonian physics | Classes and objects | A playable game with realistic control |
| Vectors | Keyboard polling/state detection | |
| Discretization | A vector ADT and its implementation | |
| | | |

Table 8.7 Table of skills and concepts for Exercise 7

| | | |
|--|-------------------------|--|
| Additional knowledge and skills | CS programming concepts | Motivational material |
| Matrices | Convolution filters | The real-world challenge present in the exercise task is highly motivational to students and encourages them to master a topic that often produces despair when presented in a more traditional image processing context |
| Image representation and Compression schemes | Matrix manipulation | |
| Bitmap versus scalar graphics | | |
| Normalisation | | |

8.7 Image Processing

Students receive basic lectures on Image formats, manipulation of pixel information and the construction of filters using convolution matrices.

They are provided with a scene of crime image in which a poorly illuminated suspect is seen next to a car with a number plate which is indiscernible. The final task requires the construction and application of filters (incrementally) that will clean-up the image to the extent that the suspect is identifiable and the number plate readable. Associated skills and concepts are illustrated in Table 8.7.

8.8 Collision Detection

Students are presented with a complete program (based upon the game of Asteroids) that visually represents coarse and fine collision detection and their affect upon maximum animation frame-rates. Students are required to assess the relative performance of coarse and fine approaches and derive a strategy for optimising the compromise between accuracy and performance. Associated skills and concepts are illustrated in Table 8.8.

Table 8.8 Table of skills and concepts for Exercise 8

| | | |
|---------------------------------|--------------------------------|---|
| Additional knowledge and skills | CS programming concepts | Motivational material |
| Further trigonometry | Collision detection algorithms | Students tend to be familiar with this problem and are inspired to find strategies that produce optimal game-play |
| | Heuristic algorithms | |

Table 8.9 Table of skills and concepts for Exercise 9

| Additional knowledge and skills | CS programming concepts | Motivational material |
|---------------------------------|--------------------------|--|
| Numeric distributions | OO Design | Students are fascinated by Boid behaviour and in particular with the complexities of behaviour that can result from the application of such simple rules |
| | Complexity | |
| | Code optimisation | |
| | Algorithmic optimisation | |

8.9 Boids

This is by far the most complex programming example provided to students—an object-oriented version of Craig Reynold’s (Reynolds 1987) Boid simulation written in the *Processing* language. Lectures introduce the concepts involved and describe the overall design, the main classes and broad implementation details.

The exercises require students to read and follow the logic of this relatively complex piece of code. For example, they are instructed to alter the weighting applied to the three vectors that determine the speed and acceleration of Boids at each discrete decision cycle (which corresponds to the frame-rate). In order to provide more realistic behaviour, they are encouraged to experiment with allocation of weights using a Gaussian distribution. They are expected to modify the definition of neighbourhoods and, as a final exercise, to consider means of improving performance of the algorithm to provide real-time flocking utilising more sophisticated Boid animations. Specifically, they are asked to consider a means of avoiding the need to check the position of each Boid in each cycle in order to determine whether it might be considered a neighbour for purposes of calculation of the required alignment vector. Associated skills and concepts are illustrated in Table 8.9.

8.10 Recursion

Recursion can be a difficult subject for students and benefits greatly from a graphical treatment. Students are provided with a range of examples including binary trees, Sierpinski triangles, Koch snowflakes and Mandelbrot sets. A range of exercises provide experience in using linear, binary and tail recursion. The final (non-trivial) exercise requires that students add realism to the construction of a binary tree. Associated skills and concepts are illustrated in Table 8.10.

Table 8.10 Table of skills and concepts for Exercise 10

| | | |
|---------------------------------|-------------------------------|---|
| Additional knowledge and skills | CS programming concepts | Motivational material |
| Fractals | Recursion and recursion types | Recursion is explored through graphics—binary trees, Sierpinski triangles and fractals. Students are highly motivated by the task of introducing realism into the construction of a graphical binary tree |
| Complex numbers | Recursion performance | |
| | Binary search trees | |

Table 8.11 Table of skills and concepts for Exercise 11

| | | |
|-----------------------------------|-------------------------|--|
| Additional knowledge and skills | CS programming concepts | Motivational material |
| Comparative programming Languages | Programming frameworks | At this stage many students have discovered the relationship between Java and processing. This topic formalises that understanding |
| | APIs | |
| | Wrapper classes | |

8.11 The Relationship Between Processing and Java (and Python, Javascript and Android).

The final topic of the module provides students with the opportunity to use *Processing* as a Java library within the NetBeans IDE. Lectures describe the language implementation and the manner in which it can be incorporated in a conventional Java application. Processing implementations for Python, Javascript and Android are also described and demonstrated. Associated skills and concepts are illustrated in Table 8.11.

The above list of exemplar-led topics is not exhaustive and we do change the examples used with each delivery of the module. The module has now been running for 8 years and we recognise that some of the examples are becoming stale and would benefit from new materials. We are also aware that the materials may have a gender bias that we would like to address without falling prey to obvious stereotypes.

8.12 Module Assessment

The module is assessed by examination and coursework. During the fourth week of the module students are required to submit a proposal for their assignment work (with the benefit of a list of future topics that will be covered). We point them at the more complex Processing exemplars provided with the environment and at the

OpenProcessing.org website (www.openprocessing.org) for inspiration. The latter provides a virtual gallery through which students may showcase their work to the world. Each student is briefly interviewed regarding their choice of subject and is only allowed to proceed if the lecturer considers the project to be suitably challenging and capable of expressing the module's learning objectives. The progress of each student on their assignment work is monitored throughout the remaining practical sessions.

8.13 Discussion

There are many variables that determine the success and failure of a module. The introduction of this module was not a controlled experiment and it would not be appropriate to make comparisons or draw conclusions as if this were the case. However, we can broadly point to some of the benefits and problems.

Retention problems were dramatically reduced and weaker students were undoubtedly inspired by the new approach. Much of the student work produced in coursework assignments far exceeded our expectations and continues to do so to the present day. Stronger students had sufficient new materials as to be challenged by the approach but there were some criticisms such as the absence of access modifiers and the simplification of type casting methods in the Processing language. In practice, some of the more capable students possibly felt patronised by the introduction of what they perceived to be a simpler programming environment. Indeed, many were quick to import the *Processing* core into a Net Beans or Eclipse Java environment. Figure 8.2 shows the main screen of a successful game produced by a first year student.

Processing has a pre-processor that wraps the processing code inside a Java class and allows access to functions without having to declare and instantiate the classes to which they belong. Combined with rich graphical features this allows the student to focus on algorithmic content. In reality, the language *is* a slightly more accessible form of Java coupled with a simple IDE and a feature rich Graphical API. Changes to the intrinsic Java are generally akin to being able to introduce `println()` rather than explain `System.out.println()`. The latter requires suspension of full understanding in weaker students and partially explains why the language creates a *learning-edge* problem with failure to understand one concept *at a specific time* leading to lack of confidence and unpreparedness to tackle subsequent related concepts.

The module has undoubtedly led many failing students from CS1 to develop an *intrinsic* motivation to further develop their programming skills. Students exhibit pride in their work and a desire to demonstrate to, and seek the opinions of their peers. The current module lecturer has persuaded a major European company to sponsor an annual award for the best student assignment work. The module's introduction was a pragmatic solution to a problem and saved us from having to meddle with the now long standing and conventional approach to teaching programming with the Java language that is embodied in our CS1-CS2-CS3 programming strand. Since

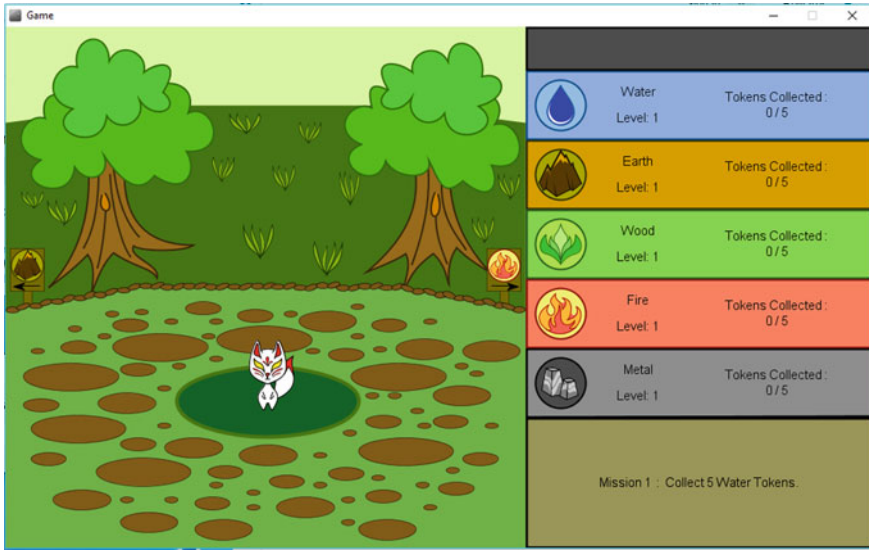


Fig. 8.2 A game produced by a first year student

that time, many UK universities have adopted *Processing* as a first programming language. Interestingly, Murphy et al.’s recent survey of language use concluded:

“The results in this first UK survey indicate a dominance of Java at a time when universities are still generally teaching students who are new to programming (and computer science), despite the fact that Python is perceived, by the same respondents, to be both easier to teach as well as to learn” Murphy et al. (2017).

Of course, the likely explanation for the above is that the respondents have goals other than ‘teachability’ in mind when selecting a first language—employability would undoubtedly feature highly.

From my perspective as a teacher of programming, the module undoubtedly extended my ‘shelf life’. Instead of trying to avoid the weary gaze of yawning students as they tested stack implementations full of meaningless characters and numbers, I was able to witness genuine excitement in students as they retrieved cards from a discard pile or added visual leaf nodes to a binary tree. Early on, it was apparent that some students from CS1 had not really understood two-dimensional arrays. Conway’s life provided both motivation and visual feedback that allowed weaker students to both perceive their relevance and master the topic. Students with little mathematical or scientific background picked up an understanding of trigonometry, Newton’s laws, parabolic curves and frequency distributions pretty much as a side-effect of wanting to make a game more playable or realistic. Above all, students became far more enthusiastic and attended lectures and practical sessions with relish rather than a sense of guilt or obligation.

Even better was the fact that no compromises were being made in the teaching process. I was still teaching Java to attain the same learning objectives in the context

of Computer Science and Software Engineering as were adopted by our CS2 module. Effectively I was provided with an additional teaching tool with which I could create teaching materials that were stimulating yet understandable. The overheads involved in using Processing rather than pure Java were negligible and ultimately led students to understand how they might design an effective application framework. I am indebted to the Processing team for their generous contribution to computing pedagogy.

References

- Durstenfeld R (1964) Algorithm 235: random permutation. *Commun ACM* 7(7):420
- Ford M, Venema S (2010) Assessing the success of an introductory programming course. *J Inf Technol Edu* 9:133–145
- Gardner M (1970) Mathematical games—the fantastic combinations of John Conway’s new solitaire game “life”. *Sci Am* 223:120–123. ISBN 0-89454-001-7
- Jenkins T (2001) The motivation of students of programming. In: *Proceedings of the 6th annual conference on innovation and technology in computer science education 2001 Jun 25*, pp 53–56
- Murphy E, Crick T, Davenport JH (2017) An analysis of introductory programming courses at UK universities. *Art Sci Eng Program* 1(2) (Article 18)
- Reas C, Fry B, Maeda J (2007) *Processing: a programming handbook for visual designers and artists*, 1st edn. The MIT Press, p 736. ISBN: 0-262-18262-9
- Reynolds CW (1987) Flocks, herds, and schools: a distributed behavioral model. *Comput Graph* 21(4):25–34
- Robins A (2010) Learning edge momentum: a new account of outcomes in CS1. *Comput Sci Educ* 20(1):37–71. <https://doi.org/10.1080/08993401003612167>
- Shiffman D (2016) *Learning processing—a beginner’s guide to programming images, animation, and interaction*. Morgan Kaufmann. ISBN: 978-0-12-394443-6
- Woodfield R (2014) Undergraduate retention and attainment across the disciplines [Internet]. Higher Education Academy, York. Available from www.heacademy.ac.uk/node/10293. Accessed 17 June 2018

Chapter 9

Best Practices for Teaching Information Systems Modelling



Steve Wade

Abstract The subject of Information Systems Modelling (ISM) grew out of computer science to fill a gap created by the difficulties programmers had in understanding and solving user problems. The intention behind ISM is to facilitate communication between technologists (many of whom have no idea of the complexity of organisations) and end-users and their managers (many of whom are unable to translate their problems into feasible demands upon technology). “Best practices” in information system development might therefore be considered to be those practices which contribute in some way to improving communication between these two parties. The work described here is primarily focussed on documenting practices that address the issues associated with the seamless transition from a requirements model seamlessly to a technology based system that satisfies those requirements. This has involved reflection on lessons learned during thirty years’ experience of teaching Information Systems Modelling in the context of higher education.

Keywords Information systems modelling · Requirements model · Communication · Pattern language

9.1 Background

The subject of Information Systems Modelling (ISM) grew out of computer science to fill a gap created by the difficulties programmers had in understanding and solving user problems. The intention behind ISM is to facilitate communication between technologists (many of whom have no idea of the complexity of organisations) and end-users and their managers (many of whom are unable to translate their problems into feasible demands upon technology). “Best practices” in information system development might therefore be considered to be those practices which contribute in some way to improving communication between these two parties.

S. Wade (✉)

Department of Computer Science, University of Huddersfield, Huddersfield, UK

e-mail: s.j.wade@hud.ac.uk

Before considering examples of best practice and how these may be taught we will consider the possible consequences of poor communication between technologists and end-users. Focussing on these will enable us to be clearer about the specific benefits that the practices we consider might offer. We will focus on two possible consequences of poor communication.

1. The end-user holds limited views about what they need and fails to realise that alternative superior solutions (which they have been unable to imagine) are possible.
2. The developers hold misguided opinions about what the users need because this has not been clearly explained to them.

There is good evidence to suggest that many information system failures are a result of one or both of these leading to a discrepancy between the system “as required” and the system “as delivered”. If we are to avoid this discrepancy we need to deploy development methods that:

1. Provide mechanisms to make sense of and understand the details of human activities that an information system is developed to support. This involves developing some kind of information requirements model.
2. Provide a seamless transition between developing the information requirements model and the design and implementation of a technology-based system to satisfy the requirements captured in the model.

The work described here is primarily focussed on documenting practices that address the above requirements. This has involved reflection on lessons learned during thirty years’ experience of teaching Information Systems Modelling in the context of higher education. Some lessons were learned in the eighties when we still made use of plastic flowchart stencils, pencils and plenty of printer paper. More came in the nineties with the rise of powerful CASE tools supporting development methods that required the designer to develop and maintain increasingly elaborate, internally-consistent collections of diagrams. The most significant lessons have been learned more recently as the evolution of powerful programming environments has encouraged a more informal approach to modelling.

In addition to learning lessons from the past it is important to prepare students for the future. Most software systems are embedded in social systems and the resulting sociotechnical systems’ boundaries and interactions can be hard to identify. For example, social networks, travel booking and online shopping applications have had far-reaching effects on the way people form relationships, how they travel and what they buy. Becker et al. (2016) have argued that software’s critical role in society demands a paradigm shift in the software engineering mind-set. We argue that our students need to be prepared for this shift which will focus on architectural issues that can be addressed through Information Systems Modelling.

9.2 Introduction

This chapter primarily draws on a number of years' experience teaching an Information Systems Modelling module jointly to postgraduate students on an MSc Information Systems Management and an MSc Advanced Computer Science. The module concerns the application of the Unified Modelling Language (UML) throughout the development lifecycle from requirements analysis to implementation. All the students arrive on the module with some background in modelling but those on the MSc Information Systems Management tend to think in terms of business models whereas those on MSc Advanced Computer Science tend to view modelling as high-level programming. This presents the challenge of moving students into a deeper understanding from different starting points and with different preconceptions about the nature of the subject.

In the process of delivering this module we have engaged in the following activities each of which will be described in more detail in the remaining sections of this chapter:

- The design of a pattern language to organise best practice in the application of systems development techniques. In developing the pattern language we were mindful of the need to encourage maximum student ownership of the development process. The patterns could not therefore comprise simple lists of instructions to be followed slavishly.
- The development of teaching materials to document the pattern language.
- Running the module. Observing the progress of the module week-by-week in a number of ways including a range of on-going student feedback mechanisms.

The remaining sections of this chapter reflect on what we have learned from engagement in these activities.

9.3 The Pattern Language

Patterns have been widely used in information systems design over the last ten years. A pattern in this context is a generic solution to a recurring problem expressed in a literary form. The approach has its roots in architecture specifically the work of Alexander (1979). In ISM patterns have been used to ease communication problems and the thinking behind complex design (Gamma et al. 1995). Patterns are usually described by templates which specify the style and structure of a pattern description. Typically the template will include sections for a description of the problem to be addressed, the forces acting to create the problem, a generic solution, a specific example of how this solution might be applied and a discussion of the benefits the solution should provide.

The following example relates to a common (and hopefully familiar) problem in domain modelling where students represent a many-to-many relationship between two objects when the relationship would be better represented by a third object.

Problem

How to model the relationship between two classes that have a many-to-many association with each other.

Forces

- Many-to-many relationships occur often in the real world.
- It can be difficult to implement many-to-many associations in some object oriented programming languages.
- Many-to-many relationships have no direct implementation in relational database systems.
- A many-to-many relationship is usually complicated enough to warrant the addition of an extra class.

Solution

Transform the many-to-many association between two classes into a trio of classes by creating an intermediary class with two one-to-many relationships. The name of the intermediary class should describe the type of relationship being captured.

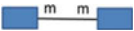

Example

A many-to-many relationship between Student and Module is reconstructed as two one-to-many relationships. One between Student and Work Record and the other between Module and Work Record.

Discussion

We can now store details of module grades for each student as attributes of the new “work record” class.

Summary

If you find this:  consider replacing it with this: 

The idea is that patterns such as this can be used to guide students away from common problems and into good practice. We have specified many more patterns related to commonly occurring problems. Initially we used patterns drawn from the publications of Ambler (1998), and Evitts (2000). We spent some time re-working and shaping the documentation for these patterns to give coherence to the collection. A key feature of the collection is that relationships are drawn between patterns. When a number of patterns are related to each other in this way we describe the result as a “pattern language”. We are therefore trying to develop a pattern language to support

the teaching of information systems modelling. Further examples of specific patterns and their relationships will be provided later in this chapter.

9.4 A Framework for the Pattern Language

Most modern courses in Information Systems Modelling are based on the Unified Modelling Language (UML). The UML provides a suite of diagrams that help us to visualise the design of a system. It is published by the International Organization for Standardization (ISO) as an approved ISO standard. Having decided to use the UML we needed to decide which development method to follow. The most popular model-centric approach currently in use is the Unified Software Development Process (USDP) but this is both large and complex. Instead of following the USDP we devised our own simplified method based on our earlier research into the design of a multi-method framework (Salahat and Wade 2009). In that research we proposed a framework for bringing together principles from object oriented approaches to designing software systems and the “soft systems” approach to analysing social systems as part of Business Analysis (Checkland 1999). This approach requires a few words of explanation.

The teaching of Information Systems Modelling tends to focus on issues related to ‘hard’ systems design. Hard systems are the technical systems that are produced during a development project. Each hard system will be embedded in its social context. This context can be seen as a “soft” densely interconnected system of human activities. It can be argued that hard systems should not be analysed in isolation from the soft systems within which they reside. In analysing the present system or designing a new system, there is the need to consider both the hard system that will be the product of the development, and the soft system within which it will be used. This is challenging because the workings of the soft system are often difficult to understand and the needs of the organisation can be difficult to predict. The UML covers all aspects of hard systems design but has much less to say about the soft system. In contrast Soft Systems Methodology (SSM) focusses on the soft approach.

In addition to augmenting UML modelling with techniques from SSM we also wanted to introduce students to Persona Analysis as a means of developing empathy for users.

Defining personas is an established practice in user-interface design. Blomkvist (2002) describes personas as follows:

A persona is a model of a user that focuses on the individual’s goals when using an artefact. The model has a specific purpose as a tool for software and product design. The persona model resembles classical user profiles, but with some important distinctions. It is an archetypical representation of real or potential users. It’s not a description of a real, single user or an average user. The persona represents patterns of users’ behaviour, goals and motives, compiled in a fictional description of a single individual. It also contains made-up personal details, in order to make the persona more ‘tangible and alive’ for the development team. (Blomkvist 2002).

We ask students to develop personas that include a fictional name and life story, a picture, and a ‘tag line’—a phrase, supposedly written by the persona, that represents the character of the persona as related to the development project. The case studies that we use in teaching relate to our own department. Accordingly we encourage students to develop a persona for each of the following: A Student of Computer Science, a Student of Information Systems, a Course Administrator, a Lecturer and a representative of an organisation providing industrial placement opportunities.

Although the primary application of personas has been in the context of user-interface design, we have found spending time developing them focusses attention on requirements in a concrete way. Rather than referring to users in an abstract form, students refer to personas by name. So our student of Computer Science becomes Jo Smith who has a first degree in Software Engineering a great deal of confidence in his programming ability but lacks confidence in writing essays and reports—his tag line is “I would rather write code than prose”. In contrast our student of Information Systems becomes Sue Rachel who has a first degree in Law, is fascinated by the impact of technology on society but lacks confidence in her ability to write code. Her tag line is: “technology will never replace great people but it can help ordinary people to achieve great things”.

Following the basic structure of this framework we developed patterns and teaching materials based around the following topics:

- How to use Persona Analysis to help the developer focus on the needs of the user.
- How to use Soft Systems Methodology to learn about a problem situation.
- How to extract Use Cases from the soft systems models.
- How to develop sequence diagrams related to each Use Case.
- How to develop a domain model from the collection of sequence diagrams.
- How to convert the domain model into a class diagram and database design.
- How to implement the class diagram as an object oriented software system using the, “naked objects”, implementation pattern.

We have used the step-by-step approach implied by these questions as the basis for a course structure and an assignment specification. We have adopted a “scaffolded” approach to teaching (Larkin 2001) which involves working through a number of exercises following the structure implied above then asking students to apply the techniques to related case studies for their coursework. The case studies used for assessment were based around the needs of an academic department like our own.

It is beyond the scope of this chapter to discuss each of these topics in detail but for those unfamiliar with the techniques, the following examples are intended to give an idea of what deliverables are produced during each step of the method.

The example used here relates to the decision to introduce a Peer Tutoring System into an academic department to provide extra help to students on a programming module. The idea being that students who are confident in their programming skills would run support sessions for their less confident colleagues. We initially asked students to develop a simple persona for the type of people who would use this system. As explained above a persona is a fictional character that typically has a name, a picture, behavioural traits, common tasks, and a goal that describes the problem the

persona wants to see solved or the benefit the character wants to achieve. Personas are not considered in the UML so we devised our own simple template based on the facets listed above. In filling out this template the developer is encouraged to visualise the user in a concrete, tangible way. So the personas mentioned above, named Jo Smith and Sue Rachel, may be involved in the peer tutor system as a peer tutor and peer tutee respectively.

Moving on from Persona Analysis we next ask students to conduct a simple business analysis—with no attention to the design of software. Again this is somewhat outside the scope of the UML so, as explained above, we have introduced techniques from Soft Systems Methodology. The first of these is a rich picture. Figure 9.1 shows a rich picture that we might use to get things started.

We have found rich pictures to be a good way to encourage discussion about the problem situation without focus on any proposed solution. The discussion leads to the development of a root definition: This is defined in SSM as a succinct description of the Human Activity System (HAS) that is required. It is possible to develop multiple root definitions each offering a different perspective on what is required. The following is a possible root definition for the peer tutoring system:

A system owned by the course that provides programming skills support to students using volunteers with programming experience from the student cohort. The quality of this support will be monitored by academic staff.

Once we have developed a root definition, or set of definitions, we move to developing more detailed activity models. These are called Conceptual Models in SSM and we would develop one for each root definition. The following example, originally presented in Wade et al. (2012), includes activities that might be supported by software and others that will be enacted by humans without the assistance of software.

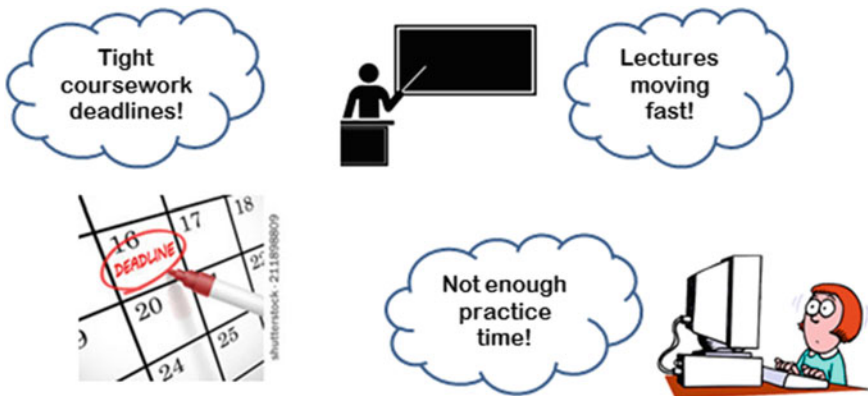


Fig. 9.1 Initial rich picture raising issues for the peer tutoring system

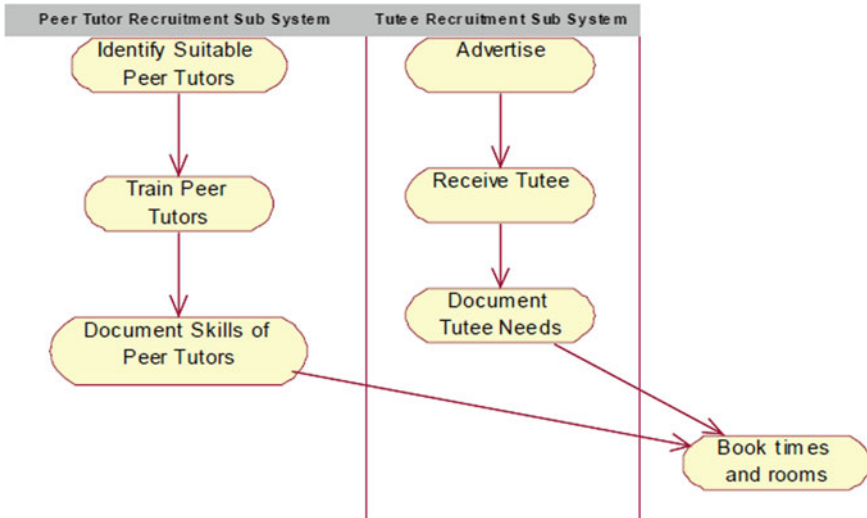


Fig. 9.2 Activity diagram

In developing this type of diagram (Fig. 9.2) we encourage discussion about the social system we are supporting. In this simple case discussion might focus on the following questions:

- Will weaker students attend these sessions or will they be primarily attractive to students who are already competent programmers but want further opportunities to develop their skills?
- Should some (weaker) students be required to attend the sessions? If so how do we identify people in this position?
- Should we pay peer tutors? Are there alternatives to rewarding them with money?

This discussion might lead us to develop additional activity models. For example we might consider the need to monitor attendance at the sessions and develop the following (see Fig. 9.3) for an “attendance monitoring” system:

In developing these models we have not concerned ourselves with the question of how software might be able to assist the people involved. This would be the role of a Use Case Model. Use Cases are part of the UML and represent activities that require software support. If we were to develop a Use Case Diagram from our activity models we would be making the transition from business analysis to software design. The Use Case diagram in Fig. 9.4 could be derived from the conceptual model above:

If we focus on the “Print Class List” Use Case we might prototype a simple user interface like Fig. 9.5:

Behind this interface our software system might be composed of collaborating objects. The high-level sequence diagram in Fig. 9.6 depicts the role that a number of objects might have, behind the scenes”.

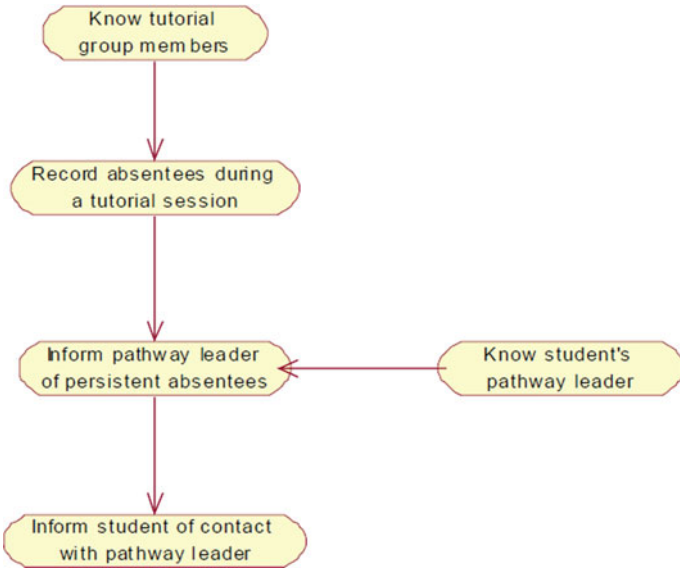


Fig. 9.3 Activity diagram for attendance monitoring. Taken from Wade et al. (2012)

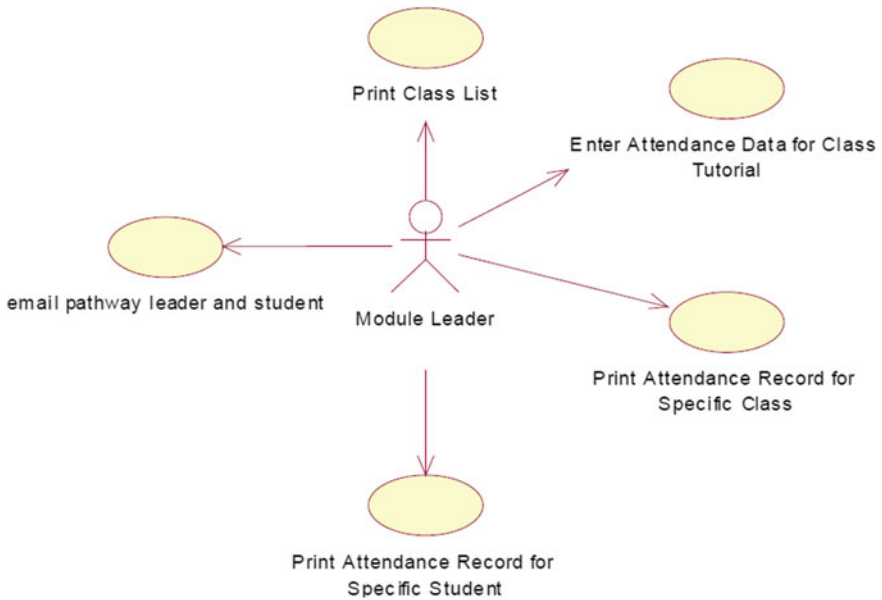


Fig. 9.4 A Use case model. Taken from Wade et al. (2012)

Fig. 9.5 Screenshot for a use case. Taken from Wade et al. (2012)

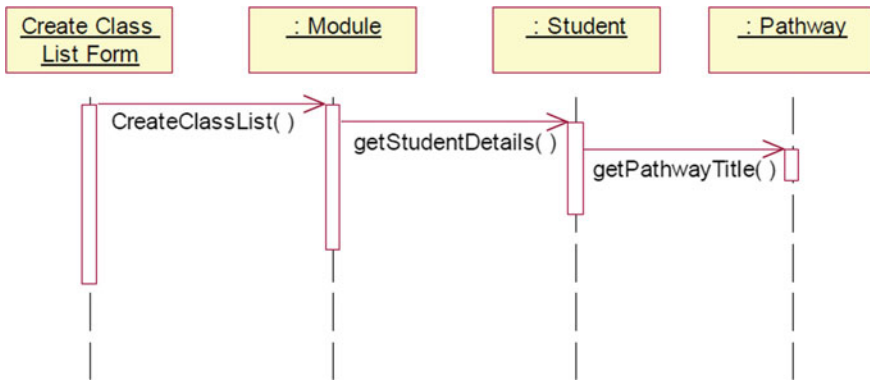
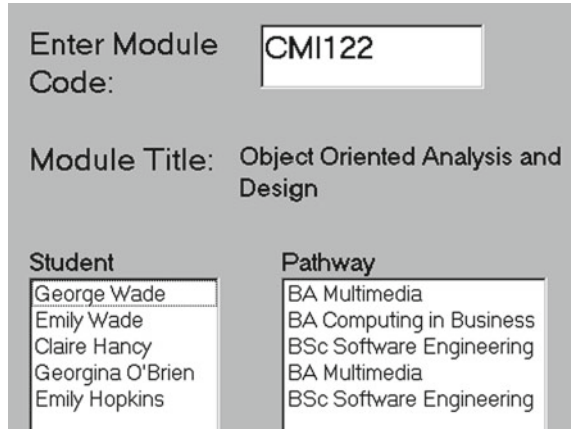


Fig. 9.6 A sequence diagram. Taken from Wade et al. (2012)

We found that students found the transition from Use Case Models to Sequence Diagrams difficult so we provided the following pattern and discussed it in class.

Problem

It is hard to develop sequence diagrams from the Use Case Module. What can I do to make this transition easier?

Forces

A high level Use Case Diagram (such as the one presented above) is fine for a, “mile high”, view of the computer systems behaviour. For many stakeholders, such as sponsors and managers, this will be enough. As designers however we need to open these up and define them in detail. We know what the system presents to the various users (or actors), we need to define in fine detail the, “how”, of that interaction; until we have done this we cannot begin to develop a sequence diagram. There is no

prescription in UML regarding what detailed information should be recorded about a use case.

Solution

Document the detailed logic of a Use Case as a series of steps. Where appropriate each step should include reference to one or more domain classes and identify the role that this class should play in the implementation of the Use Case. We can use this description as the basis for developing an initial sequence diagram. The way in which this is done is specified in the “Develop Sequence Diagram from Primary Path” pattern.

Example

This Use Case is concerned with enrolling an existing student in a peer tutor session for which she is eligible.

Key Steps:

1. The use case begins when a student wants to enrol in a peer-tutor session.
2. The student inputs her name and student number into the system.
3. The system verifies the student is eligible to enrol in classes at the university.
4. The system displays the list of available peer tutor sessions.
5. The student indicates the session in which she wishes to enrol.
6. The system checks that the student is enrolled on the appropriate module to join the session.
7. The system asks the student to confirm that she wants to enrol in the session.
8. The student indicates she wants to enrol in the session.
9. The system creates an enrolment the student in the session.

These steps can then be mapped to messages passed between objects in a sequence diagram like the one presented above. It may be that each line in the Use Case description would map to a single message passed between objects. We would develop a sequence diagram for every use case then develop a domain model consistent with all of these sequence diagrams. A domain model derived from this single sequence diagram might look like that in Fig. 9.7.

This domain model can be used as the basis for an object oriented software system design and a relational database structure. We have developed patterns to translate the domain model to a physical database design principally by adding primary and foreign keys to create relationships between tables. A separate pattern discusses ways of mapping inheritance relationships to relational structures.

In developing User Interfaces we encourage students to use the Naked Objects architectural pattern (2012) to generate a graphic user interface directly from the domain model. The pattern uses reflection to automatically generate an initial user interface. Typically this interface will present a series of windows containing icons representing each of the domain classes. A class can then be accessed by double-clicking on its icon to reveal its operations. In the above example I can select, “Module,” select a specific module then right-click on that module’s “Create Class List” operation to see a list of students currently enrolled on the module. Other

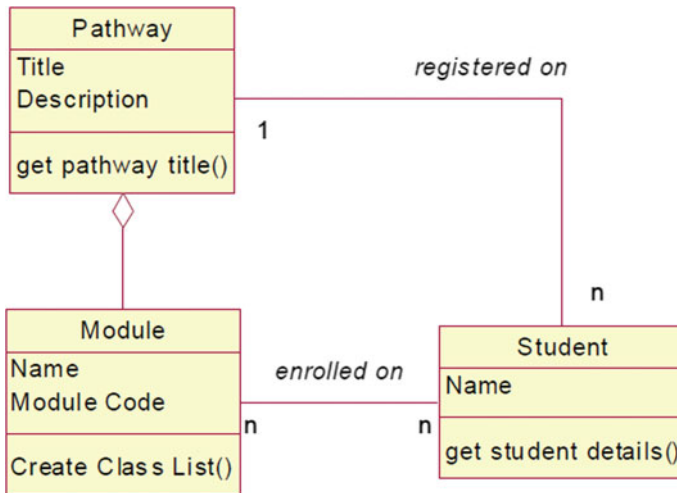


Fig. 9.7 A Domain model. Taken from Wade et al. (2012)

functionality can be achieved by dragging and dropping; so for example if I wish to enrol a student onto a module I can drag the icon representing that student on to the icon representing the module thereby creating the relationship between them. An advantage of applying this pattern is that the resulting relationship between the domain model and what appears in the interface is very direct. A change in the domain model (e.g., the addition of an operation on, “Student”, named “Get Coursework Marks”) feeds through into the code and then directly into the user interface. From a teaching perspective this helps to reinforce the idea that modelling is both about representing the real world and designing software.

An important part of our teaching has been to identify specific issues that cause difficulties for students then provide specific, detailed guidance of how to ameliorate these difficulties. More specifically we have considered the difficulty of conducting a thorough business analysis before transitioning to design, the transition from a Use Case view to a behind-the-scenes view of the software architecture, the transition from design to implementation with specific attention being paid to the design of an interface that reflects the structure of the software. We have discussed these transitions in terms of patterns that capture good practice and the relationships between these patterns.

We present the patterns in a manner based around the metaphor of different development “rooms”. The first room is concerned with developing user personas it is adjacent to a room for developing an “analysis” model based on Soft Systems Methodology. This room contains patterns for developing a range of soft systems models including rich pictures, root definitions and conceptual models. An adjoining room contains patterns for making the transition from analysis to design by translating the analysis model into a Use Case Model with carefully structured documentation of each use case. The next room contains patterns for moving into physical

design and then into code. These include: “Develop a sequence diagram showing how domain classes may co-operate in the implementation of a use case”. This will involve ensuring that the detailed steps in our use case description relate to the messages being passed on the sequence diagram. A related pattern will explain how to assign operations to classes that map to messages on the sequence diagram.

As mentioned above when a number of patterns are related to each other in this way we describe the result as a “pattern language”. We are therefore trying to develop a pattern language to support information systems modelling. We would argue that patterns are particularly suited to this purpose. They are descriptive, not prescriptive (unlike most detailed development methods). They capture expertise in an open-ended format that lends itself to a “hypertextual” structure of resources with links between related patterns that can be explored without forcing a specific sequence of activities. The patterns can also be used as the basis for developing assignment specifications. We will say more on this latter topic in the closing sections of this chapter.

9.5 Running the Module

In light of the above discussion we have been able to propose the following guidelines for developing a module in this area:

1. Design a portfolio-based assessment that can be completed in instalments each instalment being aligned to patterns used in teaching. For each pattern we specify deliverables that can be represented in an assessment grid. In the case of the patterns described above, one instalment could be a Use Case model that is consistent with earlier conceptual models and which is described in steps that map to messages in a sequence diagram. The patterns then become part of the explanation of what is required and are clearly linked to the feedback grid.
2. Provide formative in-class surveys that encourage students to reflect on their understanding of key patterns. In the first example above can they provide an example of a many to many relationship between two classes that could be better represented by a third class? Can they see how the proposed solution would help? Can they apply this learning to the coursework case studies?
3. Encourage students to discuss the individual patterns and how they may be applied to case studies before they complete the in-class surveys or work on the assignment. Students should be encouraged to identify new patterns and fit them into the pattern language or to improve the documentation of existing patterns.
4. Collect data on a regular basis by inspecting samples of student coursework on a week-by-week basis and in-class surveys. Use the feedback to inform improvements to pattern descriptions and the identification of new patterns.

These four guidelines work together to steer the students through the assessment process by frequently monitoring their progress. Hopefully this will lead to continuous improvement in the clarity of the coursework specification and the teaching materials.

With respect to Step 4 above we employ a variety of different ways to collect evaluative information. These include: a pre-course questionnaire that we distribute before teaching begins this is intended to establish the background knowledge and expectations of our students; a series of anonymous in-class surveys to test students understanding and self-confidence in applying the patterns under discussion; short reflective essays were made part of the coursework portfolio in which students were asked to give their personal opinions about the usefulness of the pattern-based approach and focus group discussions were held in class.

In addition to the above, during marking, we carried out an analysis of the most common mistakes made by students in their coursework. We discovered a number of recurring mistakes and made changes to the pattern language to discourage these. A number of examples are given below.

- Inconsistencies between diagrams. For example operations appearing in the sequence diagram that are not present in the class diagram.
- Failure to use domain-specific vocabulary as presented in the case study materials. In the above example we referred to “Pathway” where others may have used the term “Course”. It is important that the language used in the models can be found in the user documentation.
- Operations that have ambiguous or misleading names. We have seen operations with names like “Update all” or “Reconsider” these names are almost meaningless to anyone but the original programmer.
- Database concepts (e.g. primary and foreign key dependencies) used in the domain model. The domain model is meant to be an abstract representation that might be used in the design of object oriented software or an ontology it is not a physical database design.
- Operations not supported by attributes or relationships. Some operations depend on the availability of connections to other classes or of data properties that must be included in the domain model.
- A lack of consistency between the SSM models and the Use Case Model. For example Use Cases that cannot be inferred from the activities in conceptual models.

We continue to work on developing patterns that will steer students away from these types of mistake. We plan to present these via a website based on our “rooms” metaphor with hyperlinks between related patterns. We would argue that working in this way has encouraged us and our students to consider important aspects of information systems design that are often overlooked in courses that teach Information Systems Modelling. A few of these are listed below:

- We encourage students to adopt multiple system viewpoints from different personas. The acknowledgment and exploration of these viewpoints emphasises

the important point that typically most systems have more than one purpose and many unexpected consequences.

- We are encouraging our students to consider the total problem situation and to be aware of the need to ensure that all, not just the most obvious, significant issues are addressed;
- Our approach is strongly goal-oriented. The central focus on Use Cases ensures that derived requirements are justified with respect to stated goals;
- All the techniques documented in our pattern language are well established and well tried. We haven't presented patterns for anything that has not, in one form or another, proved useful to developers. We hope the manner of presentation is more effective than user manuals or detailed methodology documentation but the intention is not to present anything new but to organise and document the distilled wisdom of the many talented software engineers and systems analysts who have worked in this area over the years.

9.6 Conclusion

This paper has described our approach to teaching Information Systems Modelling over a number of years. We have described the approach as being built around the scaffold of a multi-method systems development framework which we have documented in the form of a pattern language. This basic structure has been tried and tested through a number of feedback mechanisms (including in-class surveys, focus group discussions and reflective essays) and a concordant assessment strategy. The results obtained through these feedback mechanisms have encouraged us to continuously refine our teaching materials and assessment strategies—we believe these changes have all been improvements.

References

- Alexander C (1979) *The timeless way of building*. Oxford University Press, New York
- Ambler SW (1998) *Software process patterns*. Cambridge University Press
- Becker C, Betz S, Chitchyan R, Duboc L, Easterbrook S M, Penzenstadler B, Venters CC (2016) Requirements: the key to sustainability. *IEEE Softw* 33(1):56–65
- Blomkvist S (2002) The user as a personality: using personas as a tool for design. In: Position paper for the workshop 'theoretical perspectives in human computer interaction' at the interaction and presentation laboratory of the royal institute of technology, Sweden, September 3, 2002. Available at <http://www.nada.kth.se/~tessy/Blomkvist.pdf>
- Checkland P (1999) *Soft systems methodology: a 30-year retrospective*. Wiley, Chichester
- Evitts P (2000) *A UML pattern language*. Macmillan Technical, Indianapolis, Ind
- Gamma E, Helm R, Johnson R, Vlissides J (1995) *Design patterns: elements of reusable object-oriented software*. Addison Wesley, Reading
- Larkin MJ (2001) Providing support for student independence through scaffolded instruction. *Teach except Child* 34(1):30–34

- Salahat M, Wade S (2009) A systems thinking approach to domain-driven design. In: The proceeding of UKAIS2009 conference, Oxford University, Oxford, UK
- Wade S, Salahat M, Wilson D (2012) A scaffolded approach to teaching information systems design. *Innov Teach Learn Inf Comput Sci* 11(1):56–70

Chapter 10

Promoting Design Thinking Through Knowledge Maps: A Case Study in Computer Games Design and Development Education



Carlo Fabricatore and Maria Ximena López

Abstract Modern computing is pervaded by human-centric technologies which potentiate people’s capabilities to address complex problems and needs in contextualised and meaningful ways. Creating such technologies requires thinking approaches that overcome the limitations of traditional paradigms that focus on specific aspects of human–computer interaction in well-defined problem contexts. Design thinking is an ill-defined problem-solving strategy which addresses this need through a systematic and iterative process that integrates exploration, ideation and testing of possible solutions based on the participation of stakeholders, and the investigation and accommodation of their often-conflicting needs. Developing design thinking in students is key to face the challenges of an ever-changing and increasingly complex world, and it is therefore crucial to have approaches and tools that can support educational endeavours aimed at this. In this chapter we describe the use of knowledge maps to promote design thinking for game design and development students. Knowledge maps are a variant of hierarchical concept maps created by domain experts to support learners’ knowledge construction processes. Game design knowledge maps were conceived to integrate and structure multidisciplinary knowledge regarding game systems, players, player engagement principles, and design and testing processes. Their structure was planned so that students could explore imparted knowledge iteratively and incrementally, driven by a human-centric focus. The evidence collected from students so far indicates that knowledge maps integrate large amounts of information in an easily accessible structure which fosters student’s design thinking processes. The maps seem to support students in connecting themes and ideas, and guide them through the whole design thinking process. This suggests that properly structured imparted knowledge can be effective in helping students to learn “how” to think, not just “what” to think.

C. Fabricatore (✉) · M. X. López
University of Huddersfield, Huddersfield, England
e-mail: Carlo.fabricatore@gmail.com; C.Fabricatore@hud.ac.uk

M. X. López
e-mail: Ximelopez@gmail.com; m.x.lopezcampino@hud.ac.uk

Keywords Design thinking · Concept maps · Game design · Education · Problem solving

10.1 Introduction: The Need for Human-Centred Design Thinking in Computer Science Education

Nowadays computing devices permeate and shape many aspects of our lives. They influence the way we work, socialize, learn, play, and engage in the many other types of activities that define our daily living, and through which our identities are defined. Nowadays more than ever before computational devices and computer programs are conceived because of people and for people, to address their complex needs in complex contexts.

The defining power of computational technologies is due to how these are designed to mediate human activity, putting abstract technological features at the service of meaningful human needs and wishes (Jaimes et al. 2007). Accordingly, modern perspectives on computing emphasise the importance for Computer Science—henceforth referred to as CS—to fully embrace human-centric approaches (e.g. Taylor 2000; Jaimes et al. 2007; Grudin 2012). This implies that CS should be regarded as a scientific endeavour aimed at studying in equal measure what computational artefacts “can do” in abstract, how they can mediate human life, and how they can be designed and implemented for this purpose. By extension, CS education should aim at enabling students to design, develop and evaluate computational artefacts embracing human-centric perspectives.

User-centredness is not new to CS. In fact, focus on computer–human interaction has been key to drive the evolution of CS and CS education since the 1970s (Grudin 2012). “Human-centrism”, however, entails more than attention to the process of interaction between humans and computers. Adopting a human-centric approach requires focussing on the whole human experience, rather than purely on abstract features of technological artefacts (Taylor 2000; Giacomini 2014). Computational technologies should be treated as mediators of contextualised and purposeful human activity, focussing on how they can augment the user’s capabilities to interact with their environments in desirable ways, and how technological potentialities are perceived by the “user” (cf. Bannon 1991; Buchanan 1992; Giacomini 2014). For this, the user, the technological artefacts, and the context and purpose of technology uses should be treated in an integrative way, accounting for the possible effects that technology may have on users and their environments (Bannon 1991; Taylor 2000; Giacomini 2014). This requires a systematic way of thinking, suitable to tackle ill-defined problems through iterative processes of creation of solutions driven by user-centred evaluations: human-centric design thinking. By extension, promoting design thinking should be a key priority in education focussed on human-centred creation of technology (cf. Cross 1982; Oxman 1999; Dym et al. 2005; Dunne and Martin 2006; Fabricatore and López 2015, in press).

What are the characteristics of design thinking? What can be done to promote its development through formal education? To address these questions, in this book chapter we will first discuss the distinctive properties that characterize design thinking as a systematic reasoning strategy. Then, drawing from key research on design thinking education we will outline key strategies to foster design thinking through formal education. Accordingly, we will analyse a key challenge that frequently hampers design thinking education: the provision of knowledge to students. We will then introduce our response to this challenge: design knowledge maps. These are a variant of concept maps that we have created in order to provide structured knowledge suitable to scaffold the development of design thinking in the domain of computer game design and development education. Through a case study we will outline the structure of a game design knowledge map, discussing its underpinning rationale and exemplifying its implementation. We will then report preliminary impacts investigated so far. We will conclude this chapter with some reflections on the need to increase efforts to promote design thinking within CS education, and the possibility to use design thinking knowledge maps outside the domain of game design and development.

10.2 The Engine of Human-Centredness: Design Thinking

Design thinking can be regarded as an applied reasoning strategy that integrates investigation, strategic planning, construction and testing of systems to address open-ended, ill-defined problems involving multiple interacting stakeholders (Buchanan 1992; Taylor 2000; Dorst 2011). The core purpose of design thinking is to formulate solutions aimed at generating impacts valuable for all the stakeholders involved in the problem situation (Buchanan 1992; Dunne and Martin 2006). Unlike well-defined problem-solving approaches, design thinking addresses problem situations in which there exist no optimal or definitive solution, and there is no possibility to gain full and stable knowledge of requirements and working principles underpinning problem scenarios. Stakeholders may be (and often are) influenced by conflicting needs and constraints. These needs and constraints are likely mutable, not fully knowable nor predictable, as are the environmental conditions that define the contexts in which stakeholders exist and interact. Consequently, design thinking explores and models problem situations through different perspectives, ideating, implementing and testing alternative solutions based upon incomplete and uncertain information. This contrasts with well-defined problem-solving strategies that aim at formulating “optimal” and “definitive” solutions based on full and stable understanding of the problem situation. This distinguishes design thinking as an iterative problem-solving process which systematically integrates abductive, deductive and inductive reasoning in a cycle of progressive approximation to an “acceptable”, “good enough” solution (Fig. 10.1). The structure and rationale of the design thinking process, and its focus on stakeholders and their context characterize design thinking as a human-centric problem-solving approach suitable to tackle complex problem situations (Buchanan

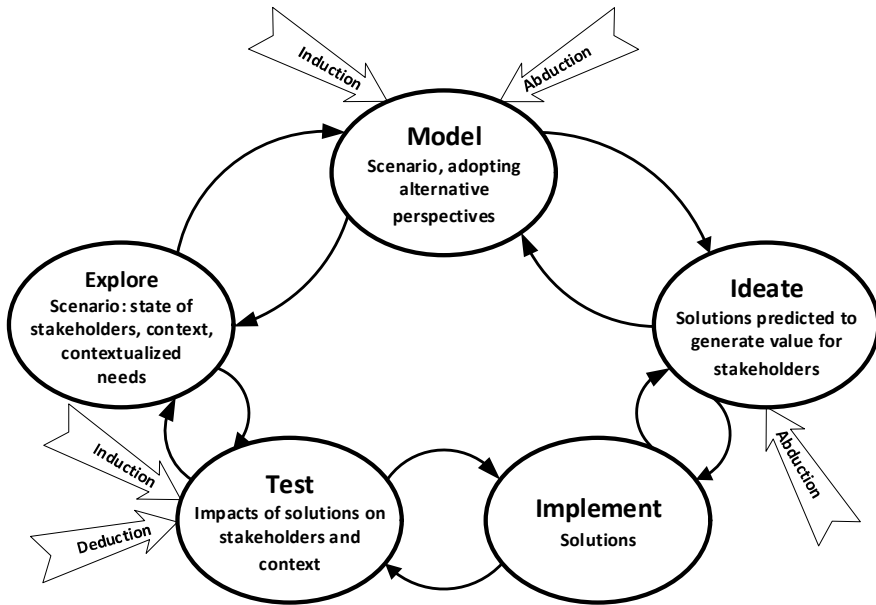


Fig. 10.1 The design thinking process

1992). Furthermore, the way design thinking integrates generation of novel ideas with their practical implementation and testing makes of design thinking a process of innovation (Dunne and Martin 2006).

10.3 Design Thinking as a Systematic Problem-Solving Approach

As previously mentioned, a key and distinctive feature of design thinking is how it systematically integrates abduction, induction and deduction in problem-solving. A problem can be generally regarded as discrepancy between a current state of matters and a desirable goal state in a given situation. A problem situation can be viewed as a system of interacting elements. Accordingly, a problem is defined by information regarding: the type and state of elements involved in the system; the contextual conditions in which they interact; the working principles that regulate their interactions and state changes; and the desirable goal to attain.

When a problem situation is stable, it can be reliably investigated and defined through direct exploration, using deductive and inductive reasoning (Dorst 2011). First of all, stability implies that the state of system elements can be known through exploration of the problem situation at any point in time. Then, if the working principles are known, given a present state of the problem situation it is possible to

apply deductive reasoning to predict future states: knowing “what” is involved in a situation and “how” it functions allows predicting “outcomes” of the functioning (Dorst 2011). If the working principles are not known, inductive reasoning can be applied to hypothesize them based on observation of patterns in system state changes: knowing “what” is involved in the situation and observing the “outcomes” of its functioning allows hypothesizing “why” it may have changed and, by extension, “how” it functions (Dorst 2011). These mechanics allow to apply a linear problem-solving approach to define how to modify a system in a desirable way. If current state, working principles and goal state are defined, abductive reasoning allows hypothesizing which aspect of the system should be modified so that the goal state will be attained: knowing “how” things work and knowing the desired outcome of their working allows defining “what” things should be modified to achieve such outcome (Dorst 2011). Deductive reasoning can then be used to confirm or refute hypotheses, through implementing the planned changes—i.e. the “solution”—and testing their effects against the predicted outcomes. For example, imagine that one wanted to improve the strength of tennis players’ swings (goal state). To achieve this, knowing that the weight of a racket affects the strength of tennis strokes according to well-defined mathematical models (working principles) one might design a lighter racket, using for the very first time a revolutionary material: graphene (Already used for racquets!?! Too bad...).

In the case of open-ended problem scenarios, the situation may be significantly more complex, unstable and ill-defined. In these cases, the elements involved in the problem situation and the working principles that regulate them may be unknowable, changing and uncontrollable to some relevant degree. This requires using abductive reasoning for the iterative formulation and testing of parallel hypotheses on working principles and solutions, through a systematic process which is distinctive of design thinking (Fig. 10.1). Given a desirable value (goal) to attain, the designer frames the problem situation adopting alternative perspectives and formulating different and likely complex models (Buchanan 1992; Dorst 2011). Each model includes inductively inferred and abductively hypothesized working principles, and a thesis that associates these with the desired value to attain: “IF we look at the problem situation from this viewpoint, and adopt the working principle associated with that position, THEN we will create the value we are striving for” (Dorst 2011, p. 525). A model is then selected and, based on the hypothesized working principles, a solution to attain the desired value is abductively formulated and implemented. Deductive reasoning is then used to test the hypothesized solution and, by extension, the underpinning working principles. If the solution generates the predicted value, its validity and the underpinning working principles are inductively generalized (Dunne and Martin 2006). Otherwise, abductive hypotheses regarding solution and underpinning working principles may be reformulated and tested through a new iteration of the entire process, or the framing may be changed all entirely (Dunne and Martin 2006; Dorst 2011).

To exemplify the process, let us consider an oversimplified fictional case. Pretend that the value pursued was enhancing wellbeing in the context of rural jungle communities in Nowhereland. After exploring the problem situation one might hypothesize

that people living in those communities enjoy socializing (working principle). Therefore, one might theorize that if socialization opportunities were enhanced, then well-being would increase (abductive thesis, relating hypothesised working principles and desired value). To complete the framing, one might have observed that most people have old mobile phones whose usage is limited to voice calls and SMS. At this point one might have sufficient information to hypothesize that a good solution could be an innovative product: JungleTxTClub®, the first ever SMS-based social networking platform interfaced with larger, non-SMS-based platforms (Already done?!? Oh no...) Assume that, embracing the “social networking framing” of the problem situation, one developed and tested this solution for a month, only to discover that the average use decreased drastically day after day. This apparent failure would then lead to reviewing the working hypotheses and the solution. Through this process, one might interview a sample of users, discovering that the idea of digital social networking is appreciated. However, users found typing on the mobile keyboard cumbersome. One might then infer inductively that there is a problem with the user interface, reviewing the solution accordingly, and initiating a new cycle of the design thinking process.

10.4 Design Thinking as Human-Centric, Social Process

Stakeholders are the fulcrum of the design thinking process (Buchanan 1992; Taylor 2000; Dunne and Martin 2006). Design thinking tackles scenarios which are “problematic” first and foremost because they involve stakeholders who have latent or manifest needs that require addressing. Stakeholders do not consider themselves mere “users” of products and services: they are human beings concerned with the quality of their lives, their activities, and the impacts that products and services can have on these (Giacomin 2014). This is why the primary focus of the design thinking process is not “the solution”, but rather “the need in context” and “the value” to be generated to address contextualised needs. By extension, design thinking is not only concerned with the direct impacts of a “solution”. Driven by social and ethical considerations, it accounts for broader implications that “solutions” may have on stakeholders and their contexts (Buchanan 1992; Taylor 2000).

Design thinking also acknowledges that stakeholders are diverse. Their perspectives and circumstances may be different. As a consequence, their needs may conflict. A given scenario will likely not represent the same “problem” for all stakeholders, and what could be perceived as a “solution” by some, might represent a “problem” for others (Buchanan 1992; Taylor 2000; Dorst 2011). It is in order to address diversity that design thinkers frame problem scenarios through different perspectives, exploring alternative frames in search of models that can accommodate as much as possible all the involved stakeholders.

Stakeholder’s needs and circumstances may be mutable and not fully manifest: not everything may be knowable through observation, and what has been observed may change. To address this, design thinking integrates observation, dialogue and

inquiry in order to involve stakeholders throughout the problem-solving process (Buchanan 1992; Dunne and Martin 2006). Background information on stakeholders is obviously a key starting point to frame the problem situation tackled. Observation serves to infer relevant behavioural patterns of the stakeholders, salient aspects of their circumstances, and impacts of implemented solutions. Inquiry and dialogue are used to gather information directly and intentionally provided by stakeholders, which closely represents their perceptions of the problem situation. Stakeholders are not just a source of information. Through dialogue stakeholders are involved in joint decision-making and evaluation processes, as in the case of the definition of the value to be attained, the selection of framing perspectives, and the evaluation of impacts attained (Buchanan 1992; Dunne and Martin 2006). This promotes the accommodation and integration of conflictive views in case of conflict.

10.5 Design Thinking as an Innovation Process

Design thinking cannot be fully outlined without highlighting its intimate connection with creativity and innovation. Exploring alternative perspectives and accommodating conflicting needs requires thinking outside existing and evident alternatives, creating and testing new options, and embracing constraints as challenges, rather than barriers and causes for compromise. This makes of creativity an intrinsic element of design thinking. The exploration of non-yet-existing possibilities is not purely conceptual. As possibilities are iteratively developed through cycles of design and testing, design thinking integrates conceptual analysis and practical synthesis in a dialectical process. Thus, design thinking fully mirrors what innovation is: a process of generation and implementation of novel and useful ideas in response to open-ended problems and opportunities (Fabricatore and López 2013). Design thinking as a whole IS a process of innovation.

In sum, design thinking can be regarded as a human-centric, systematic and creative process ideal to address meaningful human needs through the generation of innovative technological solutions. This is of paramount importance nowadays, when people need solutions to improve their lives in increasingly complex contexts, industries compete to generate innovative solutions, and formal education strives to equip students to tackle the challenges of our complex world (Dym et al. 2005; Fabricatore and López 2015; Koh et al. 2015). How can design thinking be fostered within formal education?

10.6 Fostering Design Thinking Through Formal Education: Implications and Challenges

Since the 1990s there has been a growing interest in studying the cognitive aspects of design, and educational research on design thinking has developed accordingly (Cross 1982; Oxman 1999, 2004). It has been acknowledged that learning design thinking means enhancing real-world problem-solving capabilities (cf. Cross 1982; Oxman 1999; Dym et al. 2005; Dunne and Martin 2006). Researchers have consequently investigated strategies to promote the development of the thinking skills and attitudes involved in design thinking, and foster the conscious assimilation of the reasoning patterns underpinning the design thinking process (cf. Oxman 1999, 2004; Dym et al. 2005; Fabricatore and López 2014, in press). A significant consensus has been reached regarding key features that educational strategies should incorporate for these purposes. Drawing from our past research (Fabricatore and López 2014, in press) and echoing key trends in design thinking educational research (cf. Cross 1982; Oxman 1999, 2004; Dym et al. 2005; Dunne and Martin 2006; Koh et al. 2015), we suggest that these features can be summarised as follows:

- i. Focus on project-based learning activities, aimed at addressing open-ended, ill-defined problems involving multiple interacting systems, and underpinned by multiple knowledge domains
- ii. Contextualization of problem situation mirroring real-world scenarios, involving: diverse stakeholders; properties of contextual conditions and stakeholders which are neither fully knowable nor fully predictable; social as much as technological implications for the involved stakeholders.
- iii. Iterative and incremental organisation of project work, integrating multiple cycles of design, implementation and evaluation of solutions.
- iv. Promotion of student's collaboration and self-organisation of project work, involving: team-based project work motivated by the intrinsic complexity of the problem tackled; roles and responsibilities organised by students, based on iterative exploration of problem scenarios and provisional outcomes of project work.
- v. Promotion of direct and indirect exploration of and interaction with key stakeholders involved in the problem scenarios tackled.
- vi. Provision of adaptive pedagogical support, involving: upfront provision of explicit core knowledge, firstly useful to promote shared understandings, and later on freely accessible to learners as a scaffold; adaptive provision of explicit supplementary knowledge, ad hoc, depending on project progression and learner profile/learning style; enabling, non-prescriptive tutor feedback, promoting the adoption of alternative perspectives to frame problems, and the prediction and critical analysis of implications of possible solutions; mediation of team dynamics, to promote mutual understanding and self-organisation.

- vii. Tailorization of project constraints to maximize their enabling value, ensuring that: project-specific constraints promote the exploration of alternative perspectives and the formulation of different solutions to the problem tackled; administrative constraints (e.g. academic policies and regulations, etc.) do not hinder student's self-organization and possibilities to embrace varied approaches to project work.

All the above features are underpinned by a pivotal element: the knowledge imparted to learners. Its contents and structure should reflect the comprehensive type of knowledge involved in design thinking, and support knowledge building processes similar to those carried out by expert designers (Cross 1982; Oxman 1999; Fabricatore and López, in press).

Designers develop solutions through constructing knowledge regarding a problem situation which can be generally viewed as a system of interacting elements (cf. Cross 1982; Fabricatore and López, in press). As they explore problem situations, designers observe, identify, classify and infer concepts and their connections, building meanings in order to hypothesize working principles that govern the problem system tackled, and ideate possibilities to influence it in desirable ways (cf. Buchanan 1992; Oxman 1999; Dorst 2011). This can be assimilated to a purposeful process of construction of knowledge based on meaning-making: meaningful learning (Novak 2010). The knowledge acquired and constructed through the design thinking process concerns the “what” of design, as much as the “how” and the “why” (cf. Cross 1982; Oxman 1999, 2004; Dym et al. 2005). This knowledge can be regarded as a system of concepts and propositions (Novak 2010). It may be factual, inductively inferred, abductively hypothesized, and deductively corroborated through the design thinking process. Regarding the “what”, knowledge involved in design thinking relates to attributes and relationships of key elements of the problem system investigated—including stakeholders and other relevant contextual elements—and the solution being designed. Regarding the “how”, knowledge regards techniques and tools useful to explore and modify the system being tackled, and strategies to organize and apply techniques and tools. The “why” concerns working principles regulating interactions between system components. This knowledge allows designers to reflect on the suitability of tools, techniques and strategies in relation to the effects that they might have on the systems tackled. It serves to predict or critically evaluate direct effects and broader implications of possible solution approaches.

The contents of the knowledge imparted to students should be integrative, comprehensive and multi-disciplinary, covering the “what”, “how” and “why” of design (Cross 1982; Oxman 2004; Dym et al. 2005; Dunne and Martin 2006). For this, the structure of knowledge is as important as its contents (Oxman 1999, 2004; Novak and Cañas 2008). The knowledge imparted should be organised connecting information about the “what”, “how” and “why” in scaffold structures which students can assimilate and use as a basis to build their own conceptual structures (Oxman 1999). The knowledge imparted should also be mainly explicit and objective, rather than subjectively defined and verbally conveyed by tutors (Oxman 2004). Non-explicit knowledge is only temporarily embodied in the words of the tutor, and is dependent

on the tutor's experience, personality, cognitive style, and on the student's immediate interpretation. Instead, explicit and objective knowledge provides to all students equal possibilities to access and share the same knowledge whenever needed (Novak and Cañas 2008; Fabricatore & López, in press).

The definition and provision of structured knowledge thus represents a pivotal challenge in design thinking education. We tackled this challenge in the domain of computer game design and development education through creating a bespoke knowledge representation tool, which we labelled game design knowledge map.

10.7 Game Design Knowledge Map: Structure and Rationale

A design knowledge map is a variant of concept map. Concept maps are visual representations of structured knowledge, articulated as graphs in which concepts are usually represented as nodes, and relationships between concepts as links between nodes (Novak and Cañas 2008). Concept maps are suitable to explicitly represent knowledge structures, and can help students to learn how to learn (Novak 2010). For this, concept mapping can be used in two ways: representation of evolving student knowledge, and mapping of expert knowledge provided to students (Novak and Cañas 2008; Novak 2010). In the first case, concept maps created and updated by learners serve to make their knowledge explicit as its construction process unfolds. This allows students to reflect on their knowledge structures, share them with others, and consciously modify them (Novak and Cañas 2008), facilitating an interplay between individual and collective learning processes which is key in open-ended problem solving (Fabricatore and López 2014). In the second case, expert skeleton maps are prepared by domain experts to facilitate the initiation of a knowledge construction process, and to scaffold the progressive assimilation and generation of new knowledge (Novak and Cañas 2008; Novak 2010). Knowledge construction unfolding through the exploration of a domain should be supported by reliable initial knowledge of that domain (Novak and Cañas 2008). Expert skeleton maps can be effectively used to model initial knowledge, creating scaffold structures based on which learners can construct new knowledge, assimilating, ideating and integrating new concepts and relationships (Novak and Cañas 2008). Scaffolding initial learning minimises the risk of introduction of misconceptions in the knowledge structures built by learners, and facilitates their remediation (Novak 2010). Furthermore, expert skeleton maps can foster the development of thinking strategies: by integrating knowledge about "what" experts think as well as "how" and "why" they think, skeleton maps can help students learn "how" to think as much as "what" to think.

To tackle the challenge of knowledge provision in the context of game design thinking education, we conceived the game design knowledge map as an expert skeleton map aimed at supporting knowledge construction throughout the game design and development process. For this, we conceptualised game design thinking

as a human-centric problem-solving process (Fabricatore and López 2014, in press), aimed at addressing a specific core question:

- How can a game attract players, involve them in the game and keep them playing?

This question was formulated to explicitly represent the desired outcome of the game design problem-solving process (i.e. an engaged player), and therefore serve as focus question to support a meaningful exploration of the knowledge map (Novak and Cañas 2008).

In design thinking terms, our conceptualisation considers players as the key stakeholders of the problem scenario tackled by the game designer. Sustained engagement is then the key value to deliver to the stakeholders, and the game represents the system to be designed in order to deliver such value. Finally, the context of use is represented by the physical and social space which frames a gameplay activity but is not directly involved in it. The game design thinking process then requires to abductively hypothesize working principles defining, for a given target audience, which abstract gameplay activity aspects could be accessible, attractive and engaging, and why (e.g. challenge, reward, support, control, narrative, etc.) Games can be consequently created as systems of interacting elements suitable to promote the desired gameplay activity aspects, defining concrete features based on the hypothesized working principles. Designed features can be finally implemented and tested, in order to confirm or (re)defined game system features and engagement working principles, as need be. This creation process is carried out iteratively and incrementally, through cycles of design, implementation, testing, and evaluation, until the desired engagement value is generated.

According to this conceptualisation, our design thinking knowledge map integrated multi-disciplinary expert knowledge from the domains of computer science, game design and development, systems design and the social sciences, extrapolated from leading academic and practitioner literature. Embracing a human-centric approach, we determined that the knowledge to be included in the map had to be relevant to support design decision-making processes and cover the “what”, “how” and “why” regarding:

- (i) Users
- (ii) Elements and patterns of organization of game systems
- (iii) Elements and patterns of organization of natural contexts of use
- (iv) Engagement principles that allow predicting and explaining how users may perceive and consequently interact with systems, in relation to their needs, preferences and contexts of use
- (v) Approaches for testing impacts of a system on user and context of use

Nodes in the game design knowledge map encapsulate concepts reflecting key sub-problems or activities involved in the user-centric game design endeavour, along with guiding questions conceived to orient and stimulate related reasoning, investigation, ideation of alternative options, and decision-making (Fig. 10.2).

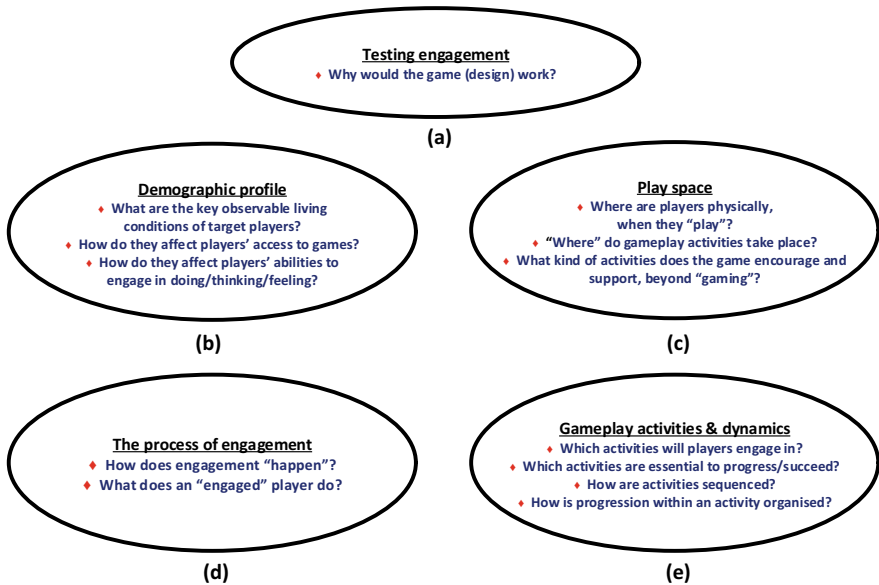


Fig. 10.2 Examples of nodes representing: **a** activity of the design endeavour; **b** sub-problem concerning the player; **c** sub-problem concerning context of use; **d** sub-problem concerning engagement working principles; **e** sub-problem concerning features of the game system

In order to promote iterative and incremental exploration of the game design knowledge map, we defined its general structure based on a hierarchical star topology pattern. The core of the map covers four key activities involved in the game design endeavour:

- (i) Profiling target players: defining key characteristics of the target audience which may affect the way they perceive and interact with a game system.
- (ii) Planning player engagement: identifying working principles suggesting which game aspects could trigger and sustain player engagement and why.
- (iii) Designing the environment for engagement: analysing and defining features of the game system and its context of use to promote player engagement according to the working principles identified through (ii)
- (iv) Testing engagement: prototyping/implementing features of the game system according to what designed through (iii), and testing to verify effects predicted according to the working principles identified through (ii)

Each node in the core represents a design activity with related guiding questions. Relationships between activity nodes represent their key logical relationships. Activity nodes and relationships were conceived to promote comprehension and exploration of what needs to be done, for what purpose, and how activities support one another throughout the iterative design process. Thus, the core of the knowledge map is overall aimed at supporting the student's assimilation of the whole design

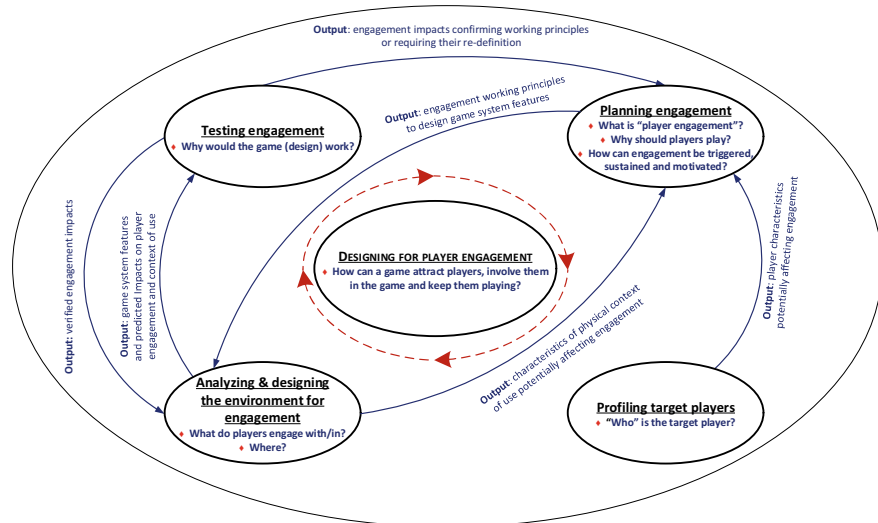


Fig. 10.3 Core of the game design knowledge map

process, fostering the understanding of its core activities as much as the thinking underpinning their planning and execution (Fig. 10.3).

Each activity node is connected to a hierarchy of relevant design sub-problem nodes, along with guiding questions to orient related investigation and decision-making (e.g. Fig. 10.4). Sub-problems nodes reflect objectives to be tackled through each activity, and have been defined to promote comprehension of which sub-problems a given activity might have to address, and reflection on what should be considered in order to address them.

For a given node, a game design knowledge map may provide additional knowledge to fully cover “what” to address, “how” and “why”, through a 4-layered knowledge block (Fig. 10.5). The first layer of the block aims at promoting deeper reflection on “what”, through outlining key foci and issues that should be considered when addressing the concept encapsulated in the node. The second layer aims at promoting deeper reflection and comprehension of “what” and “why”, through presenting big ideas explaining the nature and importance of the issues presented in the first layer. The third layer aims at promoting deeper comprehension of “how” and “why” to tackle the issues presented in the first layer, accounting for the big ideas presented in the second layer. For this, appropriate strategies are provided (e.g. formal techniques, guidelines and principles). Finally, the fourth block aims at providing knowledge that directly underpin issues, big ideas and strategies, through presenting references to relevant literature and other sources (e.g. podcasts, webinars, etc.)

Overall, the organisation of the game design knowledge map was conceived to facilitate incremental exploration, assimilation and construction of operational knowledge throughout the design process, driven by its core activities and their outcomes. For this, knowledge was clustered so that the apprentice designer could

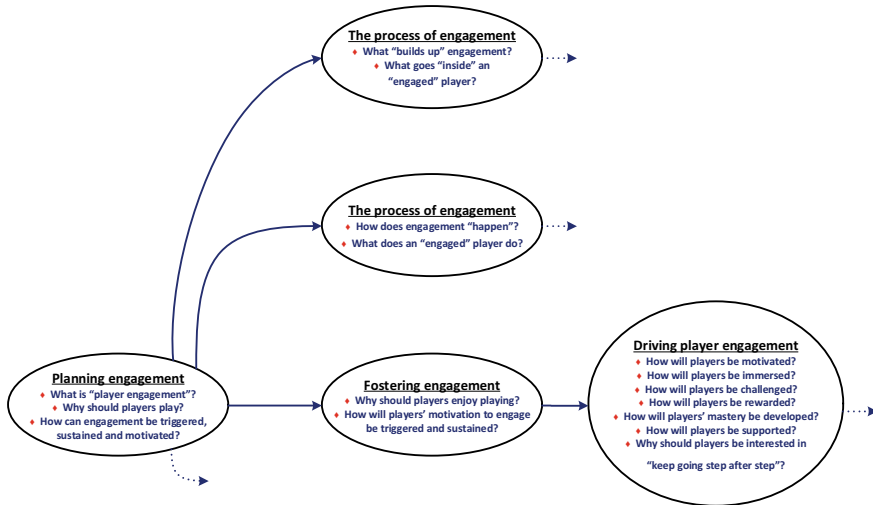


Fig. 10.4 Example of hierarchy of sub-problems

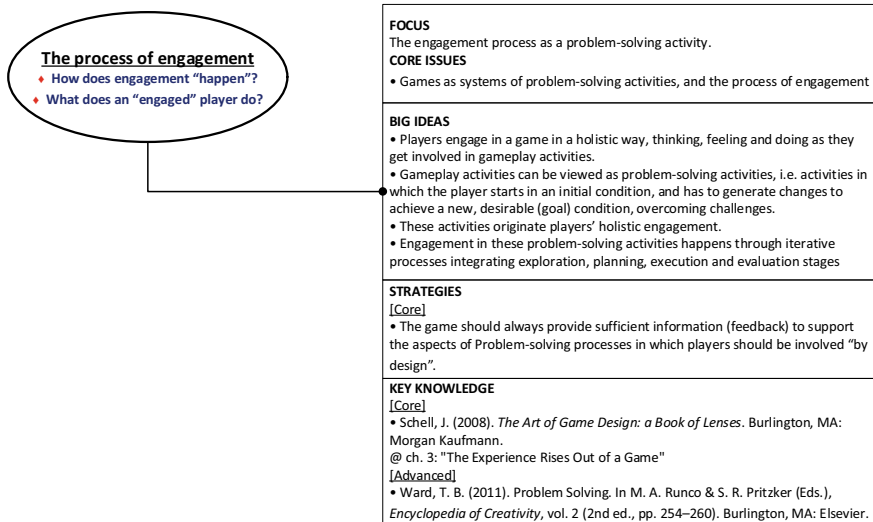


Fig. 10.5 Example of knowledge block

easily assimilate, find and access it depending on the activity being planned or carried out, and the sub-problems tackled. Nodes and their related knowledge blocks were hierarchically organised so that their incremental exploration could drive incremental understanding of activities and sub-problems, and acquisition of conceptual tools to tackle them. Finally, the structure of the knowledge map was underpinned by a rationale of "operational incrementality and immediacy". On the one hand, the deeper the

exploration of the hierarchy of nodes and related knowledge blocks, the greater the possibility to gain a deeper, more detailed understanding of design sub-problems, possibilities to tackle them and underpinning rationales. On the other hand, each node and each layer of the related knowledge block was planned to contribute by itself knowledge immediately useful to progress the design process: students do not need to explore the whole map to gain benefits, as each node may represent an opportunity to reflect and design “more” or “better”.

10.8 Impacts of the Game Design Knowledge Map: Some Preliminary Results

We developed the game design knowledge map and used it in a range of game design and development modules delivered at the School of Computing and Engineering of the University of Huddersfield (UK). The version of the map discussed in this study was developed in 2014, and used in 2014–15, 2015–16, 2016–17 and 2017–18, in a total of eight level five and four level six modules of a BA Game Design programme (second and third undergraduate year, respectively).

A master map was created to comprise and integrate all the contents covered by all modules, to promote continuity and transferability of learning for students attending different modules at different levels. Then, for each module there was a specific map, consisting in the portion of the master map that addressed the intended learning outcomes of that module. Thus, for each module the map was made available in two versions: module-specific, strictly covering the contents of the module’s syllabus, and the entire master map.

Each year the master map has undergone updates to ensure currency of contents (e.g. updating examples), address clarity issues (e.g. clarify phrasing), or update examples. This, however, has not significantly changed the structure of the knowledge map, nor its core contents.

For each module both versions of the knowledge map were made available to students online, through the University virtual learning environment (Blackboard). Students had the option to consult the maps online or download and use them offline.

The use of the map was not mandatory, and alternative resources were made available to access the required information through the virtual learning environment (e.g. lecture slides and notes from former editions of the modules, official module readings, etc.)

For all the modules the academic year was articulated in 24 sessions, on a weekly basis (excluding holiday breaks). During the first seven to nine sessions tutors used the map to introduce core contents, i.e. the ones essential to fulfil the module learning outcomes and corresponding to the module’s syllabus. Core contents were presented through a combination of seminars, case studies and guided workshops.

For the remaining weeks students worked exclusively on a game project based on a brief provided at the beginning of the academic year. The project was articulated in two or three of formative milestones (depending on the module) and one final summative milestone. Formative milestones had the main purpose of reviewing project progression and providing formative tutor feedback to provisional project outcomes. Such feedback was based on contents of the knowledge directly related to provisional project outcomes, as well as contents not apparently leveraged in the project, but which could orient future developments. Project workshops were run between milestones. These represented an instance to promote peer testing and discussion of projects, as well as further provision of tutor formative feedback, all based on the contents of the knowledge map (module-specific or master, depending on what individuals or groups of students decided to use).

In each module students were required to document their design work through a combination of written and visual materials, according to some module-specific requirements. For this, a module-specific template was provided to them, with guidelines to define and organise contents in a structure that reflecting the contents and organisation of the appropriate module-specific version of the game design knowledge map. Students were also required to produce a structured project journal, documenting key decisions and reflections on the production process and state of the project after each milestone. The journal was structured so that students were encouraged to reflect on their project using contents of the module-specific knowledge map as “lenses” to evaluate their design decisions and their implications.

We are currently investigating data regarding the impacts of this strategy, and the specific influences that the game design knowledge map has had on the student learning experience. In this chapter, we present the results from 51 s-year students that used the knowledge maps in two game design and development modules during the academic year 2014–2015. Participants were predominantly male (84.3%) with a mean age of 21.2 (SD = 2.49). Students who were enrolled in both courses had access to the module-specific knowledge map and to the same master map. At the end of the academic year, students completed a module evaluation questionnaire, which included an open-ended question asking about the main positive aspects of the knowledge maps. The qualitative data gathered through this question were analysed using a content analysis procedure as described by O’Cathain and Thomas (2004). Thematic categories were identified and coded by the two authors, who jointly decided the categorisation scheme, individually classified the comments and discussed coding disagreements until consensus was reached. Coded comments were then subject to descriptive quantitative analysis to identify trends in student’s opinions.

Results are presented in Table 10.1. Several positive aspects were perceived by students, among which three can be identified as major trends:

- a. **Enhance accessibility of knowledge.** More than 60% of students expressed an appreciation for the quantity, clarity and ease of access of the information available in the knowledge maps. Many students valued the presentation of a large amount of information in a well-structured format, which made contents easy to understand and quick to find. Examples of student’s comment were: “It

Table 10.1 Positive aspects of the knowledge maps identified by student

| Category of comment | N (Total n = 51) | Frequency (%) |
|--|---------------------|---------------|
| Cognitive and operational accessibility of knowledge | 33 | 64.7 |
| Quantity, clarity and structure of content | 30 | 58.8 |
| Ease of access | 3 | 5.9 |
| Support for thinking | 22 | 43.1 |
| Support for conceptualisation and analysis of the design situation | 13 | 25.5 |
| Support for formulation of rationale | 5 | 9.8 |
| Support for ideation of solution | 5 | 9.8 |
| Support for planning and execution of design process | 10 | 19.6 |
| Support to structure and produce design documents | 9 | 17.7 |
| Support to fulfil module requirements | 4 | 7.8 |
| Support for collaborative processes (organisation of teamwork, definition of shared goals) | 3 | 5.9 |
| Support for interestingness and enjoyment of design process | 1 | 2.0 |
| Transferability to other courses/ domains | 1 | 2.0 |
| No comments | 5 | 9.8 |

provided a quick and easy access to information to help us”; *“Easy to follow”* and *“It had all the information in an easy way to understand it”*. These results are in line with literature indicating that organisation of knowledge is as important as the content, since the structure of information implicitly embeds methods to approach and manipulate that knowledge (Kokotovich 2008; Oxman 2004). The results suggest that the knowledge maps were structured in a way that enabled the cognitive and operational access to the information, facilitating its use and application in the design process.

- b. **Support for thinking.** About 44% of students perceived that the knowledge maps supported their thinking processes, helping them to structure their ideas and analyse the design problems tackled. Some students made general comments about the support provided by the knowledge maps, like *“[It] asks enough questions to get you thinking”* and *“Helps to guide your thought processes”*. Other students reported more specific descriptions of the cognitive process aided by this tool. Among these, most student’s comments were related to the usefulness of the knowledge maps to conceptualise and analyse the design situation. They observed that *“The breakdown of questions made it easier to understand and analyse [the problem]”*, *“helped me to concisely explain my ideas”*, *“link different topics”* and *“made my study easier—more informed ideas + clearer rationale in my work”*. Previous studies (e.g. Kokotovich 2008) have compared thinking strategies of novices and experts, finding that novices often rush towards a design solution based on insufficient analysis and understanding of the design problem to be

tackled, evidencing shallower and less structured thinking processes. Conversely, the expert's approaches presented most of the features of design thinking, described as iterative, systemic-oriented, and acknowledging the complex relationship between design elements. In this regard, the knowledge maps emerged as valuable tools to support students in their design thinking, promoting the organisation of their analyses in a more reasoned, systematic and deeper way.

Students also highlighted the role of the knowledge maps in supporting the planning and execution of the design process, as well as the production of design documents. Representative examples of comments were that it *“offered a lot of information to help in the design process”*, *“the work was more manageable by using them”*, and *“the map allows organization of the project and makes it more efficient”*. Although these topics were less frequently mentioned by students, we believe they represent further indicators of benefits of using the knowledge maps. These results suggest that knowledge maps helped student not only to understand “what” to do, but also “how” to do it, directing students towards more appropriate strategies and methods for accessing and handling the design knowledge.

10.9 Conclusive Reflections

In this chapter we have discussed the importance of human-centrism in modern computing, and the consequent need of adopting human-centric perspectives in CS education. We have emphasized the necessity for CS education to foster design thinking, being this a systematic problem-solving strategy key to conceive human-centric computing technologies suitable to address meaningful human needs in complex, ill-defined problem contexts. Accordingly, we have highlighted the importance and difficulty of imparting explicit, structured knowledge to support student learning regarding “what” to think, as much as “how” to think and “why”. We have then proposed our approach to tackle such challenge of in the context of game design and development education: game design knowledge maps.

Game design knowledge maps are a variant of hierarchical concept maps, conceived to integrate and structure multidisciplinary knowledge regarding game systems, players, player engagement principles, and design and testing processes. We have designed the structure of game design knowledge maps so that students can explore imparted knowledge iteratively and incrementally, driven by the design problem-solving processes tackled and their provisional outcomes.

We have developed and tested game design knowledge maps for over four years, with encouraging results. The evidence analysed so far indicates that knowledge maps integrated large amounts of information in an easily accessible structure which fostered student's thinking processes, helped them connect themes and ideas, and guided them through the design process. All accounted for, this suggests that properly structured imparted knowledge can be effective in helping students to learn “how” to think, not just “what” to think.

Looking onwards, we plan of course to complete a thorough investigation of impacts of game design knowledge maps that we have registered so far. We believe, however, that the applicability and potential benefits of design knowledge maps should be investigated in other domains as well. Although the information contained in game design knowledge maps is obviously domain-specific, their prototypical structure and underpinning rationale are not. Many are the fields in which design can be conceptualised as a human-centric problem-solving process that accounts for the needs of key human stakeholders, context of use and possible solution to address stakeholder's contextualised needs (Buchanan 1992). Similarly, analysis of user's profiles, needs and context, definition of engagement strategies, and design and testing of solutions are activities common to design processes in diverse fields (Buchanan 1992; Dunne and Martin 2006). Hence, we believe that design knowledge map could be valuable tools to impart structured knowledge and foster design thinking in fields many other fields focussed tackling human-centric ill-defined problems, such as engineering, environmental planning, economics, architecture, etc.

Furthermore, the rationale underpinning the knowledge maps is based on the assumption that human-centric design is a process that relies on the abductive formulation hypotheses linking contextualised stakeholder's needs, value to be generated in order to satisfy those needs, current state and functioning of stakeholders, and systems to be designed in order to generate the desired value. Therefore, we believe that variants of the knowledge map could be developed for any other fields of education in which students are required to tackle open and human-centric problems.

References

- Bannon L (1991) From human factors to human actors: the role of psychology and human-computer interaction studies in system design. In: Greenbaum JM, Kyng M (eds) *Design at work: cooperative design of computer systems*. Lawrence Erlbaum Associates, Hillsdale, NJ, pp 25–44
- Buchanan R (1992) Wicked problems in design thinking. *Des Issues* 8(2):5–21
- Cross N (1982) Designerly ways of knowing. *Des Stud* 3(4):221–227
- Dorst K (2011) The core of “design thinking” and its application. *Des Stud* 32(6):521–532
- Dym CL, Agogino AM, Eris O, Frey DD, Leifer LJ (2005) Engineering design thinking, teaching, and learning. *J Eng Educ* 94(1):103–120
- Dunne D, Martin R (2006) Design thinking and how it will change management education: an interview and discussion. *Acad Managem Learn Educ* 5(4):512–523
- Fabricatore C, López X (2013) Fostering creativity through educational video game development projects: a study of contextual and task characteristics. *Creat Res J* 25(4):418–425
- Fabricatore C, López MX (2014) Complexity-based learning and teaching: a case study in higher education. *Innov Educ Teach Int* 51(6):618–630
- Fabricatore C, López X (2015) Higher education in a complex world: nurturing “chaordic” influencers. In: *Proceedings of the sixth advanced international colloquium on building the scientific mind (BtSM2015)*, Jupiter, FL, Learning Development Institute, pp 1–11. Retrieved from: <http://www.learndev.org/BtSM2015.html>

- Fabricatore C, López X (in press) Education in a complex world: nurturing chaotic agency through complexity science and game design. In: Visser J, Visser M (eds) Seeking understanding: the lifelong pursuit to build the scientific mind. Sense Publishers
- Giacomin J (2014) What is human centred design? *Design J* 17(4):606–623
- Grudin J (2012) A moving target: the evolution of HCI. In: Jacko JA (ed) *The human-computer interaction handbook: fundamentals, evolving technologies, and emerging applications*, 3rd edn, New York, NY, Taylor & Francis Group, pp 27–61
- Jaimes A, Gatica-Perez D, Sebe N, Huang TS (2007) Human-centered computing—toward a human revolution. *Computer* 40(5):30–34
- Koh JHL, Chai CS, Wong B, Hong H-Y (2015) *Design thinking for education: concepts and applications in teaching and learning*. Springer, Singapore, SG
- Kokotovich V (2008) Problem analysis and thinking tools: an empirical study of non-hierarchical mind mapping. *Des Stud* 29(1):49–69
- Novak JD, Cañas AJ (2008) The theory underlying concept maps and how to construct them. Technical Report IHMC CmapTools 2006-01 Rev 01-2008. Florida, FL
- Novak JD (2010) Learning creating and using knowledge: concept maps as facilitative tools in schools and corporations. *J E-Learn Knowled Soc* 6(3):21–30
- O’Cathain A, Thomas KJ (2004) Any other comments? Open questions on questionnaires—a bane or a bonus to research? *BMC Med Res Methodol*
- Oxman R (1999) Educating the designerly thinker. *Des Stud* 20(2):105–122
- Oxman R (2004) Think-maps: teaching design thinking in design education. *Des Stud* 25(1):63–91
- Taylor P (2000) Designerly thinking: what software methodology can learn from design theory. In: Gray J, Croll P (eds) *Proceedings international conference on software methods and tools, SMT 2000*. IEEE Computer Society, Los Alamitos, CA, pp 107–116

Chapter 11

Teaching Fuzzy Logic Utilising Innovative Approaches



Arjab Singh Khuman

Abstract Teaching within Higher Education often involves interacting with a variety of different types of students; different types of learners; all with varying types of aptitudes. This is further exemplified when considering modules that are shared between programmes, as each programme will often have specific thematic trajectories that provide for varying foundations of understanding. This chapter will highlight a particular case study involving the author and the module—IMAT3406: Fuzzy Logic and Knowledge Based Systems. Given the author’s recognition for sustained teaching excellence and use of innovative pedagogies, the case study will put forward a means to engage and enlighten a cohort comprised of significant variation. The teaching style and approaches adopted allow for a better understanding of core concepts. Making sure that *it makes sense to all*, ensures that the foundational knowledgebase needed from which to build upon is adequately in place, so that everyone in the cohort is on a level playing field. This can be achieved through dynamic teaching practices, often involving acclimation and assimilation to the cohort. Ensuring that a concrete understanding exists before the students are encouraged to undertake the assessment component, has proved to cater for exceptional output, not only in terms of detail, but both in quality and substance. Through tried and tested means, the case study used in this chapter sheds light on the attributes of a successful approach.

Keywords Inclusive · Pedagogy · Innovative

11.1 Introduction

Teaching with regards to Higher Education (HE) often does not involve a single track of execution. Teaching staff members are often encouraged to cater for the addition of students whom may not satisfy the general norm of an *expected* student, this may include students from neurodivergent backgrounds. There may be students with physical disabilities; students with underlying mental health needs; students whom

A. S. Khuman (✉)
De Montfort University, Leicester, England
e-mail: arjab.khuman@dmu.ac.uk

are not aware of issues they may currently be experiencing, and so on. Within these categories, there are many sub-categories, which splinter off into specific nuances and niches. Regardless of what issues are presented the learning objectives of the module must be attainable by all, ensuring equal parity. To facilitate this, teaching practices need to allow for a non-stringent adaptability, practices that are fully inclusive to all enrolled. Exclusion of a single learner due to circumstance is morally abhorrent as well as a blatant disregard to institutional policy and countless legalities. The qualities of a good practitioner of pedagogy should always have the ability to be dynamic and resourceful. Regardless of the context of the module, programme, course or any other instance of classroom based learning, a dynamic approach to teaching is beneficial for all. Being adaptable allows for absolute inclusion of all in the cohort, ensuring that the delivery of teaching objectives are understood, beyond that of surface learning.

The case study for this chapter will revolve around the assessment component of the IMAT3406 module. It is noteworthy to extend a mention that the author received all his degrees from the institute he now lectures for, De Montfort University—BSc, MSc & PhD. This is a key point, being a student to transitioning to post graduate study, to then transition to academic staff member, all the while at the same HE institute, provides for a unique perspective. It is precisely this perspective that has allowed for the IMAT3406 module to flourish in recent years, with consistent praise and recognition to the teaching practices being adopted. Being more aware of the culture and coupled with the fact that the author was a student on a variation of the module, has all played a part in shaping the module as it is seen today. The changes the author would have liked to have seen implemented as a student are the changes that he has implemented himself; practice what you preach, so to speak. This is the main reason why the module itself is so well received by so many, it is the embodiment of what the author would have liked to have received as a student himself, somewhat subjective, although extremely effective.

The successes of the module have also allowed for the teaching excellence associated to the module to be recognised. The author has received numerous Vice Chancellor's Distinguished Teaching Award (VCDTAs) [2018, 2019, 2021], and was awarded a Teacher Fellowship for 2022. Across a variety of metrics, student feedback and the student experience, the module is considered to be one of the best performing modules belonging to the faculty of Computing, Engineering and Media (CEM).

11.2 What Is Fuzzy Logic?

The module itself is best understood as being a mathematical means to model uncertainty. Fuzzy logic is ideally suited to modelling vague, abstract concepts such as linguistic information. To crisply define a notion like that of '*Tall*' is inherently difficult when only considering a crisp, classical understanding of membership. If one was to model the notion of *Tall* using conventional crisp means, it can be assumed that 6 foot would indeed be classified as *Tall*, and would absolutely, unequivocally belong to the set *Tall*. Fuzzy logic however, using the application of fuzzy sets, allows

for one to use a degree of membership, allowing one to quantify the belongingness to a particular set. A classical crisp set approach has only 2 possible outcomes; the observed object belongs to the set, or the observed object does not belong to the set and belongs to its complement. Black and white, up and down, left and right, very clear cut, no grey areas. It's either you are or it's either you're not.

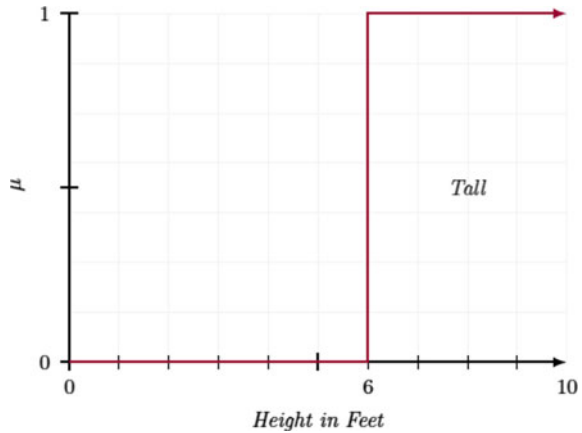
A fuzzy set allows for an observed object to have partial belongingness to a set, a degree of truth; a single object can belong to several sets to varying degrees. This is in contrast to the classical approach as we are no longer restricted to either a; yes it does belong, or a no it does not, we now have the option to see how much does it have an association. The rationale of fuzzy is that it allows for one to better model humanistic nature so that better computational models can be created, a more accurate model leads to better inference and better decision making based on said inference. The more detail one can capture foundationally, the more accurate the output will ultimately be.

Classical set theory makes use of Boolean logic (Boole 1847, 1854), whereby an object is classified as absolutely belonging, or absolutely not belonging to a particular set (Cantor 1895). The use of crisp boundaries applies an inherent level of strictness to what the set can model, instances where only two outcomes are allowed, such as an integer being either odd or even; such instances are easily handled using this classical approach. However, there is a need to encapsulate uncertainty that is associated to vagueness when considering human based perception. Human nature and inferencing does not work in such a precise and crisp manner, a humanistic approach needs to cater for the existence of imprecision based uncertainty, along with vagueness. The understanding of a set from a classical perspective is not a fitting synthesis for human intuition. The notion of *mereology* described by (Lesniewski 1929), considered the idea of an object being partially included in a set, this was the basis for the formulation of Max Black's vague set (Black 1937), created in the 1930s.

The building blocks of any fuzzy implementation involves the use of fuzzy sets, first proposed by (Zadeh 1965). A fuzzy set can be seen as an extension of the ideology of a vague set. From its inception fuzzy logic has been further expanded upon to establish itself as a powerful and successful paradigm for modelling uncertainty (Zadeh 1973). As logic is associated to propositions, fuzzy logic can be seen as the calculus of fuzzy propositions. Mathematical applications for precise reasoning will often need crisp understandings, however, this becomes problematic when concepts such as natural language are involved. Linguistic vernacular can be inherently vague, with a prevalent amount of ambiguity. Our daily existence will often be littered with varying degrees of uncertainty, further invoking various aspects of specific uncertainties (Zadeh 1999).

The use of a classical set for the modelling of unclassical behaviour will often fall foul when considering the vagueness of uncertainty. For example, the abstract concept of *Tall* cannot be universally defined, a single crisp value cannot be put forward as an indicative representation that is agreed upon by all. What is *Tall* to some may not be as *Tall* to others. Figure 11.1 provides a visualisation of what a typical crisp bounded set may look like. The plot in the figure describes any person being 6' or taller as a validated member of the set *Tall*. However, using this precise

Fig. 11.1 An example crisp set



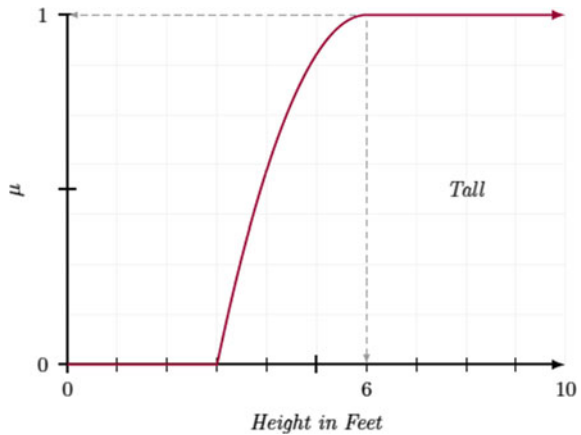
and strict definition, it would neglect any instance of anything less than 6'. It can be generally assumed that 6' is indeed *Tall*, but so too is 5' 11", at least to some extent. The only association one can attribute to a value is if it is included in the set *Tall* or not.

The problem now becomes one of determining the bounding of the set, to realistically encapsulate all common held assumptions of what satisfies the notion of being *Tall*. This echoes the sentiment of Sorites paradox, arising from a vague predicate. A single grain of sand does not constitute a heap, nor does two grains of sand. However, when we have a billion grains of sand, we then certainly do have a heap of sand, at what point do we transition from not being a heap, to becoming a heap? A fuzzy perspective will allow for a more forgiving approach, one that enables an object to have partial belongingness.

The most fundamental aspect of fuzzy set theory is its understanding of numbers. A fuzzy number is ideal for describing linguistic phenomena, where an exact description of its state is unknown. Fuzzy numbers were first introduced by (Zadeh 1975), for the purpose of approximating real numbers which deal with uncertainty and imprecision associated to quantities. It has great scope when approximating *height*, or *weight* and other such uncertain abstractions. The apex of a fuzzy number will generally be the only point where an object can be given a maximum degree of inclusion equal to 1. The varying degrees of membership for other objects will be indicative to their proximity to the apex. Fuzzy sets extend the notion of fuzzy numbers to allow for more versatility.

If one was to describe the set *Tall* as seen in Fig. 11.1, using a fuzzy approach, a possible visualisation may look like the plot contained in Fig. 11.2. In this plot, the inclusion of other possible values that could be deemed as *Tall*, values such as 5' 11" would also be included, but to a lesser degree when compared to that of 6'. Following this understanding, 5' 10" would also be a viable candidate for inclusion, but to a lesser degree than 5' 11", and so on. Using a fuzzy perspective for encapsulation, one is able to relax the expected strictness one would associate with a crisp set. Not only does a

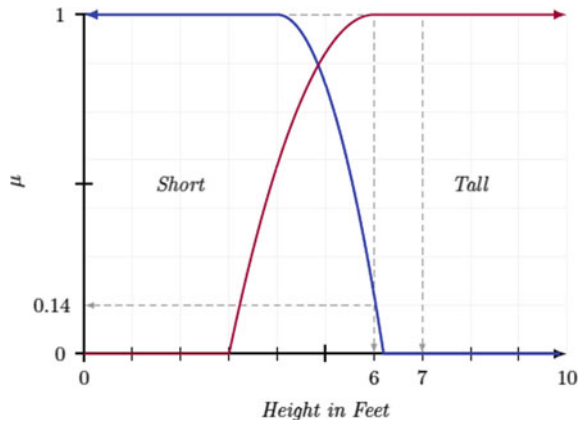
Fig. 11.2 An example fuzzy set



fuzzy set allow for a more harmonic understanding of uncertainty, but it is also able to fall back to a classical interpretation if need be. The degree of belongingness may be that of absoluteness, or absolutely not, in which case, a fuzzy set can replicate a crisp set (Klir and Folger 1998). In essence, the process of mapping a membership value to an object is known as *fuzzification*. It is only when considering that an object may have partial belongingness, does the strength and applicability of a fuzzy set become apparent.

A fuzzy set on its own can only allow for a certain amount of functionality, the combination of multiple fuzzy sets allows one to extensively model an abstract concept, that would otherwise be very difficult to represent using a crisp understanding. As it is a set of ordered pairs, the object (x) is associated to a degree of inclusion $\mu_A(x)$, the same (x) may belong to more than one set, and as such may be attributed to multiple degrees of inclusion. A fuzzy set goes against the traditional approach of classical set theory, by allowing an object to belong to different sets by varying degrees of membership. Such is the methodology of fuzzy sets, the law of the excluded middle and the law of contradiction are ignored. These two prevalent laws would stop an object from belonging to more than one set if a crisp perspective was used. Continuing with the notion of *Tall*, Fig. 11.3 demonstrates how using an additional fuzzy set, one can allow for a more humanistic approach in understanding the significance of any given value. This plot contains an additional set labelled *Short*. The value 6' in this instance can be seen to have an absolute degree of association to the set *Tall* with a returned degree of membership of 1, and a partial degree of inclusion to the set *Short* with a returned degree of membership of 0.14. If one inspects the object value 7', it can be seen to have absolute inclusion to the set *Tall*, and complete non-inclusion to the set *Short*. This logical assumption that being 7' would never be regarded as being *Short* can be easily catered for, as too can a plethora of other abstract notions. What has been presented is the foundational underpinnings that make up the fuzzy module, without delving deeper into the mechanics, this is what the students will have to grasp in order to make gains on their coursework.

Fig. 11.3 An example of multiple fuzzy sets



There is no prerequisite for the module, as the author presents the syllabus using the assumption that no one has been introduced to fuzzy before; which is often the case.

Fuzzy logic itself is well respected in the academic and research community, with many dedicated conferences and journals. The real world applicability of fuzzy has seen it be deployed in many different and varied industries, further reinforcing the effectiveness of such an approach. Given all this, the need to have a module dedicated to fuzzy is clear to see. From fuzzy one can delve into a multitude of different hybridised concepts and off-shoots (Khuman et al. 2015, 2016a, 2016b, 2016c). The author is a fuzzy researcher, so therefore the students have an academic expert on the subject. This pays dividends in that it allows for core concepts to be explained in a variety of different ways, as opposed to someone just reading off the slides or from a lab sheet. The author's expert knowledge and enthusiasm for the subject were highlighted in the feedback given by students on the module, this also constituted and contributed to being awarded a VCDTAs for 2018, 2019 and 2021.

The aforementioned foundational aspects of fuzzy logic are extended to explain how when piecing it altogether, one can then create a fuzzy inference system (FIS). The coursework component of the module involves each student creating their own FIS, in doing so, the student will appreciate the intricacies in articulating on their subjective understanding of their chosen application domain.

11.3 Teaching Practices

The lab sessions associated to the module utilise the use of MATLAB, which is a multi-paradigm numerical computing environment. A proprietary programming language developed by *MathWorks*. Contained within the install is the Fuzzy Logic Toolbox, which the students are allowed to experiment with to mock up prototypes (Grasha et al. 2000). The author anticipates that some of the cohort will be competent programmers and some may not, this does not affect the learning objectives.

Most of the students will have no experience with MATLAB, and for those that do, they appreciate the starting from the ground up approach, and see it somewhat as a refresher course on the software. Every lab session in the first weeks is spent becoming familiar with the software and gaining confidence with the fuzzy library, building up on previous sessions by understanding more functionality. By the end of the module every student will be able to code up in MATLAB and describe the technicalities of their coursework assessment; which is to create a fuzzy inference system of their choosing, while satisfying the minimal requirements for the configuration.

11.3.1 Teaching Practices—Lectures

It is the job of the lecturer to engage as much as possible, all the while to not be too overbearing. This highlights the importance of lecture material being made time specific in accordance to when it is to be delivered; it is advantageous to include content relative to the learning objectives, but not to overwhelm, as this can be counterintuitive. The current method of delivery is to ask the students in the lecture room, *‘if they have understood a particular aspect of the lecture material’*. If so, we proceed onto the next aspect, if not, we collectively repeat, presenting the notion using a different perspective. Adopting a collaborative learning approach evokes the ethos of—*‘we learn better, when we learn together’*.

The learning objectives of the module are a super-set of the learning objectives of each lecture. As such, there is more flexibility in arriving and satisfying the lecture objectives, in so doing, encompassing as much collective participation as possible. Given that this is a final year module, the students are more likely to have already formed a familiarity with the cohort. This can be used as an additional tool when incorporating various teaching practices and styles. The module itself, has many times, made use of the students in explaining core concepts vital to the understanding of fuzzy logic. Through a variation of peer-to-peer tutelage, the author encouraged the students who understand the concepts to explain it using their own words to the rest of the cohort. Sometimes hearing it through the words of a fellow student, allows for a more concrete understanding. This is not to imply that all aspects of the lectures are undertaken in this manner, but rather when the situation is called for, this only becomes apparent when the room is gauged. By incorporating student inspired explanations instils an obtainable quality, for the quality of the student doing the explanation can be extremely varied; from the stand-out high fliers, to the well-reserved and conservative. Seeing and hearing a fellow student explain can positively affect the learning curve of the room. It should be expected that a student will always be willing to explain to a potential population of roughly 170, they are never forced to, or made to feel uncomfortable. They do so using their own volition.

There are 3 distinct teaching styles that are adopted: visual, auditory, and kinaesthetic. These are not necessarily undertaken individually, but rather they are amalgamated. One can move quite quickly from a lecture slide to a group discussion highlighting a real world example of the lecture slide, to then demonstrate this through a

thought experiment. The free flowing and adaptable learning approaches do play dividends, as they allow for potentially more cohort participation. Not everyone learns and understands in the same way, being able to present and transcribe key concepts using varying teaching styles does benefit the learning quality of the group (Hsieh et al. 2011). By being able to connect what the students are learning to information they may already know, allows for them to fit new knowledge into their understanding of the topic. By allowing them to have the ‘eureka’ moment gives them more incentive to encourage future engagement and understanding. This is all possible, and enforced via good communication, the most essential quality when participating in a teaching environment. For the communication can be the difference in explaining something adequately well to get it, or explaining it very well to understand it and articulate from it.

A Universal Design for Learning (UDL) is adhered to, an approach to enhance learning and teaching for *all* our students. It provides a framework to identify and promote existing good practices across the institution, many of which already address the principles of UDL.

Good communication and trust between teachers and students is also important. Saying that you will get something done is different to actually getting it done. Maintaining a good rapport with students reduces the likelihood of insubordination, and therefore making the learning environment more enjoyable. Engaging in discussion needs to be done in such a way as to not come across as domineering, forcing perspectives onto students, even though that is technically what is happening. An understanding should be as harmonic as possible, involving the student as much as possible in the ‘journey’ of understanding for themselves. Connecting the dots for them is different from letting them connect the dots for themselves, as this takes away from their accomplishment.

11.3.2 Teaching Practices—The Labs

The labs for the IMAT3406 module involve the use of specialised software, in particular: MATLAB. This was chosen for several reasons; included as an additional installation package is the Fuzzy Logic Toolbox, which is utilised for quick prototyping. The toolbox is also needed for its incorporated function library which is used in the coursework when the students create their own fuzzy inference systems. MATLAB was also chosen as not many students have used it. This allows for the cohort to be on a level playing field, programming experience is not needed as the environment is unconventional. The labs are structured so that everyone is taught the necessary skills needed in order to properly be equipped for when they embark on assessment component.

The labs themselves are all 2 h lab sessions, having a 2 h lab allows for the learning objectives expected for each lecture, to be reinforced via lab work, for those that may have missed the lecture entirely. Those who did attend the lecture and also the lab, will benefit from a more concrete understanding. Having the lectures and

the lab material align somewhat, facilitates this aspect of learning. Making sure that everyone knows what is needed in order to understand what will be asked of them for when they start the assessment component of the module.

It is in the labs where the author is able to be more proactive as the cohort size is a maximum of 20 students. With a smaller population, and 2 h for each session, a more involved approach is adopted. Students are expected to follow along with the lab sheet as the author will be doing the same on the projector. Seeing the lab instructor also undertaking the lab exercises provides for a more inclusive environment because they are able to follow along, or do it on their own. The time is utilised very affectively, as the demonstration of the lab material and lab sheets do not take the entirety of the 2 h session up. What the author will do is to make sure that concepts from the lecture have been understood and if not, a brief recap is undertaken for the benefit of the room (Vogt and Rogalla 2009). This may involve everyone, or only the students who need it.

The author will also take the time to talk to each student individually to make sure they are comfortable with the lab exercises and the lecture content, as sometimes students may not feel compelled to raise their hand if they have a question or concern, so proactiveness on part of the lecturer should always be encouraged. As the lab groups are smaller; the attentiveness to which students are more vocal; which prefer to just get on with it and be left alone; which benefit from one-to-one, which benefit from small group discussions and so on becomes very apparent. It is playing to the strengths of the room that allow for a fruitful and rewarding lab session.

These are all aspects of dynamic teaching practices that the author adopts in presenting the fuzzy IMAT3406 module. A module that is consistently one of the best modules from the faculty it belongs in terms of metrics, attainment, feedback and the student experience.

11.4 Conclusion

There are many facets when considering dynamic teaching practices, the main objective of which should always be to foster for inclusivity, without creating an apartheid in the cohort where more energy and attention is given to a specific group. What has been put forward is a somewhat itemised list of adopted teaching practices.

A dynamic approach to teaching, one which fosters inclusivity will always fare well and be commended, as has been the case for IMAT3406: Fuzzy Logic and Knowledge Based Systems.

References

- Black M (1937) Vagueness. An exercise in logical analysis. *Philosophy of Sci* 427–455
Boole G (1847) *The mathematical analysis of logic*. Philosophical Library

- Boole G (1854) An investigation of the laws of thought: on which are founded the mathematical theories of logic and probabilities. Walton and Maberly
- Cantor G (1895) Beiträge zur Begründung der transfiniten Mengenlehre. *Mathematische Annalen* 481–512. <https://doi.org/10.1007/BF02124929>
- Grasha AF, Yangarber-Hicks N (2000) Integrating teaching styles and learning styles with instructional technology. *College Teaching* 2–10
- Hsieh S-W, Jang Y-R, Hwang G-J, Chen N-S (2011) Effects of teaching and learning styles on student's reflection levels for ubiquitous learning. *Comput Educ* 1194–1201
- Khuman AS, Yang Y, John R (2015) A significance measure for R-fuzzy sets. In: *Fuzzy systems (FUZZ-IEEE)*, 2015 IEEE international conference on, pp 1–6. <https://doi.org/10.1109/FUZZ-IEEE.2015.7337808>
- Khuman AS, Yang Y, John R (2016a) R-fuzzy sets and grey system theory. In: 2016 IEEE international conference on systems, man, and cybernetics
- Khuman AS, Yang Y, John R, Liu S (2016b) Quantification of perception clusters using R-fuzzy sets and grey analysis. In: 2016 international conference on grey systems and uncertainty analysis (GSUA2016b)
- Khuman AS, Yang Y, John R (2016c) Quantification of R-fuzzy sets. *Expert Syst with Appl* 374–387
- Klir GJ, Folger TA (1998) Fuzzy sets, uncertainty, and information. *Physica*
- Lesniewski S (1929) Grundzüge eines neuen Systems der Grundlagen der Mathematik. *Fundamenta Mathematicae* 1–81
- Vogt F, Rogalla M (2009) Developing adaptive teaching competency through coaching. *Teaching and Teacher Educ* 1051–1060
- Zadeh LA (1965) Fuzzy sets. *Inform Control* 338–353
- Zadeh LA (1973) Outline of a new approach to the analysis of complex systems and decision processes. *IEEE Trans Syst Man Cybernet* 28–44
- Zadeh LA (1975) The concept of a linguistic variable and its application to approximate reasoning—I. *Inform Sci* 199–249
- Zadeh LA (1999) From computing with numbers to computing with words. from manipulation of measurements to manipulation of perceptions. *IEEE Trans Circuits Syst I: Fundam Theory and Appl* 105–119

Chapter 12

Semi-automating the Marking of a Java Programming Portfolio Assessment: A Case Study from a UK Undergraduate Programme



Luke Attwood and Jenny Carter

Abstract Recent changes in Higher Education including larger numbers of students and larger staff student ratios mean that assessments need to be marked quickly and consistently, whilst also benefitting the students' learning experience. This paper initially introduces a portfolio assessment adopted three years ago on a software development module, with the aim of improving student engagement, and then goes on to discuss the inspiration behind attempting to automate the marking of this assessment. The paper then focuses on how the JUnit testing framework was used to achieve this, dissecting the challenges faced along the way, and how these were individually addressed. The end result was considered to be successful, with a significant reduction in marking time, and a guaranteed high consistency between markers. Students also benefitted through ongoing feedback from the unit tests they were provided during the assessment. In future, using such an approach provides the potential to assess students more frequently, giving them more regular feedback on their progress, and further helping them to engage with their studies.

Keywords Technology in education · Automatic marking · Unit testing · Java · JUnit

12.1 Introduction

OO Software Design and Development is a second year undergraduate module delivered on the BSc Computer Science and Software Engineering courses at De Montfort University (DMU) that teaches Object Oriented development principles using the Java programming language. For the 2016/17 academic year there were 170 students enrolled on the module, almost twice that of the previous academic year.

L. Attwood (✉)
De Montfort University, Leicester, England
e-mail: Lattwood@dmu.ac.uk

J. Carter
University of Huddersfield, Huddersfield, England
e-mail: j.carter@hud.ac.uk

Three years ago a portfolio assessment was adopted with the goal of encouraging students to further engage with lab material by completing a selection of questions across a variety of key topics. Ideally, we wanted to assess as many questions as possible, however, there was a trade off with what would be feasible to mark. This led to the idea of marking through sampling. Students were told that we would only mark one of their portfolio questions, but if they did not submit the one we chose to assess, then their mark would be capped at 40%. A further idea was to conclude the assessment with a lab test conducted under exam conditions, where students were asked to modify one of their portfolio questions before submitting it. This would help to an extent with overcoming potential issues relating to plagiarism, as a student would find it hard to modify a program that they had not fully written themselves.

The pass rate and average mark for students on this module has had a notable increase ever since this style of assessment was used (see Fig. 12.1), which somewhat proved our inclination that increasing engagement in the first term, where many fundamental principles are delivered, would help students in the module as a whole.

Whilst the assessment was clearly working, student feedback suggested they would appreciate a wider selection of their work to be marked. Due to the increasing number of students on the module and the general changes happening in Higher Education, as discussed in Sect. 2, this presented a challenge. Additionally, the British Computing Society (BCS) that accredits the associated courses updated

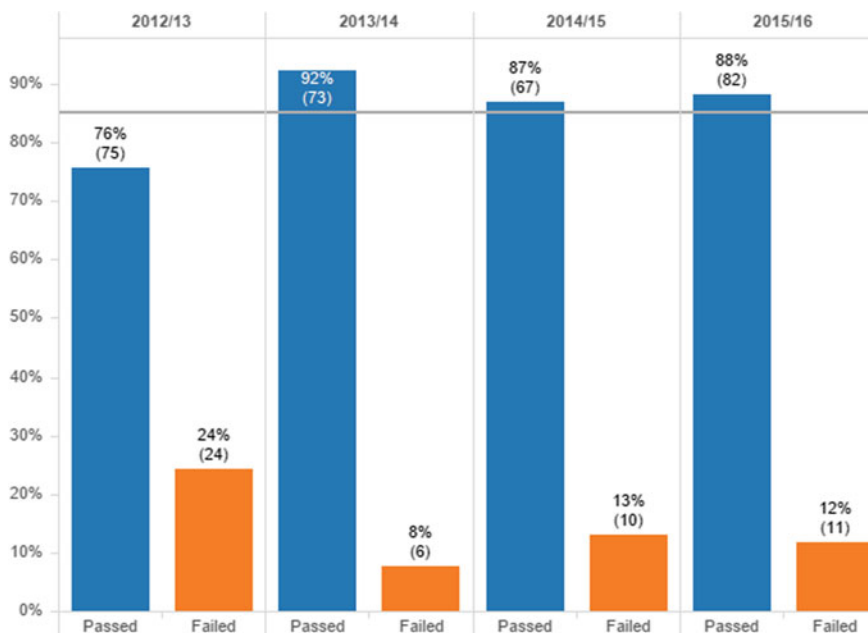


Fig. 12.1 Module pass rates and number of students (in parenthesis) over the past four years

their documentation with an emphasis on incorporating trustworthy software into the curriculum. One advised technique of delivery, as outlined in the PAS 754:2014 software trustworthiness specification (2014), was to adopt unit testing. As such, it was identified that unit testing could be used to semi-automate the marking of the portfolio assessment, integrating good practice, and also allowing more work to be marked. The target was to maintain the existing successful assessment structure whilst gaining the obvious benefits of automating the marking process and the production feedback.

12.2 Motivation

Higher Education has seen a number of changes in recent years. Notably, the NSS and the new TEF initiative have put a greater focus on learning and teaching methods and associated activities such as assessment. The HEA study 'A Marked Improvement', (Ball et al. 2012) identifies a number of issues specifically related to assessment. It emphasises how fee paying students have different expectations from their courses and associated assessments. Large numbers of students and larger staff student ratios mean that assessments need to be marked quickly, consistently, and the marks need to have integrity. The HEA report suggests better assessment practise will result in: "...greater confidence in academic standards and improved safeguarding of the reputation of UK higher education" (Ball et al. 2012).

The greater number and wider range of students also means that a variety of learning styles are evident and assessments need to cater for this variability. At DMU this has been addressed by adopting principles of Universal Design for Learning (UDL). This is an approach that encourages the design of learning materials and assessments that suit different learning styles and do not therefore need to be adapted to suit special needs (Al-Azawei et al. 2016). The UDL approach is supported by the UK's Quality Assurance Agency (QAA), in which the quality code for higher education has a series of indicators that reflect sound practice. Indicator ten states: "Through inclusive design wherever possible, and through individual reasonable adjustments wherever required, assessment tasks provide every student with an equal opportunity to demonstrate their achievement" (QAA 2013).

The HEA report (Ball et al. 2012) states that "The increasing size of student cohorts and a shrinking unit of resource mean that tutor time has become disproportionately spent on summative assessment. Students can be taught in larger groups, but each assignment or exam script still requires individual attention." This suggests the use of more formative assessments but we can also be creative about how we set, mark and give feedback on summative work. The report goes on to discuss how the use of technology enhanced approaches can help to accommodate diverse needs as well as enabling automated and/or timely feedback. This view is supported in Biggan (2010), which reports on a case study that uses automation to enhance the quality of feedback on assessments for a large module with 500 students. This article illustrates in particular that the marking was more consistent, the comments less

superficial, as well as being timely. The findings were validated by both student and external examiner feedback on the project outcomes. The Jisc report (JISC 2016) ‘Transforming Assessment and Feedback with Technology’ similarly supports the use of technologically enhanced approaches for marking and feedback, ensuring: “the work of each student is marked fairly” and “consistency of approach across different cohorts of students.”

The computer programming assignment in question at DMU was previously manually marked by tutors. The nature of checking code is such that it lends itself to automation, however, it also presents many challenges. The approach presented here provides a case study of a novel, largely automated means of marking and assessing all aspects of the portfolio assessment. The remaining sections consider how this was achieved, problems faced along the way, and the positive gains that have been made.

12.3 Semi-Automating the Portfolio Assessment

Programming assignments that are automatically marked are not new and this opportunity has been tackled in a variety of ways, over a number of years, within the computing discipline (Douce et al. 2005; Ala-Mutka 2005). It is possible to develop or utilise existing web-based automated assessment tools (English 2006), which often entail free-form data entry for part of a program (English and English 2015). As mentioned at the end of Sect. 1, we specifically wanted to expose students to an industry standard testing framework, and have them develop entire Java classes showcasing they understood standard conventions and key design principles. Therefore we wanted specific control over what and how we tested. It was decided that the emphasis would be on having students use prebuilt test cases as a means of informing their development (Janzen and Saiedian 2005) and verifying their classes had been designed correctly. This would help to instil a further level of critical thinking as supported in Rosen (2016). As the language being used was Java, it was decided that JUnit (also being used on a parallel module) would be most suitable. Furthermore, there were many existing cases of it being used effectively for automatic marking (Helmick 2007; Tremblay and Labonte 2003; Tremblay et al. 2008; Khalid 2013). JUnit is a testing framework for Java where individual *unit tests* are grouped inside of *test cases*, which can be executed individually or as part of a *test suite* that contains multiple test cases.

One decision was which, and how many, questions should be assessed. Previously, there had been 6 independent questions from 3 different topic areas—aggregation, interfaces, and inheritance. Utilising this approach would require independent test cases to be written for each of these questions, a potentially onerous task. It was decided that a better approach would be to have two portfolio scenarios: A (a `Register` class) and B (a `Player` class), for which there are each 3 questions. Each scenario would provide students with an initial question along with a test case (containing all unit tests) for the associated Java class, and then 2 further questions would act as extensions to this. Table 12.1 shows the different structure between

Table 12.1 Number of portfolio questions compared with those submitted for marking

| Year | Portfolio questions | Java classes | Test cases | Submitted questions |
|-------|---------------------|--------------|------------|---------------------|
| 15/16 | 6 | 6 | N/A | 1 |
| 16/17 | 6 | 2 | 2 | 3 |

this year and last, clearly highlighting how more questions would be submitted and assessed than before. Although only 2 classes were now used, the same design principles as before were introduced across the 6 portfolio questions.

As in Schmolitzky (2004), we also introduced the notion of Java *interfaces* at an early stage in the module but decided to not provide interfaces for the two classes linked to the portfolio work. Although this approach has been shown to be successful (Helmick 2007), we instead wanted students to interpret a UML specification in conjunction with the unit tests provided. As in previous years, the assessment criteria used were: 40% for their Java class design, 10% for a demonstration program, 20% for their documentation (i.e. `javadoc`), and 30% for their lab test modifications. The aim was to automate the marking of all but the `javadoc` (i.e. 80% of the assignment) by providing unit tests.

12.4 Constructing the Unit Tests

Normally it is considered good practice to have a single assertion per unit test (Aniche et al. 2013), however, in doing this a few issues became apparent. Some methods would require multiple unit tests, whilst others would only require one. Furthermore, some unit tests could pass without any code having been written. An example of this would be unit tests associated with testing that a Java `equals` method had been overridden correctly as the inherited version may pass one or more tests without the student having written any code. This issue can be overcome by using the Java *reflection* library (Forman and Forman 2004), which allows source code to be inspected at run time, however, this would result in having multiple asserts per unit test.

Having some unit tests passing prematurely may not normally be an issue, but the plan was to directly map the percentage of unit tests passed to the mark awarded for a given assessment criterion. Considering this, it also seemed most logical to avoid having a varying number of unit tests per method, as this would result in a higher weighting being attached to one method over another. Whilst it could be argued that some methods were more challenging than others, there was not a consistent ratio between this. It was therefore decided that the most sensible approach would be to write one unit test per method, and thus potentially have more than one assert statement per unit test. To overcome the ambiguity this may create, every assert statement had an associated unique message to ensure that students could clearly identify which assertion had caused a unit test to fail.

Table 12.2 Overview of the test cases, classes, and unit tests in a typical portfolio scenario

| Test case | Classes tested | Unit tests | Weighting (%) |
|------------------|--------------------------|------------|---------------|
| RegisterTest | Register | 15 | 40 |
| RegisterDemoTest | RegisterDemo | 1 | 10 |
| RegisterSubTest | Register and RegisterSub | 3 | 30 |

In previous years, students had been given credit for selecting the correct data types and access modifiers for their fields, e.g. `private` to ensure data encapsulation. Initially it was felt that this aspect of their design may have to be manually checked by the marker, but then it became apparent that the previously mentioned reflection library could be used to assess this. Whereas in other examples such as (Helmick 2007) reflection was used to assist the housekeeping of a custom-built testing framework, in this case it was used to assess the structure and quality of their code.

Whilst the main focus was on assessing students' ability to implement a class using a provided UML specification, the demonstration program had traditionally assessed that they could use their class, by creating object instances and invoking methods within the `main` method of a Java application. The issue was how this could be assessed with unit tests as whilst the `main` method could be executed by a unit test, it would be very difficult, if not impossible, to accurately assess if a student had used their class in a specific way. Instead, it was decided that a class containing a single static method would be used (e.g. `RegisterDemo`), similar in nature to one that students were used to, but instead with a string return type and a parameter list that accepted an object instance of the class being assessed.

In the past the 40-min lab test had involved students updating one of their portfolio questions under exam conditions. The goal was to also automate the marking of this component. A new test case (e.g. `RegisterSubTest`) was provided for this, which required them to further modify an existing class, and to extend it with a subclass. Table 12.2 shows a breakdown of the test cases for the `Register` portfolio scenario along with their associated number of unit tests and assessment weighting.

12.5 Avoiding Unintended and Hardcoded Solutions

When writing the unit tests it became apparent that it may be possible for students to pass a given unit test without following the intent of the lab exercise question. In particular, there was an issue with students being able to hardcode values initialised in their class, or returned from their methods, as they could see the expected result within the provided unit tests. One potential solution to this would be to have multiple asserts, each with different expected results, although theoretically this was still subject to a hardcoded attempt. A solution that was found, and used in some cases, was to take advantage of the `assertSame` method, which instead of assessing the

state of two objects (as `assertEquals` does) assesses two object references for equality, making it impossible to hardcode a value set or returned.

Another example of a method subject to hardcoding was one that had to search a `Register` object by using a given input parameter. If different input parameters were used then the student could ensure their search returned `true` or `false` accordingly, without actually searching through the inner collection. Therefore, multiple asserts were used, each with the same input parameter, but with the `Register` containing different data sets.

When assessing inheritance, even if a subclass was created, a student could simply update the superclass method and not override it. To overcome this, the unit test invoked the method on instances of both the superclass and subclass to check the corresponding behaviour differed. Other issues were overcome by again using reflection to ensure the superclass's field remained `private` and the subclass did not have any fields. Without these checks it would have been possible to bypass the public interface of the superclass by either accessing the superclass's field directly or by working on a duplicate mirrored field in the subclass.

12.6 Marking and Feedback

Whilst a JUnit test suite could have been used to execute each test case, it was instead decided to create a test runner that would allow a finer level of control over how the results were summarised, and additionally provide a means of giving feedback for the student's `javadoc` comments. A method was defined that accepts a JUnit test case and an associated weighting (e.g. `RegisterTest` and 40), and then produces a statistical summary of the total number of unit tests passed and failed, and their mark for that component. If 14 of the 15 unit tests passed, a rounded mark of 37/40 would be shown. Using the `org.junit.runner.Result` class, a failure overview was then provided by iterating through any failures and recording details of these.

So that all feedback could be inside a single file, a rubric-like set of predefined comments were defined relating to the quality of `javadoc`. Tutors would simply need to assess students' documentation before executing the test runner, and then be prompted to select one of these. The mark and comment could then be merged with those associated with the unit tests and uploaded for students to view. This would usually be in excess of a page of A4, without a single comment having to be written.

There were a total of three tutors on the module, one of whom was inexperienced, and ensuring consistency between markers is usually a challenge. As 80% of this assignment was marked automatically, this was now largely a nonissue. The statistics presented in Table 12.3 show even though the cohort size has nearly doubled, and the number of questions marked increased, the marking time has been significantly reduced (* measured via a sample of 20 students with an experienced marker). The mean and standard deviation have remained in a similar band to previous years.

Table 12.3 Comparison of this and last year's portfolio assessment

| Year | Cohort | Mean (%) | STDEV (%) | Questions assessed | Marking time per student * |
|-------|--------|----------|-----------|--------------------|----------------------------|
| 15/16 | 93 | 64 | 18 | 1 | 12 min 15 s |
| 16/17 | 170 | 62 | 20 | 3 | 1 min 45 s |

12.7 Conclusion

The evolution of the original portfolio assessment to incorporate unit testing has clearly been a success. Some obvious benefits include a significant reduction in marking time and guaranteed high consistency between markers, both of which are very important factors in the design of an assessment when student numbers and the number of markers are both increasing. This also allows the potential to assess students more frequently, giving them more regular feedback on their progress, and further helping them to engage with their studies.

There was a larger amount of source code assessed than in previous years, and this could very easily grow. As well as further exposing students to the importance of unit testing (as advocated by “Software Trustworthiness (2014)), they also benefitted from ongoing feedback during their assignment based on the number of tests they had passed at any given time.

Many of the difficulties came in the construction of the unit tests, particularly in ensuring that tests cannot be passed in unforeseen or unintended ways. There were a variety of techniques deployed to overcome these hurdles, and the Java reflection library was particularly helpful. In the future, having different input data sets used for the unit tests the students are provided during their assessment, and those that are actually used to mark their work, may further help combat this. Ultimately a lot of the work involved in the construction of this assessment was frontloaded and the challenge now is to see how easily this approach could be adapted across different topics or even onto other software development modules.

References

- Ala-Mutka K (2005) A survey of automated assessment approaches for programming assignments. *Comput Sci Educ* 15(2):83–102
- Al-Azawei A, Serenelli F, Lundqvist K (2016) Universal design for learning (UDL): a content analysis of peer reviewed journals from 2012 to 2015. *J Scholarship of Teaching and Learning* 16(3):39–56
- Aniche M, Oliva GA, Gerosa MA (2013) “What do the asserts in a unit test tell us about code quality? A study on open source and industrial projects”. In: 17th European conference on software maintenance and reengineering (CSMR 2013)
- Ball S, Bew C, Bloxham S, Brown S, Kleiman P, May H, McDowell L, Morris E, Orr S, Payne E, Price M, Rust C, Smith B, Waterfield J (2012) A marked improvement: transforming assessment in higher education. The HEA

- Biggan J (2010) Using automated assessment feedback to enhance the quality of student learning in universities: a case study. In: *Technology enhanced learning. quality of teaching and educational reform. Communications in computer and information science*, vol 73. Springer, Berlin, Heidelberg
- Douce C, Livingstone D, Orwell J (2005) Automatic test-based assessment of programming : a review. *J Educ Resour Comput* 5(3)
- English J (2006) The Checkpoint automated assessment system. *Proc E-Learn 2006*:2780–2787
- English J, English T (2015) Experiences of using automated assessment in computer science courses. *J Inform Technol Educ: Innov Practice* 14:237–254
- Forman IR, Forman N (2004) In: *Java reflection in action*. Manning Publications
- Helmick M (2007) Interface-based programming assignments and automatic grading of Java programs. In: *ITICSE '07: Proceedings of the 12th annual conference on Innovation and technology in computer science education*, Dundee, Scotland, 2007, ACM Press
- Janzen D, Saiedian H (2005) (2005) Test-driven development: concepts, taxonomy, and future direction. *IEEE Comput* 38(9):43–50
- JISC (2016) *Transforming Assessment and Feedback with Technology* [online]
- Khalid A (2013) Automatic assessment of Java code. *The Maldives National J Res* 1(1):7–32
- PAS 754:2014 (2014) *Software trustworthiness. Governance and management. specification*. British Standards Institution, May 2014
- QAA (2013) *UK quality code for higher education—chapter B6: assessment of students and the recognition of prior learning*
- Rosen CCH (2016) A philosophical comparison of chinese and european models of computer science education. (A Discussion Paper). In: *Software engineering education going agile (CEISEE 2015)*, pp 9–14
- Schmolitzky A (2004) “Objects first, interfaces next” or interfaces before inheritance. In: *OOPSLA '04: companion to the 19th annual ACM SIGPLAN conference on object-oriented programming systems, languages, and applications*, pp 64–67
- Tremblay G, Labonte E (2003) Semi-automatic marking of Java programs using junit. In: *International conference on education and information systems: technologies and applications (EISTA '03)*, pp 42–47
- Tremblay G, Guerin F, Pons A, Salah A (2008) Oto, a generic and extensible tool for marking programming assignments. *Softw: Practice and Exper* 38(3):307–333

Part III
Employability and Group Work

Chapter 13

The Enterprise Showcase Experience



Gary Allen, Mike Mavromihales, and Jenny Carter

Abstract The School of Computing and Engineering at the University of Huddersfield have, for several years, been experimenting with an Enterprise Showcase Event, in which multidisciplinary teams of students work intensively on real-world projects for local companies, and then present their work at a day long trade-show style event. This paper outlines the rationale for the project, explains how the event operates, discusses the advantages of the approach over more traditional kinds of team project, and analyses the student feedback.

Keywords Group work · Team work · Enterprise showcase · Employability · Multi-disciplinary · Collaboration

13.1 Introduction

The School of Computing and Engineering at the University of Huddersfield is made up of two departments, Computer Science and Engineering. Students in both departments engage in group work at various levels within their respective courses, but the students do not normally engage in collaborative, cross-departmental project work. The project work undertaken is mainly group work in years 1 and 2, followed by an individual honours project in the final year. Students on an Integrated Masters degree (MEng, MComp, or MSci) also undertake a group project in their additional Integrated Masters year. There is a rich body of educational literature in support of group work, and it is widely accepted that group work offers many advantages to the students. In the computing discipline, group work is seen as good for interpersonal skills including team working, project management, and presentation skills, as well as for the development of technical skills such as requirements capture, system design, coding, testing and evaluation. In the engineering discipline, students

G. Allen · M. Mavromihales (✉) · J. Carter
The University of Huddersfield, Huddersfield, England
e-mail: m.mavromihales@hud.ac.uk

G. Allen
e-mail: g.allen@hud.ac.uk

are encouraged to undertake group work from an early part in their undergraduate studies through several modules. This helps them develop interpersonal skills which are greatly desirable by prospective future employers. Through discipline, organisation and effective communication students learn from each other in modules such as professional development in which they research and present on a topical issue such as sustainable transport. In a mechanical design module, students will collaborate to develop a conceptual design prior to developing various aspects of it to a detailed stage. The enterprise showcase event, for the mechanical and electrical engineering students is linked to the intermediate, year 2 module of manufacturing and enterprise. The significance of the showcase event is that, post event, the students are expected to complete a business plan which outlines vital information that is required in order to bring the product to market. By this stage they have already been made aware of the commercial constraints associated with product development by partaking in the showcase event. Although teamwork in undergraduate studies is not entirely new, what is unusual in this case is the operation of multi-disciplinary teams. This has allowed for learning from fellow students with disparate areas of interest, expertise within the technology sector. This emulates exactly what they are likely to encounter in a real world working environment in which they gain an appreciation of the required expertise for a balanced product development. Effective collaborative problem solving and its benefits has been documented by many educational researchers (Nelson 1999) as has learning by doing (Schank et al. 1999). Group work is also a required component of many of our courses, by the British Computer Society (BCS) (Projects and Group Work 2018) (Guidelines on Course Accreditation 2018) for accreditation of our degrees, The Institute of Mechanical Engineers (I.Mech.E) and The Institute of Engineering and Technology (I.E.T) as well as our industry partners who require graduates with soft skills and capable of working in teams. Both Engineering Institutions who are the main governing bodies for our accredited engineering courses apply the UK Standard for Professional Competence (UK-SPEC). The standard can be found at the following link:

<http://www.engc.org.uk/engdocuments/internet/Website/UK-SPEC%20third%20edition%20%281%29.pdf>

The UK-SPEC is based on the demonstration of key competences and is the UK Standard for Professional Engineering Competence. It describes the Science and mathematics, Engineering analysis, Design, Engineering Practice competences in the economic, legal and social, ethical and environmental context, that have to be met in order to attain Engineer status at either Technician, Incorporated or Chartered level.

Over several academic years the School has been experimenting with an Enterprise Showcase Event, whereby students from both departments have been placed into multidisciplinary teams and set to work on real world problems put to them by our industrial collaborators. The event takes place in an intensive one week block and culminates in a trade-show style Showcase Event where the students present their work to both academics and to the industrial collaborators. In this paper we outline the motivation for the showcase event (Sect. 13.2), explain how the event operates (Sect. 13.3), analyse the reaction and feedback from our students (Sect. 13.4), and

outline several ideas for how we might improve the event in future years (Sect. 13.5). Furthermore, the paper reports on revised practice which has been introduced as part of stepped changes that were initially enforced due to the 2020 Covid-19 pandemic and remote working.

13.2 Limitations of Existing Project Work

Current approaches to team based project work suffer from several limitations and weaknesses. Timetabling restrictions make it difficult or impossible to schedule interdisciplinary groups, so groups tend to be drawn from one course, one course suite, or at best one subject area. This means that the students are rarely, if ever, expected to work outside of their comfort zone in terms of working with students they don't know or working with students from different courses who bring a wide range of unfamiliar knowledge and skills. Additionally, when creating project ideas for students within a single discipline, the tendency is to come up with project ideas that fit the skill set of those students. This potentially stifles creativity and reduces the range of opportunities for learning afforded to these students. If we give the same problem to all students or groups within a module then there is little variation to keep the students interested and there is potential for collusion and plagiarism. If, on the other hand, we let students come up with the problems themselves then there tends to be a lot of variation in the level of complexity, making it difficult to keep the learning experience equitable and the marking fair. Students are unaware of what they don't know, so they tend not to consider projects outside their comfort zone. Similarly, students often have limited commercial and industrial experience, so are unable to easily consider realistic problems. Collectively considered, these issues often mean that the project work undertaken in team project modules is rather repetitive, unoriginal and unimaginative and does not provide the students with a realistic experience of working on commercial projects. An additional concern is that long time-scales (typically these modules are yearlong, starting in September/October and due for submission in the following May) often implies that students postpone assignment completion until urgency hits thus producing rushed and sub-standard work at a late stage. These projects do not reflect the "need it yesterday" urgency of many real commercial situations. Taken together these problems led us to reconsider how team based project work, combined with activity based learning (ABL) should operate, and led to the introduction of the Enterprise Showcase Event.

13.3 The Enterprise Showcase Event

13.3.1 *Timetabling and Assessment of the Event*

The University of Huddersfield introduced a “Consolidation Week” into the timetable several years ago. Consolidation Week is the first week after the Christmas vacation, when normal timetables are suspended to allow other activities to take place, such as in-class tests, catch-up and revision and practical sessions before the start of term 2 teaching, and short one-off events. The University has been keen to find interesting and innovative ways to utilise this time, and offered the opportunity to bid for small amounts of funding via the university’s Teaching and Learning Institute (TALI) to support such projects. Our Enterprise Showcase Event has, for several years, taken advantage of this Consolidation Week. Alternatively, Problem Based Learning (PBL) activities are also run outside the normal teaching term, in Spring. This offers the opportunity for learners to reinforce curriculum knowledge gained during teaching whilst also honing their soft skills. Normal classes do not operate in these weeks therefore this gives us the opportunity to timetable events across subject areas or (as in this case) across the departments within the school. This allows us to create interdisciplinary teams made up of (almost¹) all students in the Computer Science and Engineering departments. This includes students of Automotive Engineering, Electrical Engineering, Mechanical Engineering, Computing Science, Software Engineering, Computing, Information Systems, ICT, and Web Technology can all be mixed together within the project teams.

All of the students involved in the Enterprise Showcase Event are already enrolled on yearlong projects modules. This applies for all disciplines and provides a relatively simple way to build the assessment of the Enterprise Showcase Event into the curriculum. The event has been added as an additional assessment within those project-based modules (a sub-element of the main in-course assessment, the Enterprise Project). In the Computer Science department the students are all registered on module CII2350 Team Project. The Enterprise Showcase Event forms a sub-element of the module coursework assessment. Similarly, in the Engineering department the students are registered on either module code NIM2220, known as Manufacture and Enterprise or either NIE2208, known as Enterprise: Electronic Product Design and Manufacture. The former module is for undergraduate engineers studying either Mechanical, Automotive or Energy engineering. The latter module is for undergraduate Electrical or Electronic engineering students. All modules stipulate pre-requisite study that is more technically focussed. The modules associated with Enterprise focus on the commercial aspects of technological products and the soft skills required to develop these. Teamwork and inter-disciplinary awareness are therefore of the essence and is therefore why the Enterprise Showcase Event has been added as a sub-element of assessment within those modules.

¹ Students on the Computer Games Design and Computer Games Programming courses do not take part as they already have a range of activities taking place in Consolidation Week.

For electrical and electronic engineering students this to provide an introduction to business, finance, marketing, engineering management and design for manufacture (DFM) in the context of electronic product design and manufacturing. It is intended to promote an understanding of the lifecycle process of product design and develop the skills required by professional engineers to play an active role in that process.

13.3.2 Clients and Commercial Partners

One of the key objectives of the Enterprise Showcase Event has been to involve practitioner clients providing existing actual problems for the students to work on, in order to make the event as realistic as possible. The problems must have genuine commercial application, and the client companies are encouraged to work with the students with the best ideas after the showcase event in order to explore opportunities for further development and for commercial exploitation. The opportunity to develop a real commercial product as a result of the Showcase Event should act as a motivator both for the client companies and for the students themselves.

In the first year of operation one of our client companies, was a leading Health Care provider. They outlined the problems caused within the NHS and private health care by bed sores and ulcers resulting from prolonged periods of being bedbound. The students were asked to think of innovative ways to minimise bed sores. Some groups focussed on high-technology solutions, such as sensors, monitors, and alarms; some turned to the use of novel materials or shapes and profiles for mattress design, or the use of springs within the mattress to keep a patient moving; some researched the impact of air-flow on bed sore formation and suggested fans or mattress covers designed to minimise moisture; some groups explored the role of Big Data in collecting as much information as possible via a range of sensors (such as temperature, pressure, humidity, and time intervals between the patient being moved) so as to build better models of the conditions that cause pressure ulcers; while some teams identified low-technology solutions such as simple timers to notify nursing staff that it is time to move the patient. The wide range of solutions suggested by the student groups impressed the company, who found several of the ideas worth further investigation. Similar transferrable technologies utilising sensors for monitoring could also be applied to other problems, such as monitoring the internal living environment of a modern home and adjusting for human comfort (based on warmth in occupied rooms, humidity and oxygen levels).

One of the criticisms of the initial event was that by providing only one client with one problem we did not provide options to the students. In the second year it was therefore decided to seek several companies who could each pitch their problem to the students, thereby giving the students a choice of problems to work on, and making the event more appealing. Three projects were offered, two from local companies and one from the university's Student's Union. In summary these were:

- A local construction sector company presented a problem based on optimising the means by which roofing A-Frames are accommodated on a flatbed truck for delivery to a construction site, in a safe and efficient manner. Each set of A-Frame trusses is unique to the project to which it is intended. According to the company there is no off-the-shelf solution currently available for this problem. By failing to utilise the entire space on the truck, a second truck would be required to be booked through a haulier, at considerable cost.
- A local software house with a passion for encouraging children from schools in the surrounding area to get involved with STEM (Science, Technology, Engineering, and Maths) subjects and in particular to get them writing computer code. The brief here was very general, to come up with ideas and associated products that could be used with the children to inspire them to get involved with STEM subjects, code clubs, and “hackathons”.
- A student society from the Huddersfield University Student’s Union called Enactus had presented. Enactus is a global student movement aiming to “use the power of entrepreneurial action to transform lives and shape a better more sustainable world” (Enactus—2018). The Huddersfield Enactus group had taken ownership of a rundown greenhouse on an area of land owned by the university. They were looking for ways to utilise the greenhouse in a cost-efficient way to bring benefit to as many people as possible. Their pitch asked the students to identify low cost ways to support this objective, with a particular focus on energy efficiency, sustainability whilst also ensuring a secure site.

All of these clients were able to provide opportunities for further work and for the potential commercial development of the ideas generated. The students were happy to be given the range of projects to choose from. All of the clients were available on the first day to allow the students to discuss the projects, and to carry out some requirements capture. This alone, is an important part of the Enterprise Showcase Event, as students working on projects do not normally get the opportunity to perform requirements capture from real clients, nor do they often get to work on projects with such uncertain and volatile requirements. This is an important part of the ‘real-world’ experience which the Enterprise Showcase Event is designed to provide.

We again received a wide range of suggestions for each of the project ideas. These included:

- Use of 3D modelling software to help with the A-Frame loading problem. Some teams found open source software that could be enhanced or tailored towards the specific problem, while others set about developing prototypes of bespoke software built to solve the problem. One team suggested an Artificial Intelligence (AI) solution based on machine learning.
- Many varied solutions were suggested for the STEM project. These included 3D printed cogs and wheels to allow students to experiment with basic engineering; software development platforms aimed specifically at younger children to allow drag-and-drop development of code; and 3D printed build-it-yourself car kits and associated mobile apps to control the finished car, allowing children to experience both the physical building of the car and the software side of controlling it.

- greenhouse monitoring and control systems based around Raspberry Pi or Arduino devices, with attached sensors and motors to automatically maintain ideal conditions through automated watering and opening and closing of vents; intruder detection systems; and remote monitoring coupled with mobile apps to allow the greenhouse to be operated remotely.

The clients were all impressed by the range and diversity of the suggestions, as well as the level of technical detail achieved by many of the teams.

13.3.3 Organisation and Operation of the Event

The Showcase Event is designed to run in a short, intensive time-scale of one working week. The students are placed into groups in advance of the start of the event. Groups are announced at the outset. The students arrange themselves into their groups, and the industrial partners then immediately pitch their problems to the teams. The industrial partners then remain on site during the first day (for at least half the day) to enable the students to ask questions, discuss the problem and any initial ideas they may have for potential solutions, and to carry out some requirements capture. Students are then organise themselves in order of tasks and priorities in order to meet the required outcomes by the end of the week. Initially they are expected to brainstorm ideas and potential solutions, carry out research in the university library or on-line, and to develop their ideas.

The first deadline the students are expected to meet is to submit an electronic copy of an A2 poster to showcase their ideas. This is required during the second half of the week. The poster should visually represent the group's idea or proposal, and will be displayed at the Showcase Event at the end of the week. Note that this is not a formal assessment point. If teams successfully submit by the deadline then the department will arrange and pay for the printing of the poster ready for the Showcase Event. Any team that fails to submit their poster by the deadline will be required to arrange and pay for printing themselves. This 'soft' deadline is in part intended as an encouragement to the students to start work as quickly as possible and make initial progress.

As well as working on the poster, the teams are also required to start work on their prototype product. For this we have available a range of kit that the students can sign out and borrow for use on the project. The kit includes Raspberry Pis, Arduinos, breadboards, a range of sensors such as temperature and humidity measures, motion sensors, and cameras, indicators and LCD displays. 3D printing facilities are also available. When designing artefacts for 3D printing, the teams must ensure that the print time is limited to one hour maximum, and that their designs are suitable for the size of printer available. The necessary details are included in a briefing pack given to each team at the start of the week. There is also a small budget available, so the students can reclaim up to £20 per team if they need to purchase any specific equipment which we are unable to provide. For this receipts must be presented,

along with a description of the use of the item. Only one member per team can claim back the expenses. The teams have the rest of the Wednesday and all day Thursday to work on their prototype product in preparation for the Showcase Event on the Friday. Students are therefore required to work to constraints and learn by doing (Schank et al. 1999). The learning benefits of undertaking Problem-Based tasks have been well documented (Barrows 1985).

To ensure fair access to support for all teams there are drop-in advice sessions scheduled throughout the week, where the students can consult academic staff. These sessions are supported by Computing and Engineering academics, who will help the students by providing advice and guidance, suggesting changes or improvements to an idea, directing the students to relevant resources, or whatever other help they may deem appropriate. The details of these sessions, which run for an hour per day on the Tuesday, Wednesday, and Thursday, are again included in a briefing pack provided to each team.

The Showcase Event itself takes place in a large room organised along the lines of a trade fair. Each team has a pre-allocated stand. The posters have been printed and are ready to be displayed. The students bring their prototype product and are given time to set up and prepare for the event. Academic staff and industrial partners visit stands, observing and scrutinising posters and prototypes, listen as the students pitch their ideas, question the students, and then give immediate verbal feedback. The Showcase Score Card used for the event is included in Appendix A. The final mark awarded to each team is the average of the best three marks on the day. In the first year that the event operated each member of academic staff was asked to walk around and look at as many projects as possible. This did cause some problems, as the teams at the front of the room found it much easier to get three academics to mark their work than the groups towards the back. To improve this, in the second year of operation each academic was randomly allocated five teams that they were required to visit and assess, and were then encouraged to walk around and look at as many more teams as possible. In this way every team was guaranteed a visit by at least three academics in order to ensure the marking could be completed. Another change in the second iteration of the event was to ensure that the academics could not see the marks awarded earlier in the day by their colleagues. This was to ensure each mark was awarded entirely impartially, and staff were not swayed by the grades awarded by their colleagues. One of the interesting things to emerge was the way in which the teams were able to take on board the feedback given by the academic markers throughout the day to improve their pitch and therefore improve the marks received as the day progressed. Many teams commented on how they really welcomed the immediate feedback and the opportunity to use this to refine their pitch (and in some cases the idea itself) so as to maximise their grade.

As with any group work, the students are encouraged to play to their strengths, and so to give team members the roles that they can best fulfil, be it development, documentation, or presenting the idea. Some teams do this well, and ensure that the idea is pitched in a positive light by a team member with good presentation and marketing skills. It can be a valuable lesson to some students to realise that a good idea described badly can accrue a lower grade than an average idea pitched in a positive

and enthusiastic manner. However, as can be seen from Appendix A, the academic assessors are required to consider Research, Innovation, Manufacture/Execution, Marketing, Costing, Commercial Potential, and Teamwork. The feedback provided and the grades awarded should therefore summarise all of these areas, and should not be too heavily swayed by the presentation skills of the team members.

13.3.4 Outstanding Issues

As with any new idea or novel event there have been some issues which we need to address going forward. These include: communication of team allocations and other required information; dealing with students who fail to attend or contribute to their groups; the layout of the room for the Showcase Event; and ensuring the industrial partners know which teams are working on their projects. Here we briefly address each of these concerns:

- **Communication:** Some students complained that the team allocations were not made clear, or that students arriving just a few minutes late on the Monday morning were unsure which team they had been allocated to. We feel that it would not be appropriate to announce the team memberships before the Monday morning, as we want to ensure all teams have a similar experience, and we do not want students to get together in advance. Part of the idea is that the students are working with people they do not know, so we feel it is sensible to keep the team memberships secret until the event begins. We do, however, need to make sure the team allocations are complete before the event begins, and that the information is ready to be given out at the outset of the event.
- **Students who don't engage:** Thankfully the overwhelming majority of students did participate in the event. However, as with any student activity, there are always a few who do not. Progressively, moving forward we experimented with a reporting system to allow groups to notify the project coordinator of any students who failed to engage. The coordinator then contacted these students to inform them that, if they did not take part in the event, they would receive a mark of 0% for this component of their relevant module. This approach did help to bring a few students into the event, but did not address the issue of students attending but making little real contribution. The future adaptation of Peer Group Assessment is likely, whereby the members of each team rank the contribution of their peers at the end of the project, as this is our established vehicle for managing group work. The perceived difficulty, and the reason we did not try this before, is that the interdisciplinary nature of the teams makes it difficult for the team members to get together after the Showcase Event has finished. We will therefore have to ensure that it is done on the Friday before the event closes.
- **Layout of the room:** We need to ensure that no team is 'hidden in a corner' or feels that their location at the Showcase Event puts them at a disadvantage when it comes to attracting academics to evaluate their work. This was much better the

second time around, when academics were allocated teams to see, as it ensured that all teams saw a minimum of three assessors. There were also some issues where teams needed access to mains power for their prototype devices, which led to power extension cables being laid across the floor. We need to think carefully about this in order to find an ideal layout for future years.

- **Industrial Partners:** The industrial partners attend the Showcase Event on the Friday to see the students' work, discuss it, and provide feedback. This year, with three client companies involved, we realised too late that we were unable to tell each company which teams had elected to work on their particular problem. The representatives of the companies therefore had to walk around and identify relevant teams for themselves. This had not been a problem the year before, as we only had one company involved. Next year we will use the Poster Submission on the Wednesday to collect data about which team is working on which project, so that we can then provide a list of relevant teams to each company.

13.4 A Paradigm Shift in Practice Due to the Pandemic

The Enterprise Showcase Experience had proceeded outside of the teaching terms, during the Summer term, whilst under the pandemic lockdown but with revised practices.

Since the start of lockdown, teaching and learning was facilitated online and therefor a fundamental shift in practice was necessary for the hosting of the Enterprise Showcase Experience. Whilst we academic staff and learners initially resorted to the use of Zoom as an online live communication tool. Microsoft Teams was later widely adopted and became the standard means for group discussions and forums throughout the University. It has remained so since, resulting in permanent changes in Teaching and Learning practice as well as course management and administration.

Microsoft Teams was launched in 2016 as a way of combining voice, chat, file sharing and video. Although its use would no doubt have spread and developed naturally, the COVID-19 pandemic caused an explosion in the need for such systems and Teams is now used very widely. Various other platforms were used to enable meetings, deliver teaching material and so on, Zoom being one of the more popular, but as the last two years have passed Teams seems to have become the more commonly used tool. New features were added in 2020 and 2021 to provide comprehensive collaboration features.

The pandemic accelerated a move to remote working for many workplaces and organisations. This change was happening anyway but at a slower pace. The same can be said for HE teaching; with increased costs of educational fees and the expense of moving away to study, courses were more frequently being offered in distance learning study mode. Also, the use of blended learning was increasing as more tools that supported it became available. However, the pandemic forced an almost

overnight shift to the use of such approaches. The resultant changes to both workplaces and learning institutions would have no doubt happened eventually but they came about much more quickly.

A study by Shalin Hai-Jew (2020) showed how Teams is very useful as an enhancement to teaching provision in areas of Higher Education. Examples included short courses, project-based learning, creating a virtual learning community and for the sharing and critiquing of designs. They also concluded that although it would not be enough to replace their Learner Management System, it complemented it and acted as a tool providing additional features.

Similarly, a comparison of Moodle and Teams in (Krašna et al. 2020) shows that although Moodle is still more appropriate for many traditional elements of course management (such as assessment, grade recording, quizzes etc.), “MS Teams excels in different types of communications (chat, videoconferencing, lecturing, screen sharing), and is superior in general content creation.”

Some key features of Teams include its excellent security facilities and from a teaching perspective, the flexible ways in which collaboration can take place. For example, Teams, Groups, Breakout Rooms, Chat to name a few (<https://amaxra.com/articles/7-things-you-should-know-about-microsoft-teams>). Similarly, virtual meetings and video calls are very easy to set up and work in a reliable way. It is the ability to enable group work remotely in a very easily manageable way that makes Teams the best choice for delivery of parts or all of many HE modules. Teamwork is an essential and necessary component in teaching as it emulates the way in which projects will be carried out and managed in the workplace. The ability to be able to do this as easily or even more easily than it is face to face, is a highly attractive feature.

The changes that were initially introduced as mainstream practice during the pandemic lockdown have now led to a permanent transition of hybrid delivery. Lectures for prior learning leading to the Enterprise Showcase Activity are pre-recorded into ‘bitesize’ delivery duration and are backed with physical, on-campus support tutorials. When it comes to the Showcase Activity, students are able to meet in virtual rooms within Microsoft Teams to progress the project. They are also able to meet in person on campus for support from academic and technical staff during allocated timeslots. Physical building of prototypes is enabled in this manner. Although posters were showcased in virtual rooms during the pandemic lockdown period, this event has reverted to a large physical showcase event with all the anticipated enthusiasm and enjoyable atmosphere that the day brings.

The gains from such collaborative group working have shown (Johnson et al. 2002) that collaborative instructional methods lead to cognitive and affective gains for students at different levels, including undergraduate and post-graduate level. Numerous educational empirical studies, have shown this be the case, time and time again (Johnson et al. 2002). The researchers base it on social interdependence theory, that underlies the most widely used collaborative learning procedures.

13.5 Student Feedback

Feedback from participants was sought in the form of comments after the event. There was therefore no formal feedback questionnaire to qualify learners' own perceived learning outcomes. The feedback comments were therefore based entirely on learner comments reporting on the experience of participation and engagement. A selection of comments relating to the activity were as follows:

'Very good idea with the enterprise project, given a chance to apply practical skills and improve presentational skills. Nice range of work, offers more information and practical applications than most other modules'

'Working with students from other courses was a necessary and valuable experience. 2) Support, when questions were asked and answers were required, has been excellent'

'Facilities are very good and the enterprise activity was a good experience especially in working with people from other disciplines'

'Provided a positive experience in working within a team-based structure to research, design and develop a product'

'I had a good team that really wanted to deliver something great to the rest of the class and enjoyed helping create this'

'I really enjoyed working with other student from different courses and to me it was more interesting than the other assignment in the module'

Overall, the comments that were received from students were positive and favoured the practical aspect and group work. There were however a number of negative comments which were received by students who couldn't see the relevance of the practical aspect of problem to their particular course and discipline—typically such comments came from web design students. Some students were also critical of the structure, organisation and what they considered as erratic scoring depending on assessor. The negative comments were mostly received during the first year of delivery of the Enterprise Showcase event. We took on board the comments regarding structure and organisation and made improvements for the second event delivery, including the choice of projects. The scoring of posters and prototype solutions by various tutors can by nature be erratic as what constitutes a good and viable solution to a practical problem can be a subjective matter. It is for that reason that the average of several scores is determined and allocated as the final score.

13.6 Conclusions and Further Work

Our experiences in the first few years of operation of the Enterprise Showcase Event have been highly positive. We believe we have identified benefits to our students, benefits to our industrial collaborators, and benefits to the school. We have also identified some areas in need of further work and improvement.

13.6.1 Benefits to Students

The students are given the chance to work in interdisciplinary teams, learn new skills, and experience ‘real world’ short term intensive project work. This is particularly useful for their CVs and for discussion at interviews. Several of the students have reported back to us that, during employer interviews for placement jobs, they have been able to draw upon their experiences in the Enterprise Showcase Event to provide examples of working to tight deadlines, working in teams with people they did not previously know, and working on ‘real’ projects. In several cases the students believe that this has been key to their securing an offer of a placement job as they were able to confidently discuss their role and contribution as part of teamwork in order to help solve a practical problem. The experience has therefore provided a range of opportunities for the students to impress potential placement or graduate employers. The event also gives the students the opportunity to make new friends and create contacts outside of their usual peer group, and simply to learn a little about the focus and content of courses from across the School of Computing and Engineering. Overall the experience has provided many positive benefits to the students who have taken part.

13.6.2 Benefits to Our Industrial Collaborators

There are also many benefits to the industrial collaborators. These include the opportunity to receive original input to a problem or business need, including the potential to take this further and develop a new product or solution in collaboration with the students involved; the chance to see and meet potential (placement or graduate) employees; and the chance to get involved with the university, which could then lead to a range of other opportunities for collaboration, including student placements, Knowledge Transfer Partnerships, or an invitation to become more actively involved with our Industry Panel, affording the opportunity to contribute to course and curriculum development. To date the industrial partners have all expressed satisfaction with the event, and have found it to have been a worthwhile experience.

13.6.3 Benefits to the School

There are also a range of benefits to the School and to the Departments therein. These include opportunities to build or enhance links with local companies, which could in turn lead to a range of other opportunities for collaboration as mentioned above; the chance to collect and document useful and novel experiences for discussion at professional body validation visits, university subject reviews, quality audits, etc.; and the chance to provide students with opportunities not usually available to them.

We can then include this information in marketing materials, and for discussion at open days and applicant visit days. The opportunity to engage in ‘real’ projects has been popular with applicants and with their parents and guardians as examples of such activities are often discussed during applicant visit days. The overhead in terms of running the event is relatively low compared to the range of potential benefits to be accrued.

13.6.4 Areas to Improve

While we, our students, and our industrial collaborators all feel that the events to date have been highly successful, we are aware of a number of areas that could be improved. There is the need to address peer assessment, to ensure the marking is fair and that the students who make the greatest contribution to the projects do receive the recognition they deserve. We must also ensure the layout of the room is clear, that there are no advantages or disadvantages to individual groups based on their location within the room, and that power can be supplied safely to the groups that require it. We must also ensure that our industrial collaborators are able to easily identify and locate those teams which are working on their particular problem. We do feel, however, that none of these issues are particularly complex, and we are confident that we can improve the event year on year.

Acknowledgements Many thanks to our colleague Dr Leigh Fleming, who came up with the original idea for the Showcase Event and who worked very hard to make it so successful.

Appendix A—The Showcase Score Card

Showcase Score Card

Your mark will be based on the assessment of 3 judges, if you are able to attract more than three academic judges to your stand, the average of the top 3 marks will be calculated and taken as your assessment mark.

Feedback on this element of assessment will be verbal during the showcase event.

Judges will be looking for the following attributes:

Research—Background to the project and demonstration of understanding of the problem

Innovation—Your innovative and creative approach to solving the problem

Manufacture/Execution—How well the project has been carried out by the team

Communication—How effectively you communicate your ideas through the poster and orally

(continued)

(continued)

Research—Background to the project and demonstration of understanding of the problem

Marketing—How well you sell your idea

Costing—Demonstration of an appreciation for the cost and methods of production

Commercial potential—Potential for a commercial market and consideration of pricing

Teamwork – Demonstration of good team-working and overall effort

| Print Name | Signature | Mark (--/100) |
|------------|-----------|---------------|
| | | |
| Comments | | |

When a judge has completed their scorecard, please place in the envelope provided to ensure that subsequent judging remains impartial.

References

Barrows HS (1985) How to design a problem based curriculum for the preclinical years. Springer Publishing Co, New York

Enactus—dedicated to making the world a better place through entrepreneurial action <http://enactus.org/> Accessed 19 May 2018

Guidelines on course accreditation <https://www.bcs.org/upload/pdf/2018-guidelines.pdf> Accessed 19 May 2018

Hai-Jew S (2020) C2C digital magazine (Spring/Summer 2020) <https://amaxra.com/articles/7-things-you-should-know-about-microsoft-teams>

Johnson DW, Johnson R (2002) Cooperative learning methods: a meta-analysis. *J Res Educ* 12(1):5–14

Krašna M, Pesek I (2020) Influence of moodle and MS teams on teaching-learning-studying (TLS) processes. In: 2020 43rd international convention on information, communication and electronic technology (MIPRO), 2020, pp 612—616. <https://doi.org/10.23919/MIPRO48935.2020.9245356>

- Nelson L (1999) Collaborative problem-solving. In: Reigeluth CM (ed) Instructional design theories and models: a new paradigm of instructional theory. Lawrence Erlbaum Associates, Hillsdale, N.J., pp 241–267
- Projects and group work <https://www.bcs.org/content/ConWebDoc/57834> Accessed 19 May 2018
- Schank RC, Berman TR, Macpherson KA (1999) Learning by doing. In: Reigeluth CM (ed) Instructional design theories and models: a new paradigm of instructional theory. Lawrence Erlbaum Associates, Hillsdale, N.J., pp 161–179
- UK Standard for Professional Competence (UK-SPEC). Engineering Council website Accessed 12 June 2018. <http://www.engc.org.uk/engcdocuments/internet/Website/UK-SPEC%20third%20edition%20%281%29.pdf>)

Chapter 14

Task Versus Process: A Taxonomy for Group Projects



Clive Rosen

Abstract Group projects appear to be a mandatory requirement of undergraduate degree programmes in software engineering and other, similar courses. But what are these projects for, and what do we expect students to learn from them? Often the group project exists within the curriculum simply because it is required by various authorities and without a clear pedagogic rationale. The language used in module specifications usually refers to employability skills and collaborative working, but often disguises a wide range of different delivery approaches. Assessment regimes may prioritise outputs over the process of producing those outputs. Even the word “process” is ambiguous. Do we mean the systems development process or the group process? If the former, why do we need a group? If the latter, how can this be assessed to the satisfaction of students and external examiners? The “Task Versus Process” taxonomy firstly suggests a model for categorising the aims and objectives of the group project, secondly, evinces an appropriate assessment strategy and finally suggests some approaches that can be taken if staff teams wish to address the “Personal Development” quadrant of the taxonomy. The chapter contextualise the student group project within broader aspects of curriculum design and then attempts to clarify the multifarious and often contradictory objectives proposed for student group projects. Fully appreciating what it is we, as instructors hope to achieve in group projects takes us beyond resigning ourselves to the compulsory inclusion of the project. Projects offer the opportunity to help our students achieve their full potential and to actually deliver the learning outcomes.

Keywords Learning and teaching strategy · Student group projects · Group project assessment

C. Rosen (✉)
Passerelle Systems Limited, Newcastle Under Lyme, UK
e-mail: passerelle.ltd@outlook.com

14.1 Introduction

What is the purpose of student group projects in software engineering courses? Their ubiquity suggests that they must be a very important component in such courses. However, when one explores the range of learning outcomes associated with group project modules, one finds a wide range of potentially conflicting objectives and often assessment approaches that are not wholly compatible with the learning outcomes. This chapter explores the reasons why this might occur and proposes a taxonomy that associates the actual objectives of the group project with the assessment approach taken. The purpose of the group project is situated within the more general model of the learning process (Fig. 14.1) in order to plot clear connective patterns from programme learning outcomes, through module learning outcomes to appropriate assessment methodologies. It is hoped that this approach will enlighten discourse on programme design in IT and computing related courses.

Group project module learning outcomes often cite such employability related requirements as the ability to work in a team, decision making or effective communication. These nebulous terms are often treated as an inexorable consequence of the group project process, avoiding the fact that learning about oneself requires self-reflection and challenging preconceived personal constructs. Students need to be presented with opportunities to learn and also be given support for self-reflection. Computing students in particular are often reluctant to engage with this sort of process. The group project module provides the opportunity to integrate employability skills into the curriculum. This chapter offers some suggestions of how this might be done. However, if it were that easy, everybody would be doing it. Resources play a part, but there is also a challenge for staff. We cannot expect students to reflect

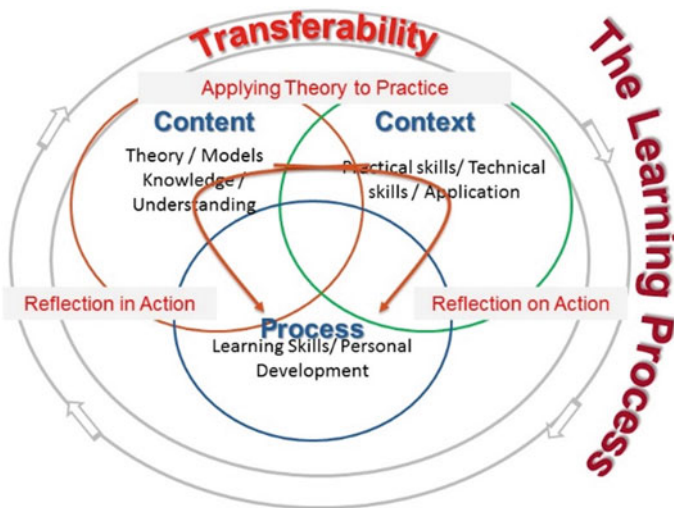


Fig. 14.1 The learning process in higher education (Rosen and Schofield 2011; Rosen 2015)

on themselves without having gained some self-awareness and some ability for self-reflection ourselves. Many would not see this as part of their lecturing role. It is hoped that some of the suggestions made in this chapter will help to overcome this reluctance on the part of both students and staff.

14.2 The Purpose of Student Group Projects

The demands made on student group projects are often extensive. Projects are often expected to deliver a wide range of outcomes including “real world” experience of industrial weight project work, employability skills, team-working and communication skills. Group projects are also viewed as capstone modules, integrating knowledge from other parts of the curriculum (such as database development, web site development and programming). This range of expectations can lead to students becoming confused regarding the purpose of the module and what is expected of them. It can also result in a situation where the mechanisms used for assessment either fail to test the actual learning outcomes, fail to challenge students to produce their best work, or fail to provide sufficient opportunity for students to exercise appropriate critical self-awareness.

Before considering the role of the student group project, it might be helpful to stand back a little and consider how higher education relates to governmental and industrial expectations of software engineering students and how the group project sits within this context and the software engineering curriculum. The diagram above helps to illustrate this positioning.

This model identifies three intersecting circles representing models of student learning which can support the development of a learning and teaching strategy at both the curriculum or module level. The three inner circles represent “content”, “context” and “process”. These circles intersect in a “Venn” style diagram suggesting the lack of precise boundaries between the three forms of learning.

“Content” is comprised of the theories pertinent to an academic domain, abstract knowledge associated with it and accumulated experience concerning the subject area. It is abstracted from any given situation. Content can normally be found in text books, journal papers etc. and is often delivered in HE settings through formal classes. Content is the domain that academia traditionally focused on delivering and, more particularly, developing, through on-going research. So an example of content in IT would be normalisation.

“Context” is the application of a particular theory to a given context and requires practical/implementation skills and the ability to select an appropriate solution from the body of knowledge (content). Vocational subjects generally give more emphasis in this sphere than the humanities, as, being able to deliver the product or service is an essential element of success. Context is situated in real world problems and encompasses knowledge and experience associated with practical problem solving. When employers complain about the lack of readiness of graduates to work, it is often issues associated with the context sphere that they complain about.

The association of context with competency contrasts with the association of content with understanding. An example of the difference between the two within the computing domain would be the capability of a student to program in a particular language (context) compared to an understanding of the principles of programming languages (content). In reality, both are needed. Students struggle to understand programming principles without knowing how to solve problems in a given language, yet they must understand the principles to be able to learn the range of languages they may be confronted with when they leave university. Most courses contain elements of both content and context. Examples and case studies are often used in teaching for just this purpose. Without context/application, abstract knowledge is interesting, but essentially unproductive.

Whilst both understanding and technical competence are considered to be essential requirements of the higher education system, neither are sufficient either singularly nor together. An implicit demand in recent years, particularly from industry (but arguably often undermined by governmental policies) has been the ability of graduates to transfer knowledge from one context to another. This requires the third sphere, “process”.

Today’s students need greater capability; in their ability to learn independently; to use their knowledge and understanding to solve unfamiliar problems; to understand the processes and standards involved in learning; to recognise which tools and techniques are applicable to a particular context as well as other social and learning skills. Students need the problem solving skills of logical reasoning, deduction, research and critical evaluation. These capabilities do not automatically emerge from content and context, but need to be nurtured and encouraged in equal measure through reflective practice. Schön’s (1991) concepts of “reflection in action” and “reflection on action” can be applied here to support the learning process. “Reflection in action” is the idea that (in Schön’s phrase) the “Reflective Practitioner” (in this case the student) does not simply do work, but thinks about it whilst they are engaged in it to ensure that any particular course of action is the best available at the time. “Reflection on action” is the idea that once a particular project is complete, a professional should not give a huge sigh of relief and move on to the next project, but should review the project to see what can be learnt from it. Were there any mistakes made? Could the output have been more effective? Were any inefficiencies introduced into the process? (The parallels between this and the Shewhart Cycle (1986) and Deming (2000) quality improvement process should not be overlooked.) Traditionally, one might argue, that higher education has used research as a vehicle to enculturate students into these activities, but the group project represents an ideal opportunity to encourage students to become reflective, particularly if ongoing progress reviews are included in the module implementation. However, reflection, particularly reflection on one’s own performance, involves the ability to be self-critical which necessitates a level of self-awareness and self-confidence; qualities that are often limited amongst computing students.

In practice content, context and process are not independent of each other. Teaching staff switch between them in rapid order, drawing on case studies to expound

theory, asking students what they think might be going wrong (theorising) in a practical application and asking them to consider alternative approaches to the problem they are trying to solve. However, this model does offer an idealised conceptualisation of what we might be trying to achieve. This objective for Higher Education may be aspirational and idealistic, but it is expressed in Benchmark statements for all the computing subject areas. And, the group project provides curriculum designers with the opportunity to demonstrate the development of these skills in the programme. This, it might be suggested, is often one of the sources of conflict within the group project module. For how can we help develop self-reflection in students and how do we assess it? One might argue that this is particularly difficult with computing students who are often resistant to the notion that self-awareness, personal reflection and personal development are necessary to becoming software or systems engineers.

Student group projects rarely include new knowledge and skills. They are more likely to focus on applying the skills that students have already acquired in other modules to a particular situation (or “context” in the diagram above) and often in a more complex environment. But the learning outcomes for the group project module will often include “the ability to work in a team”, “real world understanding”, “creative thinking” and so on. These skills are “Process” skills as defined by the model above. However, students will usually be expected to be able to demonstrate that they have actually understood what they have done. They might be expected to explain why they did something one way and not the other. In other words, to use critical analysis to show that they know why they did something. This activity is situated in the “Content” part of the model above. So, when student group projects are considered in the light of the above model, they sit very firmly at the nexus of the diagram. It is unsurprising therefore that lecturers may feel pulled in several different directions at once. For any particular group project module, should the emphasis be on the real world experience, reasoned decision making, team working of something else? The consideration of “Task Versus Process” may help to articulate the answers to some of these questions by informing the module design process and helping to make decisions more explicit.

14.3 Task Versus Process

The “Task Versus Process Grid” provides a means of considering the primary purpose of the group project in a particular programme. The grid offers two parametric dimensions. On the vertical axis we have the question of, “for this module, on this programme, do we consider students’ capability to complete the task more or less important than their understanding of how to undertake the task (i.e. the process)?” However, process can have two different meanings here. The first meaning suggests that the process we are considering is the development process. A second interpretation of process is the team process. Are we more concerned about the way the system is produced or an understanding of the group dynamics and hence the ultimate capability of the student to work in a real industrial team? The second axis helps to clarify

this difference. If the former, we would be more concerned with the outputs from the students' activity. If the latter we would pay more attention to the students' ability to think about the way the group has operated, and how the intra-group process has affected the conduct of the task.

The four quadrants therefore provide us with a taxonomy to consider where we wish to place the emphasis for the module. If we are concerned with a student's technical capability, this would place us in the lower left hand quadrant of the model. We would want the student to demonstrate their ability to produce some outputs (probably in the form of a database, website, program or app) and we would concentrate on the knowledge associated with "doing" the task. Alternatively, we may be concerned less with the actually getting the task done than with how the students went about the task. Did they establish the requirements? Did they produce a product design? How rigorous was their testing? These are questions regarding the governance of the project and place us firmly in the "software engineering" quadrant.

If the purpose of the project is to promote employability skills, we would be concerned with how students worked together to produce their outputs. The artefacts themselves would be less important than the students' ability to reflect on what worked, what did not, and how they communicated with each other to arrive at appropriate and timely decisions. What did the students learn from their experiences?

The final quadrant, the "personal development" quadrant is perhaps the most ephemeral and also the least well-articulated quadrant. It may also be the quadrant most aspired to when curriculum designers construct courses, and perhaps the hardest to achieve. Questions relating to students' personal development include how did the group process support or prevent you from contributing to your maximum ability? Were you able to contribute to decision making? Were you able to reflect in and on action (Schön 1991)? These questions can be the most difficult in which to engage students, and the most difficult to assess. Teaching staff are probably least qualified to support student learning in this area and, perhaps therefore least willing to focus on this aspect of the group project.

Figure 14.3 summarises this discussion.

14.4 Assessment

The argument above would be incomplete without a discussion of how to assess team projects within each quadrant as, if each focusses on particular forms of output, each should use those outputs as the means of assessment. Figure 14.4 outlines the means of assessment that would be best associated with each quadrant.

It can be seen that the range and variety of potential assessments is extensive. Learning outcomes defined in module specifications are often ambiguous and lack precision, leaving it to the module leader to design their own assessments. The breadth of scope however means that it is not possible to assess every outcome which is why having more clearly defined learning outcomes becomes important (notwithstanding the point that if the module team are not clear what they are trying to achieve, the

students are likely to be confused). Some may argue with the details of the taxonomy presented here, but this analysis clearly demonstrates the difficulties associated with assessing group projects. Whilst the “Software Engineering”, “Technical Capability” and “Employability Skills” quadrants all have relatively self-evident outputs that can be assessed, the problem is deciding on what to focus.

The Personal Development quadrant is more problematic for many reasons. It has already been suggested that computing students often fail to appreciate the importance of the more esoteric skills resident in this quadrant. Often staff feel reluctant to stray into an area they might feel unqualified to assess. The subjectivity associated with assessing personal development is also problematic in a discipline aspiring to achieve engineering status. Furthermore, many staff have developed an antipathy to assessing personal development from the experience of trying to assess “key skills” in HND, “core skills” in apprenticeship schemes and “employability skills” as part of the employability agenda. One lesson many have learnt from these experiences is that if employability skills are not fully integrated into the curriculum in general, and the group project module in particular, students react poorly to their add-on, ad hoc injection into a programme. This raises the question of how can seamless integration be achieved in such a practical subject area such as computing. The next section of this chapter suggests some approaches that can be used as part of the group project. Other chapters in this volume look at other approaches.

14.5 Implementation

This section will concentrate on the introduction and assessment of the skills associated with the upper right, “Personal Development” quadrant of the Task Versus Process Grid (Figs. 14.2, 14.3 and 14.4). Staff may feel more confident implementing some suggestions than others. Logistics and resources also play a part as do the peculiar academic environment surrounding all group projects. With this in mind, staff may choose to introduce changes over a period of time if they feel any of these ideas are appropriate to the pedagogic context in which they are working.

14.5.1 *Live Client Projects*

A point of dispute amongst group project module leaders is whether or not to trawl for real client projects, allow students to construct their own projects or for the module team to design projects the students can do as a group. The choice may well depend on the particular circumstances in which the project is being conducted and the objectives of the project. The Task Versus Process taxonomy above can help with this decision. However, often the choice boils down to more pragmatic considerations. Do the module team have sufficient time to find and vet potential projects? Can we trust students to deliver for real clients? What mechanisms are available for installation on

Fig. 14.2 Task versus process grid

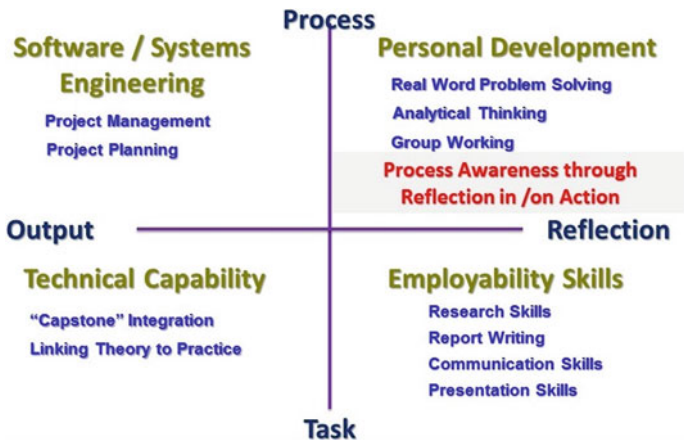
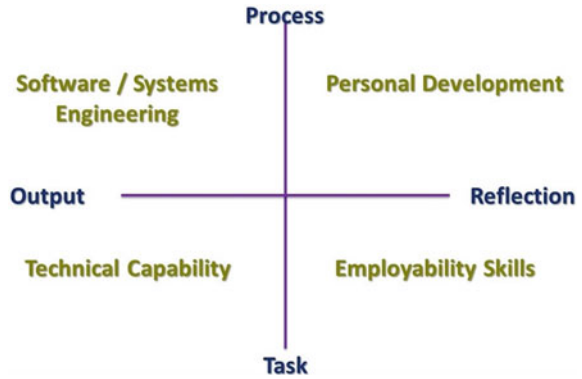


Fig. 14.3 Activities most associated with each quadrant of the task versus process grid

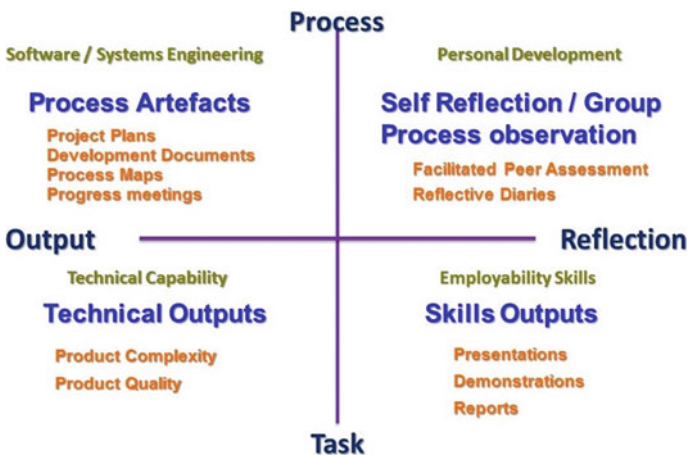


Fig. 14.4 Assessment associated with each quadrant in the task versus process grid

client systems and ongoing support? Real projects tend to be much messier and more complex than contrived projects. This means that students need to be given much more hands on support during the project; particularly with client management. But this is also the strength of using real projects and real clients. Students discover that part of systems development is client management; working with clients to identify their actual requirements, agreeing with them what is practical and what can be delivered given the time and resource constraints. Clients do not conform to the stereotypes implied in software engineering texts. Working with real clients exposes students to these realities and provides many powerful learning opportunities.

Furthermore, students engaged on real projects experience one of the fundamental differences between employment and student life which often goes unacknowledged; that work is undertaken for an employer, not for oneself (excepting self-employment). Studying is a very self-centred occupation and feeds into many students' egocentricity. Employment implies doing work for other people (even when self-employed) and this involves a radically different mind-set. Real, live projects can help with this transition in a relatively safe environment, at least one where the module staff can provide a safety net if necessary.

One further factor when considering whether live projects should be used, especially when external (to the institution) clients are adopted, is the role of the student as ambassador to the college or university. This can be a very positive experience for students, and the institution, if students embrace the responsibility, but can cause difficulties for the programme staff if things go wrong. Teaching staff must decide for themselves the balance of risk and benefit. Client preparation is essential and staff cannot absolve themselves from continuous client management, particularly when working with external clients. There is undoubtedly less work for the teaching team when contrived projects are undertaken, but they are potentially less rewarding for both students and staff. Group projects present this opportunity. There is rarely a shortage of potential clients. Only the module team can assess the cost/benefit ratio in their environment.

14.5.2 Student Application to Chosen Project

One approach that can help to integrate employability skills with the group project is to require students to make a formal application to a particular project. This works particularly well when external clients are offering projects. The assessment would require students to produce a CV, write a letter of application to the project provider and attend an interview for the project. (Students would only get their first choice project if they pass the interview.) Careers staff and project providers can be recruited to help with the interview process so the experience mirrors more closely a genuine recruitment exercise. An assessment centre can also be included in the process if time and resources permit.

One essential element of this activity is immediate feedback to students from the interview. It is remarkable the learning that can occur even in this relatively

contrived situation when a student has underprepared for the interview and then received feedback from an interview panel on their performance. A large majority of students will never have previously experienced a formal interview and the reality and immediacy of such a meaningful interview is a very powerful learning experience. It is important for the interview panel to set the right tone. Hostile interrogation could generate a student's defence mechanisms which would not be helpful, a potential danger when a student fails to take the process seriously or fails to prepare for the interview. Having the external client on the interview panel brings a number of benefits. Not only does it make the experience more real, it also increases the client engagement in the project process. At the same time, being interviewed by a real client promotes greater student commitment to the group project and therefore improves the prospects of their full engagement.

A commonly recognised problem with the assessment of student group project is the evaluation of the student free-rider, i.e. a student who contributes little to the project, but expects to be awarded equal marks to those that have worked on the project. Facilitated Peer Assessment, see below, helps to alleviate this problem, but interviewing students for the project in the first place, also helps student engagement by increasing their initial investment in the project.

14.5.3 Skills Inventory

The skills inventory presented here is a development from a skills matrix that was presented to the HEA BMAF Placements workshop in 2006. The inventory has evolved to its current version since that date. Six principles have emerged during this evolution:

1. To provide a supportive structure for student reflection.
2. To minimise the psychic threat to students.
3. To encourage students to identify specific examples to work on.
4. To focus on learning from personal experience.
5. To make the process of self-reflection intrinsic to the exercise.
6. To use the skills employers say they want as the vehicle for self-evaluation.

As can be seen in Table 14.1, each row represents one of the skills employers consider to be important. The student is asked to identify a specific example of when they have demonstrated each skill. If they have more than one example, they can insert additional rows under the same heading. The "activity description" column requires the students to articulate the specific circumstances. The "when" and "where" columns are included to ensure a particular time and place, and not a generalised activity. In the "what did you do" column, the word to emphasise is "you". Students often experience these activities as part of a group, and it is important that they learn to differentiate themselves as an individual rather than simply a member of a group so that they can attribute the outcome to their personal contribution. Students are often reluctant to appreciate what they do well, and this exercise helps

them identify strengths as well as learning experiences. The seventh column, the “learning” column is perhaps the most important. It is these entries where the ideas of self-reflection and self-evaluation are expressed. Having the experience alone is insufficient. Professional development requires personal reflection on what has been learnt from the experience. To experienced professionals this may become integral to their professional practice, but to students and new entrants into the profession, the concept of continuous self-evaluation and reflection are not self-evident and often students are never told that it is required. This exercise makes explicit an activity experts often take for granted.

The final column is optional. It is there to help students develop an action plan and to decide on which skills should be priorities for future development.

The inventory can be used as a standalone exercise, but has greater relevance if students are preparing to apply for placement or post graduate opportunities. UK employers commonly adopt a competency based approach to interviewing applicants (sometimes known as CAR, Context, Action, Result). This takes the form of asking questions such as ‘describe a situation when ... what did you do ... what was the outcome?’ The skills inventory helps prepare students for such questions and therefore improves their chances of performing well in interviews.

Table 14.1 Skills inventory

| Skill | Activity description | When | Where | What did you do? | Outcome | Learning | Priority |
|------------------------------|----------------------|------|-------|------------------|---------|----------|----------|
| Leadership/initiative taking | | | | | | | |
| Influencing/negotiating | | | | | | | |
| Team work | | | | | | | |
| Effective communication | | | | | | | |
| Self-motivation | | | | | | | |
| Decision making | | | | | | | |
| Planning/organisation | | | | | | | |
| Working under pressure | | | | | | | |
| Personal development | | | | | | | |
| Commercial awareness | | | | | | | |
| Presentation/report writing | | | | | | | |

14.5.4 Facilitated Peer Assessment

Facilitated Peer Assessment (FPA) was first described in Rosen (1996) as a means of assessing the individual contribution to group projects, but it is also a means of supporting student reflection in that students are required to assess their own performance in relation to other members of the team and to give feedback to other team members. As such, it provides a vehicle for exploring self-reflection both as part of the project and as part of the assessment process.

FPA has the form of a post project review in which students in a face to face group meeting with a module tutor evaluate the success or otherwise of the project. As a group, they must also reach a consensus on the allocation of individual members' contributions to the project. The role of the staff member is to facilitate the discussion rather than to influence it (other than to encourage the students to reach a conclusion).

FPA has several advantages over other forms of peer assessment which usually consist of students completing undisclosed assessment forms about each other. The lack of transparency with this traditional approach undermines the value of peer assessment. Students can attribute any contribution level to their fellow team members, but it is unclear what criteria or perceived equity of treatment they are applying. Group projects can provide excellent opportunities to explore unconscious bias in teams, but undisclosed assessments of each other fail to challenge these biases in the assessment method. As a result, it is often down to the module team to moderate the students' assessments. It can be argued that moderation of marks is the responsibility of the module team, but if the module team are going to moderate the peer assessment, students may question the reliability, and possibly the validity of the process. Furthermore, academics' moderation of the peer assessment process undermines the students' sense of responsibility for the process as they cannot take full ownership of the process.

FPA avoids these problems. Firstly it is open and transparent. Students know what is being said about them by other team members, both positives and negatives. FPA therefore has the capability of being a very affirming activity as well as a functional one. Students retain responsibility for the process as the role of the staff member is facilitation, not moderation. FPA also provides the opportunity for students to reflect on the assessment process and the legitimacy of various criteria. They must decide whether the team's project manager has contributed more or less than its chief designer. What is the nature of contribution? This has the effect of requiring students to reflect on their own performance, as does the receipt of feedback from other team members. FPA does challenge the assertiveness of less articulate students, but it is the role of the facilitator to support students to express themselves (one of the key requirements for employability).

Student ownership of the FPA process has an important side effect. Group project module leaders will be familiar with the concept of the "free riding" student, the one that hopes to benefit from the work other students do without contributing much themselves. Often the only (meaningful) sanction the staff team have is to sack such students from the team. This is very much a nuclear option as it results in the dismissed

student failing the module and staff have to have very clear evidence that a particular student has not, and will not, contribute. In group dynamics terms, a student's lack of engagement in the project can be very complex. Gender, race, culture and language can all play a role in peer acceptance or rejection within the team, and an apparently non-contributing student may feel that they are being excluded by the group rather than the student not wanting to contribute. Students have responsibility and authority for FPA so they have greater responsibility for the successful functioning of the group. FPA gives the students themselves a sanction against non-contributing members; one that can be referred to during the project to bring pressure to bear on a student and one that avoids the nuclear option. The result is that contributing students feel more in control, and often respond positively to the responsibility they have for project progress. The nuclear option is still available, but if it does need to be invoked, there is usually a clearer evidence trail to support it.

FPA is problematic as it does depend on the confidence of academic staff to manage the FPA process. Smouldering conflict that has been present within the team can, and sometimes does emerge during the FPA. Staff may feel that managing such conflict is not one of their responsibilities, but if this tension has resulted in the team not achieving its full potential, staff should at least be aware of it, and it must surely be beneficial for students to explore how such tensions affect the progress of a development project. After all they are likely to face these sorts of situations during their working life and if they have reflected on it within the relatively safe and supported group project environment, they will be better prepared for it in the world of work.

Managing conflict in group projects is challenging for staff. It is probably the aspect of student group projects academics find most off putting, and probably one of the reasons why many staff, when charged with the responsibility of running group projects, avoid the "Personal Development" quadrant of the "Task Versus Process" grid. Yet the ability for graduates to be able to work successfully with other team members often marks them out as potential employees, and in a profession where such capability is a rare commodity (Teague 1998; Capretz 2003), suggests that the effort may produce its own rewards.

14.5.5 Team Membership Selection Process

The team membership selection process is somewhat orthogonal to the taxonomy of group projects as any team selection method could be used whichever quadrant the project is focussed on. However it is nevertheless a bone of contention and teaching staff need to be cognisant of the influence the selection process can have on the group dynamics. In group theory, group formation is one of the most significant events in the life of a group, and the method of its formation reverberates through the group process in one form or another.

The debate over the team selection process revolves around whether or not students should be allowed to select their own groups or whether team members should be

selected by staff members. The arguments over which is best have been well rehearsed and are not going to be regurgitated here. However, two options are available that may be of interest that somewhat sidestep this problem and provide an opportunity to encourage student reflection.

The first is to make use of an early tutorial session to discuss with the students the method of selection to be used. This involves brainstorming the various ways in which groups can be selected. Students usually identify between 20 and 40 different possibilities ranging from the obvious, that they (the students) should select their own groups or the teacher announces the teams, to geographic proximity to where students live and the physical height of the students. The class then go on to discuss the pros and cons of each method, eliminating some for practical or ethical reasons and looking at the more legitimate approaches critically. This discussion often raises some ethical and moral questions such as ‘what happens to the student no one wants in their group?’, ‘what about the new student who nobody knows?’ or ‘what happens when friends fall out?’. After this discussion, the students then decide which method to adopt. It has to be said that students usually adopt a self-selection method, though not always. However this approach does place the responsibility of the selection method with the students. They are therefore unable to claim that if the project goes sour it was the teacher’s fault. Students do seem to like this discursive approach and it is nearly always enlightening, both for students and academic staff.

The second approach that has been alluded to above is the ‘project first’ approach. This is where the projects are defined first and students then choose the project they want to work on. This approach need not be combined with the interviewing process described above or include external clients, but it does mean that fairly clear specifications of what the project involves need to be provided to the students. Again this approach places the responsibility for the team selection with the students, albeit indirectly in this case. The project first approach subtly changes the initial dynamic of the module emphasising the importance of the project task over being a bit of fun with mates at the end of the year.

14.6 Conclusions

This chapter aims to develop a pedagogic taxonomy for determining the focus of student group projects in the context of the objectives of any given software engineering course. It identifies four quadrants that might become the focus of a group project module on that programme, the “Software Engineering” quadrant, the “Technical Capability” quadrant, the “Employability Skills” quadrant and the “Personal Development” quadrant. For each quadrant, it identifies the type of activities associated with the quadrant and the assessment artefacts that might be used to assess student capability. In providing this mapping, it is hoped that coherence and consistency between assessment methodology and learning outcomes can be maintained, and that, as a result, staff may be better able to articulate their own aims for the module and prepare students for the assessments they might expect. Congruence between

learning outcomes and assessment methodology seems to be taken for granted in other modules within a software engineering curriculum. Because there can be such a wide variety of both overt and covert objectives for group project modules, it seems important to explicitly resolve any ambiguities that might exist regarding what the group project module in a given programme is really for. It is hoped that this taxonomy will help in this regard.

The “Personal Development” quadrant in particular can be quite challenging. It takes academic computing staff into unfamiliar territory and areas which it might be argued are not their responsibility. However, group projects are often cited as being key components of the employability agenda, and UK universities and colleges have signed up to this agenda, however reluctantly in some cases. In reality, it is not easy to differentiate elements of personal development from employability in general. Team working for example always involves managing personal differences and sometimes even conflict. If we are genuinely to address such aspects of employability, we probably need to provide students with the opportunities to reflect on, and learn from appropriate experience. The group project module can be such an opportunity. Some suggestions have been proposed in this chapter which can be employed to help to stimulate student personal development. Many may still wish to steer clear of any involvement in this aspect of the current higher education agenda. Whichever approach is adopted, it is hoped that the taxonomy presented here provides a framework for decision making.

References

- Capretz LF (2003) Personality types in software engineering. *Int J Hum Comput Stud* 58(2):207–214
- Deming WE (2000) *Out of the crisis*. MIT Press, Cambridge, Mass
- Rosen C (1996) Individual assessment of group projects in software engineering—a facilitated peer assessment approach. In: 9th annual conference on software engineering education. Daytona Beach Florida: IEEE Computer Soc Press
- Rosen CCH (2015) A philosophical comparison of Chinese and European models of computer science education. (A Discussion Paper) 11th CHINA—Europe International Symposium on Software Engineering Education Zwickau Germany April 2015
- Rosen CCH, Schofield R (2011) Reliability and validity in work-based learning, work based learning futures V, Derby
- Schön D (1991) *The reflective practitioner: how professionals think in action*. Aldershot, Arena
- Shewhart WA (1986) *Statistical method from the viewpoint of quality control*. Dover, New York
- Teague J (1998) Personality type, career preference and implications for computer science recruitment and teaching ACSE98 Brisbane Australia

Chapter 15

Realising the Threshold of Employability in Higher Education



Chris Procter and Vicki Harvey

Abstract A substantial body of work has tested and developed ‘Threshold Concepts’. A Threshold Concept may be considered “akin to a portal, opening up a new and previously inaccessible way of thinking about something ... it represents a transformed way of understanding, or interpreting, or viewing ... without which the learner cannot progress” (Meyer and Land in *Threshold Concepts and Troublesome Knowledge 1—Linkages to Ways of Thinking and Practising in Improving Student Learning—Ten Years On*. OCSLD, Oxford, 2003). Little research however exists on the relevance of the concept to employability. Employability is fundamental to Higher Education, yet its role in the curriculum is unclear and contested. Our practice suggests that developing knowledge about employability is a threshold which, when reached, empowers and gives confidence to the student. To achieve this means embedding this knowledge in the curriculum. The paper discusses the delivery of a large module with this aim, explaining how the design of assessment was fundamental in guiding students to a transformed way of understanding employability.

Keywords Employability · Threshold concepts · Competencies · Professional development

15.1 Introduction

Since the mid 1960’s there has been significant change which has paved the way for the development of the current employability agenda in Higher Education (HE) in the United Kingdom (UK). The Robbins report of 1963 (Barr 2014) heralded the creation of many new Higher Education Institutions (HEIs) to ensure that all who were qualified and wished to enter should be able do so (Barr 2014). Figures from the Higher Education Statistics Agency (HESA 2017a) indicate that there have

C. Procter (✉) · V. Harvey
University of Salford, Salford, UK
e-mail: Chris.t.procter@gmail.com

V. Harvey
e-mail: v.m.harvey@salford.ac.uk

since been consistent increases in student enrolments. With the number of first year, first degree student enrolments standing at 542,575 for 2015/2016, this shows a 3% increase year on year since 2006/2007.

The UK has the second highest graduation rate in the OECD, with around 47% of school leavers entering HE, but such growth is not replicated in the graduate labour market (The Guardian 2016 and CIPD 2015). Government figures suggest that 31% of all graduates are not doing graduate—or high-skilled—jobs (BIS 2016a). Employers report (Archer and Davison 2008; Woods and Dennis 2009) that graduates are not work ready and do not have the requisite competencies. Despite a substantial number of under-employed graduates, employers continuously report a skills shortage (CIPD 2014; CBI 2015), and the role of HE in facilitating this gap is continually called into question (Cranmer 2006). Successive UK governments have responded by steadily increasing the pressure on HEIs to demonstrate the employability of their graduates. When considering Computing graduates specifically, The Tech Partnership report (Matthews 2017) emphasises the gap between student and employer expectations and the need for Universities to develop employability skills. The review conducted by the Higher Education Funding Council for England (HEFCE) led by Nigel Shadbolt (HEFCE 2017) recommended a much greater emphasis on employability and proposed integrating ‘work ready’ skills as an accredited part of the curriculum.

Following a brief explanation of the evolution of policy, this paper discusses how employability is defined as a set of competencies by employers. It then addresses how best HEIs can help students to demonstrate their employability arguing that this is best achieved by initiatives embedded within the curriculum. We explain the relevance of theory concerning Threshold Concepts and how assessment can be used as the lever to enable students to realise what we term the threshold of employability. The paper explains the design and implementation of a large multi-disciplinary module undertaken by undergraduates. Whilst they were largely on business related programmes (including IT) the approach is equally valid for computing students. Student reflection and feedback is used to demonstrate the relevance of our approach. Conclusions are drawn concerning the value of embedding employability in the curriculum to help students conquer their ‘Monsters of doubt’ (Hawkins and Edwards 2013).

15.2 Development of Government Policy on Employability

For over two decades successive governments have sought to intervene to ensure that HEIs addressed the gap between graduate capability and the requirements of the labour market (Artess et al. 2017). The Dearing report (NCIHE 1997) provided a major impetus for UK HEIs to become engaged in employability skills development. There has been a continued policy emphasis on the strong relationship between HE and economic prosperity, and consequent need for the production of ‘employable’ graduates (BIS 2016b).

Since the turn of the century policy measures have sought to inculcate employability into programmes of study by the use of national metrics. The Destination of

Leavers from Higher Education (DLHE) survey, which commenced in 2003, has been conducted with all former undergraduate students six months after their graduation. It allows for comparison between HEIs in relation to the quality of career/employment destinations post-graduation. The data from this is used as a key measure in assessing the performance of HEIs in league tables. More recent initiatives in the UK linking the achievement of such measures to student funding (i.e. the Teaching Excellence Framework, TEF) are designed to further force the hand of University management in delivering employability. It is worth noting however that any data applied for such measurements needs to be placed in context, as data collection methodologies are reviewed to meet the scrutiny of a wider range of users. The most recent data set for DLHE, implemented as the Graduate Outcomes collection was published in January 2020 (HESA 2017b). Importantly what remains is that Artess et al. (2017) suggest that employability has now become one of the top priorities for HEIs.

15.3 Defining Employability: Understanding Competencies

Understanding the employer approach to employability is fundamental to addressing the issue. Professional bodies have a substantial influence over the curriculum in traditional vocational degrees, such as law or nursing, and the importance of the curriculum in defining employability, knowledge acquisition and training for a career is widely accepted. Many other programmes, such as those in Computing, whilst they are vocational are not training individuals for specific jobs, do aim to facilitate the possession of a mix of employability skills and business knowledge desired by employers. However, today's school children will likely go on to graduate from University and in future work in organisations and job roles that don't yet exist. Upheaval and fluidity in the job market mean that employability is increasingly defined by one's possession of, and ability to articulate, a given set of competencies, alongside requisite technical skills (e.g. Independent 2015; Times Higher 2015).

The focus on competencies in the job application process has become ever stronger in the twenty-first century: students who can demonstrate these competencies have a substantial advantage in seeking graduate jobs and future promotion.

The Higher Education Academy (HEA 2015) identified a composite list of 34 terms associated with graduate attributes, suggesting that employability is difficult to define, and being able to demonstrate competency is also rather slippery, described as "*a personal state that individuals occupy*" (Artess et al. 2017, p. 10). More usefully, Dubois (1998) defines competencies as those characteristics—knowledge, skills, mind-sets and thought patterns—that, when used whether singularly or in various combinations, result in successful performance. There are numerous lists of the competencies most in demand (Diamond et al. 2011): these typically include teamwork, communication and networking, leadership, business awareness, initiative, flexibility, enthusiasm, personality and many others.

15.4 The Importance of Articulating Competencies

Therefore, the challenge in HE is not just to teach these competencies but to help students realise their possession and develop the ability to articulate them. Students, however, may not fully recognise the importance of engaging with employability while studying, beyond managing their part-time work (Tymon 2013; Greenbank 2015). Tomlinson, in his Review of Graduate Employability (2012), suggests that many do not appreciate the competencies sought by employers, nor have the knowledge to articulate these, despite this articulation being the most critical component of a typical recruitment process. He adds that it is not just about individuals possessing certain competencies which enhance their employability, but being able to package this for employers:

Brown and Hesketh's (2004) research has clearly shown that ... for graduates, the challenge is being able to package their employability in the form of a dynamic narrative that captures their wider achievements, and which conveys the appropriate personal and social credentials desired by employers. (Tomlinson, p. 420)

Tomlinson et al. (2017) have developed this further arguing the need for HEIs to develop a range of 'capitals' with the student:

Capitals can be understood as key resources, accumulated through graduates' educational, social and initial employment experiences, and which equip them favourably when transitioning to the job market. (Tomlinson, p. 17)

Such 'capitals' not only include intellectual capital but social capital as well.

If, as suggested, employability for recruiters rests upon an individual candidate's ability to demonstrate their competencies, this has great significance for HEIs developing employability. The challenge therefore is not whether or not HE should seek to develop competencies, but *how and in what ways this could be done*, taking into account some students' historical lack of engagement in employability initiatives.

15.5 How to Deliver Employability?

All Universities have traditionally encouraged their students to consider their future employment and provided support for this, managed via an extra-curricular Careers Service. Typically, they organise employer visits, recruitment fairs and offer careers information, advice and guidance. However, optional engagement with such activities is not sufficient to reach all students and ensure graduates have appropriate employability knowledge and understanding. Thus, many institutions look to the curriculum to see whether and how they can embed employability. This has been given recent impetus in the UK by the support of the Higher Education Academy (HEA) which published a framework for employability in tertiary education (Cole and Tibby 2013). The HEA's report on Pedagogy for Employability (Pegg et al. 2012) includes many case studies on embedding employability in the curriculum.

None of these however, explicitly link course assessment to the articulation of student competencies in the way described later in this paper.

15.6 Should Employability Be Developed Within the Curriculum?

Cranmer (2006) conducted research amongst a group of UK Universities, looking at different models of delivery, and concluded that there was limited evidence of the development of ‘employability skills’ through classroom teaching. She argued that resources would be more usefully deployed by HEIs in employment based training, employer involvement in the curriculum and opportunities for experience with employers such as placements/internships. Her conclusions suggested that teaching employability skills was not achievable or (even) desirable. Tymon (2013, p. 853), acknowledges the complexity of employability and further questioned whether the development of employability was within the capability of HE institutions:

It is also unclear whether many of these skills and attributes can be developed in practice and, if so, what the role of higher education institutions should be. Putting aside the arguments about whether higher education institutions are able, willing or designed to develop employability, there is evidence to suggest there are alternative options which may be more appropriate.

Tymon (2013, p. 853) goes on to declare that with improvement from HEI’s:

Skills can be developed and are embedded in the curriculum, but many first and second year students appear to lack engagement with these activities. This must reduce their motivation to learn and inevitably impact on successful development.

Tymon concludes that “development of these [employability skills] is possibly outside the capability and remit of higher education institutions.”

15.7 Making Employability in the Curriculum Work

There is a lack of evidence of the efficacy of seeking to ‘teach’ or ‘develop’ the competencies sought by employers. Such efforts can appear to clash with the priority of the curriculum, rightly focused on the subject content of the degree. Our own initiative arose from reflection on the failure of: (a) previous efforts to prescribe a Progress File as part of an earlier Personal Development Planning (PDP) initiative and (b) extra-curricular employability initiatives to engage the majority of students. In terms of this study, the demographics of the student population were a key driver behind the embedded nature of the module described below. The University has a relatively high proportion of students from Widening Participation backgrounds, currently standing at 42% of the population (University of Salford 2017).

This paper suggests that while it may not be possible to instruct students in employability, we can guide them to construct their employability on the basis of their prior experience and knowledge. This knowledge comes from their learning both within and outside the curriculum. For this to be effective, students not only need to appreciate the importance of employability in general but, most importantly, they need to cross the threshold between *their* understanding *and* that of employers.

15.8 Threshold Concepts and Employability

Meyer and Land developed the theory of Threshold Concepts in 2003 following a research project into the characteristics of effective undergraduate education, particularly in the field of economics. Akin to a portal, achieving a threshold opened up a “new and previously inaccessible way of thinking about something ... it represents a transformed way of understanding, or interpreting, or viewing ... without which the learner cannot progress” (Meyer and Land 2003). Subsequent investigation has shown that the central tenet of mastery of a subject via Threshold Concepts could be applied to any subject, demonstrating the broader applicability of Meyer and Land’s original findings. In particular, work by Cousin (2010) has demonstrated the significance of Threshold Concepts in developing pedagogy as well as facilitating subject specific knowledge. A good example to illustrate new understanding would be a shift from a student of French to a French speaker (Cousin 2010).

This understanding, considered so important by their tutors, is both transformative and irreversible and is not (we would argue) discipline specific. Threshold Concepts have non-subject specific features in common. Their significance in HE has been explored more fully in a more recent collection of work published in 2016 (Land, Meyer and Flanagan). Flanagan (2017) has summarised the features of Threshold Concepts, in Table 15.1.

The concept of liminality resulted from additional research by Meyer and Land, published in 2006. Liminality involves the active engagement of the learner, as this threshold is crossed back and forth as the student experiences both positive and unsettling shifts in comprehension. Cousin (2010) compares this idea to the age of adolescence. In learning it can involve a period of understanding and misunderstanding at the same time: the experience can be very emotional. The first experience of a job interview is a good example: the experience can be very emotional involving a combination of understanding and misunderstanding. Effective performance in face to face interviews and other employer engagement (for example assessment centres and online video interviews) require a clear understanding of the employer perspective which, when achieved, is a threshold of employability. Achieving a clear understanding of employability meets the description of a Threshold Concept well.

Burch et al. (2014), in their paper on ‘Identifying and overcoming threshold concepts and conceptions’, stress the importance and difficulty of changing curriculum design and delivery in order to apply the theory. Cousin (2010) argues that a teaching strategy informed by the practical application of Threshold Concepts

Table 15.1 Summary of threshold concepts

| Threshold feature | Impact characteristics |
|-------------------|--|
| Transformative | Once understood, a threshold concept changes the way in which the student views the discipline |
| Troublesome | Threshold concepts are likely to be troublesome for the student. Perkins (1999) has suggested that knowledge can be troublesome e.g. when it is counter-intuitive, alien or seemingly incoherent |
| Irreversible | Given their transformative potential, threshold concepts are also likely to be irreversible, i.e. they are difficult to unlearn |
| Integrative | Threshold concepts, once learned, are likely to bring together different aspects of the subject that previously did not appear, to the student, to be related |
| Bounded | A threshold concept will probably delineate a particular conceptual space, serving a specific and limited purpose |
| Discursive | Meyer and Land (2006) suggest that the crossing of a threshold will incorporate an enhanced and extended use of language |
| Reconstitutive | Understanding a threshold concept may entail a shift in learner subjectivity |
| Liminality | Comparing the crossing of the pedagogic threshold to a rite of passage, involving a potentially messy journey to learning. Liminality requires active engagement of the learner, as this threshold is crossed back and forth as the student experiences both positive and unsettling shifts in comprehension |

allows academics to steer a path between teaching-centred and student-centred education, making it appealing across all disciplines. In practice this involves a change of focus from the teaching to the learning, with a particular focus on how the student can develop their learning on the basis of their prior experience.

15.9 Constructing Employability

In considering our best approach to guide the students towards the threshold of employability, the solution devised follows a constructivist approach to learning and teaching: our approach can be characterised by a saying attributed to Plutarch: “the mind is not a vessel to be filled, but a fire to be lit”. Boud et al.’s (1985) work on ‘Reflection: turning experience into learning’ was valuable. This advocates the use of assessment and reflection in motivating learning. Wiggins and McTighe (1998) also argue in their book ‘Understanding by Design’ for the idea of deciding first upon the outcome required and then designing the assessment accordingly. Boud suggests further that the teaching and assessment process needs to enable students to draw on their previous experience and knowledge so that they can ‘take some significant responsibility for their own learning over and above responding to instruction’ (Boud 1988, p. 32). As Villar and Albertin (2010) suggest, students need to become more actively involved and responsible for their education, investing in their own social

capital. Providing students with a better understanding of how to do this and opportunities to participate in student-driven activities can develop and/or demonstrate proactive personality in a practical way. These ideas have become well established in the work on Assessment for Learning (Sambell et al. 2013), promoting the idea of using assessment to promote learning rather than measure it. We were also influenced by arguments concerning the significance of authentic assessment, a term popularised in the in the paper with that title by Fook and Sidu (2010).

15.10 The Patchwork Approach to Assessment

We adopted a patchwork approach to assessment as a mechanism to force the involvement discussed above. This is similar to an assessment approach possible more widely known as scaffolding. Winter's paper 'Contextualising the Patchwork Text: Addressing problems of coursework assessment in Higher Education' (2003) explains the patchwork approach as follows. Academic staff define the module assessment as a sequence of tasks. The tasks themselves are a process of development designed to guide students to construct their own learning, whilst assimilating new ideas within their existing experience. The tasks are both analytical and experiential.

Winter (2003) likens this to bricolage, where the student is encouraged to improvise for each task according to the social, material and experiential resources they have to hand. In this case students are given quick feedback, and social feedback amongst their peers is encouraged.

Students are asked to synthesise the patchwork through self-reflection at the end. In arguing the value of this, Winter (2003) cites Barnett; "Only in that moment of self-reflection can any real state of intellectual freedom be attained ... Only through becoming a continuing 'reflective practitioner' can the student ... gain a measure of personal integrity." This offers the opportunity, as Moon (1999) suggests, for engagement in personal or self-development in addition to gaining insights and empowerment. This final review and interpretation thus also holds the possibility of students demonstrating achievement of a threshold of learning about employability.

15.11 Professional Development Module

We now discuss how this was put into action in a large 20 credit module (one sixth or a year of study) undertaken by 600s year undergraduates on 11 different degree programmes.

A patchwork of assessment (Winter 2003) was designed to achieve student enquiry into their own employability; assessments replicated typical recruitment processes. Figure 15.1 illustrates the assessment tasks of the module which closely follow processes used by employers in an application process. Academic and careers



Fig. 15.1 Professional development patchwork assessment

staff together with employers were involved at every stage—significant evidence is available of the benefits of such partnerships (e.g. O’Leary 2017).

These were initiated by an ‘as advertised’ job description for which students were set a mixture of tasks, commencing with a self-evaluation presentation to their class. They had to judge their stronger and weaker competencies in relation to the job description, drawing upon a wide range of experiences. They were required to give evidence of their strengths using the Situation, Task, Action, Result (STAR) approach, and an action plan to address areas of improvement. Individual feedback was given and (where necessary) support provided.

Students then created a digital profile using LinkedIn. Unlike other forms of social media, the focus here is upon a professional outlook that illustrates the students’ current and future employability, highlighting their skills, experience, extra-curricular activity, as well as a summary of their education.

Students then submitted a CV and covering letter for the job description provided. Evaluation of this work focussed on the quality and relevance of the application to the criteria specified, rather than the calibre of the student achievement.

All students were required to complete a series of industry standard Psychometric tests. These tests were treated as a learning opportunity to be reflected upon in the final assessment.

Students from many degree programmes were randomly mixed for an assessment centre which was conducted together with a company partner, aiming to be as authentic as possible, simulating standard practice in recruitment as used by approximately 90% of medium to large employers (AGR 2016). Each student was allocated to a group to tackle a business problem followed by a team presentation. The following competencies: leadership, teamwork and communication were assessed.

By once again using the ‘as advertised’ relevant job description provided, each student undertook a panel interview. The preparation of relevant answers and positive engagement was key for this element. The process was conducted as realistically as possible with appearance, punctuality and body language forming part of the assessment criteria.

At the end students were required to synthesise and reflect upon their experience and discuss how they would continue to develop their employability and articulate their competencies. The potential to reach the Threshold Concepts for employability permeated the whole development process. Importantly students were asked to reflect *not* upon the module delivery but rather their own experience of professional development. This could be a critical incident, or perhaps consequence of their performance, that they considered influential to their future development.

This sequence is illustrated in Fig. 15.1, commencing with the self- assessment, moving clockwise around the hexagon to the interview and then concluding with the student self-evaluation:

Every part of the assessment followed the cycle:

- (1) Introduction in lectures by academic staff and explanation by employers and other visitors of the thresholds they expected,
- (2) Seminar with opportunity for exercises and detailed explanation, plus questions and answers and feedback on draft work,
- (3) Submission of assessment,
- (4) Quick feedback online with opportunity for personal feedback in the following seminar or by appointment, allowing for development between tasks,
- (5) Final summative feedback following the final written self-reflection submission.

The delivery was resource intensive. Sixty staff, employers and postgraduate students were involved in delivering and assessing the module involving many issues of co-ordination, equity and moderation. Substantial deployment of technology in the assessment process (for example in psychometric testing), numerous different opinions, different cultural backgrounds, and many other ‘business as usual’ issues created significant complexity.

15.12 Presentation of Data

The data presented in this paper is drawn verbatim from reflective statements submitted by students undertaking the Professional Development module. It is presented anonymously with their permission. The reflective statements used formed the final element of the patchwork assessment developed for this programme of study (see Fig. 15.1). In this final element, students are encouraged and required to examine both positive and negative aspects of their professional development and are assessed upon quality of reflection rather than specific performance outcomes.

We have deliberately selected those statements (i.e. purposive sampling) that illustrate the argument of the paper. Threshold features were used as a framework for qualitatively analysing this data, taking a discourse analysis approach (Alvesson and Karreman 2000). It should be pointed out at this stage that a significant minority of students did not consider employability to be a subject worthy of time in the curriculum: these views are not used in the following section. The sample reported upon in this paper has not been broken down by specific degree programme or any other demographic parameter. In combination we suggest that the comments used demonstrate the attainment of intellectual freedom referred to by Barnett cited in Winter (2003) above. The selected quotes are intended to illustrate student insights that suggest the development of an appreciation of employability based upon threshold and constructivist concepts. They are presented as distinct sections but the content is all inter-related:

1. A number of students discussed how the module helped them come to understand the processes of employment:

“During the study of professional development module, I realised the importance of employability skills. I came to know that set of qualities, skills and knowledge that all newly graduates should obtain to ensure they have the skills of being persuasive in the workplace for their own benefit and their employers.” (S1)

“From this module, I have found that I have developed my key employability skills a great deal and now feel I would have a lot more confidence when applying for a job role. I have a greater understanding of what employers are looking for in an ideal candidate ... I have learnt that whilst it is important to possess skills such as teamwork, commercial awareness and leadership skills ..., it all depends on how you can demonstrate them to the employer by using key experiences and situations to evidence them.” (S2)

Both Student 1 and 2 make the point, also raised by others below, that first they needed to understand what was required, and then demonstrate their attainment of this. For some students, including S2, it was patchwork of the module assessment that gave her confidence.

2. Students discussed from their learning that despite initial misgivings, employability was a process of development,:

“At the beginning I was very apprehensive about it working as it is just a University module. However, from completing it I can say that I have improved in all aspects

especially compared to the weaknesses I pointed out I had during the personal skills assessment at the beginning of the module.” (S3)

“My journey through Professional Development has been an enlightening and welcome experience. My understanding of self-employability has had a boost and I now feel more confident with the knowledge of what makes me employable such as knowing that being able to demonstrate a skill is just as important as possessing it.” (S4)

“By taking this module my employability skills have improved drastically... When I started this module in January I had been rejected by five different companies that I had applied for a placement with. Towards the end I had two placement offers ... My employability has increased ... and my confidence has grown. By using Gibbs reflective model (1988) “to promote self-improvement and to link practice to theory.” I am able to improve my employability even further.” (S5)

It is interesting to see how many students refer to the development of their confidence, a theme discussed further below. What can also be seen is a level of commitment to the task and perseverance as the patchwork assessment progresses and that the relationship between the different recruitment processes was being understood.

3. As with Student 4 above, others discussed the importance of reflection and self-evaluation in the employability journey:

“I have learned from this module that regular self-evaluation is essential, that employability is vital to develop continuously, and that seeking and recording experiences that will enhance employability will enable me not to just have a career for life but be employable for life.” (S6)

4. Although students didn't use the term 'threshold' the quotes below suggest that this is indeed what some students reached.

“At the beginning of this module, I was terrible at reflective thinking and writing and didn't see the point in it. Then I began to understand the importance of reflection on everything I do within my life. It made me begin to think about why reflection was important and how it linked with my personal employability. It is important, within an ever changing world, to conduct regular reflections in both thought and in writing, to enable us to be the best version of ourselves that we can be.” (S7)

“Looking back at the whole experience of the professional development module, I have realised that the subject is a life changing experience. The module has made me become a better person with a strong persona ... Although things were hard at first, it got better with time. This module obviously gave me confidence... I believe after accomplishing this Professional Development module I am now able to manage myself as a professional. I have identified areas that need to be developed and I have already started building on them.”(S8)

5. As can be seen throughout the feedback, gaining confidence was the single most important achievement for many students. This in turn led to a development of their employability and realisation of a threshold:

“The module has helped me transition from an introvert to an extrovert in the working environment by developing my employability skills. I feel that everything I have learnt and developed throughout this module I can develop it by putting it into context when I am on my placement year and further down the line in my working career. From this moment on it is down to me as an individual to develop on the skills I have gained.” (S9)

“When I originally learnt the details of the course I was incredibly nervous as I never identified myself as a person who excelled in areas such as group and face to face interviews... However, having reviewed feedback from each assessed piece of work I have completed I realise my worries were without basis. ... This allowed me to identify my main weakness; and it wasn't my time management, my level of business awareness or my Microsoft skills; it was in fact my level of self-confidence. This module has enabled me to build confidence and has allowed me to demonstrate to myself that I can excel in a recruitment process... It has helped me grow in terms of my skills but also as a person and I believe my future employment prospects are improved thanks to the strengths and weaknesses I have identified and built upon.” (S10)

“Before starting this module I had low self-confidence and underestimated myself ... Overall my feelings at the beginning of this module were negative, however after completing assessments, attending lectures and doing my own research I have learnt a lot about myself in relation to employment. Through feedback and self-evaluation I have determined my strengths [and] this module has helped me develop along with weaknesses which I could improve. As a whole I have achieved well throughout the module which has helped build my self-confidence.” (S11)

“I had not given any thought to the expectations that employers will have from me and was focussed on attaining a good degree. This module has been an enlightening experience to my professional character ... The self-reflection I have continuously made in this module has made me realise where my strengths and weaknesses lie.” (S12) Whilst quantitative analysis of this module has not been conducted it is probably relevant that in the two years for which destination data is available for students who have undertaken this module, i.e. 2015–2016 and 2016–2017, there has been a 4% increase from 2014–2015 in the proportion of graduates reporting themselves in work or further study (increase from 84 to 88%). In the absence of further research it isn't possible to claim an association with the module.

15.13 Discussion

15.13.1 *Constructing Employability*

The evidence provided by these reflective statements illustrates how the module experience helped students actively construct their employability rather than passively learning about careers. Authentic assessment (Fook and Sidu 2010) was used as an

opportunity for learning (Sambell et al. 2013) rather than a simple measure. The Patchwork of assessment, linked to expert advice, feedback and reflection allowed students to build upon the development of competencies. This accords with Perkins' (1999, p. 8) characterisation of constructivism as an energising process of discovery, one that yields deeper understanding. In attempting to capture the form of active learning described by Perkins, the teaching and learning activities sought to allow students to discover or re-discover principles that fostered understanding with practical results. This is illustrated by comments from S4 who discusses using reflection for self-improvement, and consequently changing from receiving repeated employer rejections to securing new job offers.

It can also be seen that students (S5 and S11 for example) were learning from employer rejection and assessment feedback, experiencing phases of liminality within the process. By continually engaging in various activities, such as job applications and CV improvement, students as learners entered this liminal space, engaging with the mastery of employability (Meyer et al. 2006).

15.13.2 Achieving a Threshold

The reflective statements illustrate the transformative and irreversible aspects of achieving a threshold of understanding of employability. Understanding why an employer requires a combination of competencies is integrative, and can be morale boosting since the threshold of employability is bounded. For example, the employer may not expect detailed technical expertise from an employee but they may require someone with enthusiasm to learn and an appreciation of their own employability. Student statements highlight the transformation, with S4, S5, and in particular S10 stating their change in attitude towards the module, its teaching and their personal learning. The statements also demonstrate liminality aspects relative to attaining thresholds. The process of understanding is not without difficulty for learners: comments such as feeling "nervous" (S11), having "negative feelings" (S12) or "insecurities" (S13) were common and illustrate the disorientation typical of a state of liminality as discussed by Meyer and Land (2006). Students alternated between embracing and rejecting the module, between anxiety and confidence. Because liminality indicates a period of oscillation, by definition some students will not have realised the threshold of understanding of employability until later in their learning journey.

For students, this new understanding can be troublesome in a very specific way. For many employers, demonstration of competencies is valued just as highly as qualifications. Some major employers have explicitly said that qualifications are not their primary recruitment metric (Times Higher 2015); rather evidence of employability is of greater significance. Most students have been brought up to believe that the fundamental purpose of their participation in education is the achievement of the highest marks possible. As the final student (S13) says above "my focus was on attaining a good degree", but as S2 comments; "I have a greater understanding of what employers

are looking for in an ideal candidate”. Empathising with the employer perspective and achieving the threshold of employability involves assimilating this troublesome knowledge and this is an essential part of self-realisation and transformation.

15.13.3 The Importance of Confidence

The student feedback also demonstrates that a constructivist approach can foster confidence, which is an essential ingredient in achieving the threshold of employability. Other researchers have also noted the significance of confidence in the achievement of a Threshold (e.g. White et al. and also Berg et al., cited in Land, Meyer and Flanagan 2016). The module helped students understand the perspectives and requirements of employers, understand their own development, appreciate the importance of reflection, gain confidence and thus realise a threshold. As S8 comments, “I have realised that the subject is a life changing experience. The module has made me become a better person with a strong persona ... This module obviously gave me confidence”. Hawkins and Edwards (2013) discuss the ‘monsters of doubt’ that students experience when learning about leadership. These same monsters equally apply to employability. A key moment is when students understand clearly the competencies employers seek, why they are sought, and how to articulate them. This threshold may be reached after numerous job applications, assessment centres, interviews, or through real experience in employment. Discovering how to effectively articulate their competencies can be a lightbulb moment for students.

Many of the reflective statements submitted for assessment across the cohort include comments relating to increasing confidence. Once students grasp the linkage between their own skills and those required by employers, i.e. an integrative concept, they are creating knowledge *for* and *of* themselves that can be applied across the various aspects of employability. The confidence and perseverance we saw from many of the students was also an important aspect of maintaining engagement that Cranmer (2006) and Tymon (2013) suggested was lacking in other initiatives concerned with the development of employability in the curriculum. This is particularly important as it feeds into the benefits of an active and constructivist learning approach as discussed in our research.

Contrary to the work of Tymon (2013) and others, this paper suggests that HE can and should help guide students toward this self-realisation through the curriculum. This is demonstrated via a collaborative approach. To accomplish this does require a re-organisation of the curriculum (as suggested by Burch et al. 2014), starting with a focus on the final aim (the articulation of competencies), and working backwards through design of multiple assessments to achieve this aim. Substantial teamwork including the involvement of employers—was essential in guiding students to this final aim or threshold. This approach succeeded in engaging the great majority of students. It also enabled a mind-set change for some of the significant minority

to become engaged, to realise the importance of employability and how it applies to them. Overall it was thus much more effective than relying solely on ‘traditional’ extra-curricular support.

15.14 Conclusions

Demonstrating employability has become a central component of UK government policy in regards to Higher Education, and HEIs have made employability a key component of their mission. However, the ability to articulate employability skills, more commonly now known by employers as competencies, is not equally distributed and is related to the social capital that is held (Villar and Albertin 2010). Students have or can nurture competencies, but may not appreciate how to do this or how to express the same. This is where Higher Education can provide interventions and, to some degree, light the fire. If well designed, HE can help students construct their own employability, and thus enhance their experience of education more broadly. This paper has argued for the importance of embedding employability in the curriculum, guiding students to realise the threshold of employer expectations, and as a consequence construct their employability in more effective ways. The use of assessment and specifically, a Patchwork Approach (Winter 2003), is valuable in this process.

The paper explains the design and implementation of a module that has allowed students to recognise their competencies so widely cited as requirements by employers (Tomlinson 2012; Tymon 2013). This module has now been undertaken over a period of five years. Evidence of the efficacy of this approach is provided in excerpts from student reflections: further longitudinal and quantitative research would be necessary to gauge the long term impact. The single most important lesson to come out of the feedback from students involved is that confidence is fundamental to expressing competencies and demonstrating employability. Confidence is key to achieving the Threshold of Employability. Whilst the significance of social capital to employability is widely accepted, we were not aware of the importance of emotional (Cousin 2006) and psychological capital (Luthens 2007) for students in navigating the liminal space between HE and employability. These issues are clearly presented by Rattray (2016) opening up a whole body of work in Positive Psychology (e.g. Ivztan and Lomas 2016) which we haven’t addressed in the paper.

Valuable further research can be conducted on employability initiatives in HE which take as a starting point the development of student confidence built within the curriculum.

References

- AGR (2016) The Association of Graduate Recruiters Annual Survey at: <http://justoncampus.co.uk/wp-content/uploads/2016-AGR-Annual-Survey-2.pdf>. Accessed 20 June 2018
- Alvesson M, Kärreman D (2000) Varieties of discourse: on the study of organizations through discourse analysis. *Human Relat* 53(19):1125–1149
- Archer W, Davison J (2008) Graduate employability: the view of employers. Council for Industry and Higher Education, London
- Artess J, Mellors-Bourne R, Hooley T (2017) Employability: a review of the literature 2012–2016. Available at <https://www.heacademy.ac.uk/knowledge-hub/employability-review-literature-2012-2016>. Accessed 20 June 2018
- Barr N (2014) Shaping higher education: 50 years after Robbins. In: Barr N (ed). LSE, London.
- BIS (2016a) Graduate market labour statistics 2015. Department for Business, Innovation and Skills, London
- BIS (2016b) Success as a knowledge economy: teaching excellence, social mobility and student choice. Department for Business, Innovation and Skills, London.
- Boud D, Keogh R, Walker D (1985) Reflection: turning experience into learning. Routledge Farmer Oxon
- Boud D (1988) Developing student autonomy in learning, 2nd edn. Kogan Page, London
- Brown P, Hesketh AJ (2004) The mismanagement of talent: employability and jobs in the knowledge-based economy. Oxford University Press, Oxford
- Burch G, Burch J, Bradley T, Heller N (2014) Identifying and overcoming threshold concepts and conceptions. *J Manage Educ* 39(4):476–496
- CBI (2015) Inspiring growth CBI/Pearson education and skills survey 2015. London, UK
- CIPD (2014) Industrial strategy and the future of skills policy. CIPD, London, UK
- CIPD (2015) Policy report: over-qualification and skills mismatch in the graduate labour market. CIPD, London, UK
- Cole D, Tibby M (2013) Defining and developing your approach to employability: a framework for higher education institutions. Higher Education Academy. Available at: https://www.heacademy.ac.uk/system/files/resources/employability_framework.pdf. Accessed 20 June 2018
- Cousin G (2006) Threshold concepts, troublesome knowledge and emotional capital: an exploration into learning about others. In: Meyer JHF, Land R (eds) Overcoming barriers to student understanding: threshold concepts and troublesome knowledge. Routledge, London and New York
- Cousin G (2010) Neither teacher-centred nor student-centred: threshold concepts and research partnerships. *J Learn Dev High Educ* 2
- Cranmer S (2006) Enhancing graduate employability: best intentions and mixed outcomes. *Stud High Educ* 31(2):169–184
- Diamond A et al (2011) Global graduates into global leaders pub. AGCAS and others
- Dubois D (ed) (1998) The competency casebook. HRD, & Silver Spring MD: International Society for Performance Improvement, Amherst, MA
- Flanagan M (2017) Threshold concepts: undergraduate teaching, postgraduate training, professional development and school education. Available at: <https://www.ee.ucl.ac.uk/~mflanaga/thresholds.html>. Accessed 20 June 2018
- Fook CY, Sidu GK (2010) Authentic assessment and pedagogic strategies in higher education. *J Soc Sci* 6(2):153–161
- Gibbs G (1988) Learning by doing: A Guide to Teaching and Learning Methods. Oxford: Oxford Further Education Unit
- Greenbank P (2015) Still focusing on the “essential 2: 1”: exploring student attitudes to extra-curricular activities. *Education + Training* 57(2):184–203

- Guardian (2016) Editorial: the guardian view on graduates and employment: degrees but not destinations. <https://www.theguardian.com/commentisfree/2016/may/03/the-guardian-view-on-graduates-and-employment-degrees-but-not-destinations>. Accessed 30 March 2017, Accessed 20 June 2018
- Hawkins B, Edwards G (2013) Managing the monsters of doubt: liminality, threshold concepts and leadership learning. *Manage Learn*
- HEA (2015) <https://www.heacademy.ac.uk/knowledge-hub/graduate-attributes-framework>. Accessed 29 March 2018, Accessed 20 June 2018
- HEFCE (2017) Computer science employability and accreditation. Review led by Nigel Shadbolt. <http://www.hefce.ac.uk/skills/gradstemreview/csreview/>. Accessed 21 May 2018
- HESA (2017a) Higher education student enrolments and qualifications obtained at higher education providers in the United Kingdom 2015/2016. <https://www.hesa.ac.uk/news/12-01-2017a/sfr242-student-enrolments-and-qualifications>. Accessed 15 Dec 2017a
- HESA (2017b) NewDLHE: destinations and outcomes review. Accessed 27 June 2018
- Independent (2015) Leading employers prefer value work experience among graduates over grades. <http://www.independent.co.uk/news/education/education-news/leading-employers-prefer-work-experience-over-grades-says-new-research-10286829.html>. Accessed 31 May 2015
- Ivtzan I, Lomas T (2016) *Mindfulness in positive psychology*. Routledge
- Luthans F, Youssef CM, Avolio BJ (2007) *Psychological capital*. Oxford University Press, New York
- Matthews S (2017) Tech city talent and computer science graduate employability: an independent learning review for the tech partnership and the JPMorgan's chase foundation. https://www.thetechpartnership.com/globalassets/pdfs/research-2017/techcitytalentandcomputersciencegraduateemployability_report_may2017.pdf. Accessed 24 April 2018
- Meyer JHF, Land R (2003) Threshold concepts and troublesome knowledge 1—linkages to ways of thinking and practising in improving student learning—ten years on. In: Rust C (ed). *OCSLD*, Oxford
- Meyer JHF, Land R, Davies P (2006) Implications of threshold concepts for course design and evaluation. In: Meyer JHF, Land R (ed) *Overcoming barriers to student understanding: threshold concepts and troublesome knowledge*. Routledge, London and New York
- Meyer JHF, Land R (2006) Threshold concepts and troublesome knowledge: Issues of liminality. In: Meyer, J H F and Land, R (eds.), *Overcoming Barriers to Student Understanding: threshold concepts and troublesome knowledge*, London and New York: Routledge pp19–32
- Moon JA (1999) *Reflection in Learning and Professional Development*. Routledge, London
- NCIHE (1997) Higher education in the learning society. <http://www.educationengland.org.uk/documents/dearing1997/dearing1997.html>. Accessed 20 June 2018
- O'Leary S (2017) Enhancing graduate attributes and employability through initiative with external partners. *Pract Evid Sch Teach Learn High Educ* 12(3)
- Pegg A, Waldock J, Hendy-Isaac S, Lawton R (2012) Pedagogy for employability. Available at https://www.heacademy.ac.uk/system/files/pedagogy_for_employability_update_2012.pdf
- Perkins D (1999) The many faces of constructivism. *Educ Leadersh* 57(3)
- Rattray J (2016) Affective dimensions of liminality. In: Land R, Meyer J, Flanagan M (2016) *Threshold concepts in practice in educational futures: rethinking theory and practice*; Peters MA (Series ed), vol 68. Sense Publishers, Rotterdam/Boston/Taipei
- Sambell K, McDowell K, Montgomery C (2013) *Assessment for learning in higher education*. Routledge, London
- Times Higher (2015) Ernst and Young drops degree classification threshold for graduate recruitment. Available at: <https://www.timeshighereducation.co.uk/news/ernst-and-young-drops-degree-classification-threshold-graduate-recruitment>. Accessed 20 June 2018
- Tomlinson M (2012) Graduate employability: a review of conceptual and empirical themes. *High Educ Policy* 25:407–431

- Tomlinson M, McCafferty H, Fuge H, Wood K (2017) Resources and readiness: the graduate capital perspective as a new approach to graduate employability. *J National Inst Career Edu Counselling* 38(1):28–35
- Tymon A (2013) The student perspective on employability. *Stud High Educ* 38(6):841–856
- University of Salford (2017) TEF 2 provider submission. University of Salford
- Villar E, Albertin P (2010) 'It is who knows you'. The positions of university students regarding intentional investments in social capital. *Stud High Educ* 35(2):137–154
- Wiggins G, McTighe J (1998) *Understanding by design*. Association for Supervision and Curriculum Development, Alexandria, VA
- Winter R (2003) Contextualizing the patchwork text: addressing problems of coursework assessment in higher education. *Innov Educ Teach Int* 40(2):112–122
- Woods A, Dennis C (2009) What do UK small and medium sized enterprises think about employing graduates? *J Small Bus Enterp Dev* 16(4):642–659

Chapter 16

Exploring the Landscape of HE Industrial Placements Within Engineering and Technology Subjects. Observations of Recruitment, Graduate Attributes and Student Experience



Michael O’Grady, Fran Rimmer, Dan Everson, and Luke Swift

Abstract The School of Computing and Engineering (at The University of Huddersfield) typically places up to 200 undergraduate students annually across engineering, music, computing and games industries for a 12 month work experience. These thick-sandwich placement students return to final year with a wide range of experiences and skills, having worked regionally, nationally and internationally. Around 30% of students are offered a potential graduate opportunity from their placement company or partner, when entering final year (**High Fliers** High Fliers Research Limited (2018) The graduate market in 2018: annual review of graduate vacancies & starting salaries at the UK’s leading employers. https://www.highfliers.co.uk/download/2018/graduate_market/GMReport18.pdf. Accessed 27 Nov 2022). We set out a range of factors impacting Year 2 students in their search for a placement position—the barriers and benefits. This is from an academic, a Placement Team and employer viewpoint. We also conduct a brief desk-study of hard- and soft-skill attributes desired by industry for their new graduates and compare with those prescribed by the Chartered Management Institute for automatic accreditation to Level 5 of their membership qualifications (**Chartered Management Institute 2022**). Finally, we assess some common skills attributes from placement feedback, to tease out what the students valued in their placement year. General expectations from industry are for graduates to be experienced (or keen to develop skills) in: leadership; career-planning; emotional resilience; positivity; optimism; problem-solving; and project management. Returning student debrief comments (2017–2018, n_{students}

M. O’Grady (✉) · F. Rimmer · D. Everson · L. Swift
Department of Computer Science, School of Computing and Engineering, The University of
Huddersfield, Huddersfield, UK
e-mail: m.ograde@hud.ac.uk

F. Rimmer
e-mail: f.rimmer@hud.ac.uk

D. Everson
e-mail: dan.everson@hud.ac.uk

L. Swift
e-mail: l.swift3@hud.ac.uk

= 112, $n_{\text{comments}} = 101$) indicate that most develop some of these professional attributes, such as: self-reliance; resilience; responsibility; accountability; socialising; networking; and project management of live projects. Clearly, the placement year provides students with graduate-level skills and attributes prior to entering final year, enabling enhanced graduate outcomes.

Keywords Industry placements · Work experience · Sandwich year · Barriers · Hard-skills · Soft-skills · Graduate outcomes · Employment attributes · Higher education

16.1 Introduction

Collectively, the authors have 75 years' experience working with employment and/or placing Higher Education (HE) students in year-long industrial placements in the technology and engineering sectors. We have experienced the ups and downs of student abilities and expectations as well as the economic ups and downs of both the UK and European economies over the last 20 years or so.

We have noticed that student perceptions, experience and confidence to engage with the placement-seeking process is both changing and becoming harder for some. We are operating in a post-Brexit aftermath and exiting the Covid-19 lock-down phenomena which affected the Placement Team from around mid-2019. The effects of the latter are still being felt especially in terms of existing and new employer companies struggling with market volatility and the commitment to placement student employment.

In this Chapter, we firstly outline our operation of the one-year sandwich placement, which is structured and managed to allow full academic credits to be gained. Placements are generally paid, the typical range for 2017–2018 being £14–17k. The pros and cons of the effects on the placement student from an academic perspective are reviewed, particularly with respect to performance in final year of study and degree classification.

The experience of the Placement Team regarding student engagement with CV and applications, mock interviews, job search, communication and job interviews are also assessed, particularly regarding barriers faced by students. We also look at the feedback, benefits and barriers faced by a variety of employers that we work with.

Whilst placement students benefit enormously from the work experience year, we attempt to qualify what desirable attributes they gain. Hard- and soft-skill attributes are assessed from a generic employer Human Resources (HR) viewpoint, tabulated for individual and interpersonal attributes. We also take a brief look at Huddersfield's Global Professional Award (**The University of Huddersfield: Global Professional Award**) which outlines eight key factors for graduates to maximise their impact for employment including leadership, project management, emotional resilience and awareness of the world around them. We assess over 100 student comments from the returners of 2017–2018 and conclude that many grow perceptibly in soft-skills

as well as experience accountability and responsibility of ‘stepping up to the role’, gaining experience in problem-solving, project management and the importance of making contacts.

16.2 The Year-Long Industrial Placement

Many of our placement students work in the UK, but for several decades, we have served the wider areas of Europe and, beyond. Some further detail on the placement operation can be seen in Rimmer et al. (2019), however this Chapter deals generally with all our placement students, irrespective of location.

For many companies, the practice of recruiting placement students is an effective and reliable method for employing graduates of the future. Both students and companies commonly treat the placement year as an opportunity to prove themselves to one another. The student may hope that, by their performance, they will earn the offer of a graduate role and, similarly, the placement employer offers the student such a positive experience during placement that the student is willing to make a long-term commitment to the organisation.

Universities have increasingly supported employers in the recruitment of placement students. This benefits HE institutions because they can offer courses with industrial experience, which is a positive selling point when recruiting new students. The placement year also adds to students’ professional and technical skills which, in turn, improves student performance in their final year. Finally, students, because of the behaviours and skills they have developed in placement, become more employable which has a positive knock-on effect in league tables and graduate employability outcomes.

The Placement Team in the School of Computing and Engineering at the University of Huddersfield delivers a wide range of support for its students which is covered in more depth later in the Chapter. The team have five full-time equivalent staff to support all stakeholders through the process of the industrial placement year. This section will focus on what happens with a typical student’s placement year. It is quite different from the shorter and more informal Internship form of work experience, typically found in the USA (**Hora, Parrot and Her**).

16.2.1 *Pre-placement*

Support is offered to placement employers through a consultancy and recruitment support service managed by the Placement Team staff. In recent years, universities have chosen to distance themselves from employers by using recruitment portals (employers post vacancies and students apply directly to companies via an online resource), but the Placement Team choose to work directly via telephone, video call and email with companies’ recruitment staff and department managers undertaking

the recruitment. By operating this way, the team get to know the culture of the organisation, the values, behaviours and technical skills they desire in new hires. This helps when matching employers and students to find one another. Getting to know students is an essential part of the jigsaw.

Operating this way helps staff ensure that the quality of placement experience being offered to students is suitable as an assessed part of their degree. Placement students need to operate in a role that is directly related to the modules they study but, occasionally, job descriptions do not provide that. Where that is the case, placement duties and responsibilities can often be adjusted to be a better fit for the course. The relationship-building and discussions held with recruiters also prevent students applying to the increasing number of placements that appear online where the actual employer, location of placement, duties and responsibilities of role and salary are so unclear, it leads the experienced placement professional to assume that the placement doesn't exist.

When companies are ready to progress with interviews and job offers, the Placement Team manage that process, further cementing a positive relationship with employers and saving them hours of coordination and follow-up in arranging interviews with students. The interaction between placements staff and students to arrange interviews is also useful in relationship-building and advice-giving. On occasion, students have two or more good job opportunities rolling at the same time, and the more interaction that they have with Placement Team, the more likely they are to make good decisions.

16.2.2 During Placement

Support for employers and students continue during the placement year. This provides opportunities for relationship-building for all parties concerned.

Students are visited twice in placement by a Visiting Tutor—an academic who is experienced in coaching both industry partners and students to successful placement outcomes. During these visits, employers are given guidance on how to get the best from their placement students as well as discussing how the university might support them in broader organisational objectives via further placements, research or Knowledge Transfer Partnerships. The students are coached through the process of assignment writing and submission, succeeding at work in hard- and soft-skills, keeping an effective placement log and addressing any challenges or difficulties. The work of the Visiting Tutors is greatly valued by employers and the Placement Team.

We maintain contact with placement students, informing them on upcoming assignment deadlines, holiday closures, changes in arrangements and what to expect when they return to final year of study. The placement team also ring the students in placement twice a year with a welfare call to check on how they are doing. Occasionally, issues and difficulties are raised by students, so we will work directly with them and, if necessary, the employer and colleagues in wider support services (Guidance, Disability and International offices) at the university to address these issues.

The most common outcome being that the challenges are resolved to the satisfaction of all stakeholders. On the rare occasions when challenges cannot be overcome, the student leaves the placement but with the continuing advice and support of the Placement Team and colleagues from other supporting services within the university.

16.2.3 Return to Final Year Study

Students return to campus during Fresher's Week and attend a debrief lecture where they are given a final opportunity to provide feedback on their placement experience and listen to presentations from both a subject-specific Careers Advisor and the Associate Dean of Teaching and Learning. They are reminded of their step-change in work experience and encouraged to bring as many of their new career attributes to bear on their studies as possible. The authors note that time management, creativity, ambition, communication and team-working are more evident in placement returnees than those moving up directly from Year 2.

Returning placement students are given opportunities to act as peer mentors to current Year 2 students. The placement team host a Q&A session in Term 1 where Final Year (placement) students share their responses and reflections to Year 2 questions about the placement search and their experience of industry. It is a very popular session where both year's students enjoy contributing their experience.

16.3 HE Teaching Experience

Placements are promoted at our university Open Days, Applicant Visit events and special Taster Days (where schools are invited in for teaching sessions) and are the first time that the year-long placement year is promoted. During Year 1, different staff have a different approach and interest in promoting the 'sandwich' year but by Year 2, formal timetabled sessions are provided to equip and acquaint students with what they need to do to engage, and why they should engage.

As a department, we attempt to make assignment work as close as possible to the work environment. This includes a wide range of assignment-based tasks, broadly offered to the cohorts as part of the process to becoming a 'professional student'. Such work, for Web students for example, may include:

- Email signature and general message etiquette. It's surprising how unprofessional and confusing (and seemingly rude) some student emails can be;
- Creation and management of a professionally-developed LinkedIn profile page;
- Some coursework geared around making a website portfolio with the express purpose of appealing to employers. Selling, or marketing themselves in a positive light and getting them to reflect on their skills can be quite challenging. They are

told that employers want someone with personality, interests and 'a life' and not just the technical hard-skills;

- Students are encouraged to actively research freelancing their new skills and to this end prompted to set up an e-commerce shop facility whereby they can advertise for making and modifying websites and undertaking Search Engine Optimisation work, all for a fee;
- Live/real clients are arranged for some team projects, requiring teams of students to compete to create websites with the aim of the best version being chosen to be made live. Students really value meeting with clients: 'in the flesh' in class, in phone calls and video meetings;
- Random emails from local business owners suggesting that 'my new website would make a great student project' are sent to appropriate cohorts for them to liaise with the potential client and engage in freelance work. Many of these external clients expect to pay the student a fee for their time and effort;
- Asking one or more final year students, who have been on placement, to talk with Year 1 students, with the intention of making it an aspirational and motivational target;
- Put on a yearly Showcase event where the Web-based course students (formerly Interactive Multimedia) put on a show of their work in early May with local/regional employers invited. In the past up to 40% of students have received either job offers or requests to attend for interview from the event.

16.3.1 Placement Year—Academic Benefits

There are three types of student, who in their first two years, are facing the competitive challenge of winning a placement position. Those:

- who already have the confidence and open mind to work through their degree with the best outcomes. We are 'preaching to the converted' and they have probably held previous part-time jobs prior to or during HE;
- who are not committed and/or don't have the confidence to engage fully and on-time with the application process, but who manage (possibly contrary to their own expectations) to get a position anyway; and those
- who are too anxious, don't want to or expect not to get a placement position and so don't try or are self-defeating during the process.

The latter category contains a potentially wide range of outlooks. From mature students, possibly with family, outgoings, savings and trying to minimise time in education, knowing that they already have a wide range of experience and soft skills. They typically have a suitable reference lined up and can step straight into work dynamics and relationships. Additionally, the group of students who we specifically want to help via future research, suffer with a range of issues, including: emotional immaturity; poor communication skills; anxiety; nervousness; and trepidation. They

perhaps fear the application process or of living/working away from home. It is easier for some to 'put off' finding a job until after graduation.

When students return to final year of study therefore, there is a mix of older/mature-minded/experienced placement returnees and those who went straight to final year from Year 2. Purely as an observation in class dynamics in a final year team module, there seems to be some element of barrier with non-sandwich students talking and socialising with placement students, perhaps analogous to Junior School cohorts reluctant to socialise with older, more experienced and confident pupils in higher years. When allowed to self-select their own teams, the two groupings diverge, tending to stick together and not mix. This can be quite noticeable in the progress and management of teams as well as the final outputs; the better work coming from the sandwich students. The lesser-able students tend to stick together because they are comfortable working with 'similar' students whilst placement returnees tend to want to work with high-performing similar students with the aim of achieving high marks.

For those on placement, they undertake two pieces of coursework including: an overview of the company, industry and competitors; as well as providing a detailed description of one aspect of work that they have completed. Generally, their written work does not achieve as high a mark that their actual placement work achieves (as rated by their employer), but overall, the final aggregated placement-year mark is typically better than their Year 2 average and frequently in the 70% plus 'First Class' range. This will be the subject of future research. This is frequently noted in our Course Assessment Boards, where year marks and degree classifications are agreed. When taken, the sandwich year accounts for 25% of the degree classification.

Personal observations from the classroom revolve around comparing the students' personalities and outlook before and after the placement year. Students moving from Year 2 to Final Year without a placement seem to exhibit little change over the intervening period, however we can notice significant changes in some placement returnees. Many of these include:

- time management—scheduling and planning of learning and assignments;
- more purposeful—career direction, choice/change of optional modules, many have found their 'passion' within the industry;
- better at working in teams—more willing to be 'team leader' or 'project manager', they produce high quality work and to schedule. Some are so good that they don't need a leader;
- more focused, determined and analytical at getting the best grades for their degree classification—more willing to work within the academic rules and work strategically to get the best marks;
- more attention to detail with their personal appearance—haircuts, beard trims, clothing—generally more office-presentable;
- clear ideas and a possible live client/real project idea for their final year individual project;
- possibly working part-time hours for their placement employer (generally seen as a positive, but this can be over-done);

- around 30% returnees have the promise of their first graduate job with the same company, so the 'job search pressure' is eliminated as graduation approaches in final year.

The attainment level for sandwich students is typically one classification level higher than without it. In general, placement students exit with a First or high Upper Second Class degree. We see that their year in industry either raises their overall degree mark or it becomes a pre-cursor to an even better final academic year mark.

Anecdotally, depending on the academic year and variations within cohorts, we tend to have around a third of final year students sitting on a job offer from their placement employer. Whilst this is often a good full-time first job for new graduates, it undoubtedly is a good position from which to find a second job.

16.3.2 Students not Taking a Placement year

Part of the motivation for this Chapter is to try and assess some of the fundamental aspects why students do not want to engage with the placement process or don't show full commitment. Sadly, on first appearance, it seems that we have students who enter our degree with adequate academic qualifications but who are not as confident of themselves as we would like. This is evidenced in several areas including:

- an unwillingness of some to speak openly in class;
- unwillingness to have a webcam operating in a screencast or video meeting;
- lack of engagement in team project discussions and digital communications;
- difficulties reflecting on themselves, their performance and any shortcomings;
- 'hiding' in group video meetings by not activating a camera or microphone and not engaging in text chat;
- 'hiding' generally in peer assessed group work; and those
- preferring the distraction of their mobile phone during class-time.

Some students in Year 2 openly admit to not having the confidence in their academic skills to take on a full-time placement job or to being too nervous or anxious to engage with the recruitment process. There can be some issues around a defeatist attitude competing with peers who they rate better for the position than themselves. Some are put off by the effort involved in research and writing a company-specific letter along with their application and some will give up too early following a few rejections or unsuccessful interviews. Many do not realise the depth of company research needed to reflect to the company representative at interview: not seeing the need or knowing how to research appropriately.

Whilst we have systems in place to get feedback from returning placement students (outlined below), there is much more difficulty getting informative feedback from non-placement students. So we will be conducting future research with our students about their fears, anxieties and real/perceived barriers to engaging with the placement year.

What we do find in the final year classes is that the placement students are more willing to engage with whoever they need to, whilst non-placement students tend to remain in groups/friendships formed in Years 1 and 2, perhaps validating and condoning their non-placement status. Some hold the returnees in awe (or as a 'class above') as they achieved a double success: a placement job and the tremendous experience and knowledge that was gained.

16.4 Placement Team Observations

Our School places students across a wide spectrum of the STEM industry areas including: Engineering (design, manufacturing and planning); Computing and Information Systems (programming, help-desks, design, games, data, web); and Music (recording, stage, events, design and programming). The Placement Team work with three key stakeholders—students, employers and the university itself. We will take the opportunity to address two of these stakeholders as follows;

Students—we provide promotional sessions at university Open Days, Applicant Visit Days and Taster Days, timetabled class sessions to Year 2 students on the placement opportunities and how to engage, a CV checking and dummy job interview provision, management of a jobs database, manage and promote placement positions from new and existing employer relationships, find and promote external web-advertised placement opportunities. We manage the student through all these new processes, manage the visiting tutor visits, manage the written coursework aspect of this educational-credit year, and any emergency issues such as performance, job security, student health and wellbeing.

Employers—we offer considerable support to employers—promotion, communicating with technical and HR staff, pre-placement training and responsibilities, role advertisement, interview management, liaising with visiting staff, answering employer questions and managing performance assessment, expectations and student-based issues.

What follows in this section are our observations of the **students** as they pass through our placement system and, closely allied to this, the following section reviews our experience of contacts with the many individuals and companies providing placement opportunities: the **employers**.

16.4.1 Student Engagement

The Placement Team organise and deliver a series of placement preparation lectures in the first semester of year two. We see the biggest attendance of students within the first few lectures, in particular the *Introduction to Service* and *CV preparation* lecture. As we move further into the semester attendance slowly drops but picks back

up again when we have our *Employer* lecture where we invite companies to speak to students. The Q&A with final year students (returning placement students) where they talk about their experience of finding and working in placement is also well attended.

During these lectures students have the opportunity to ask questions to understand the content we are teaching them. At the end of every lecture, we are approached by several students with further questions which reinforces how engaged they are with their placement search.

The Placement Team also offer on-line and in-person drop-in sessions for a total of 12 h a week. In these sessions, placements professionals are available to work with students on an individual basis to address personal questions and challenges in their placement search. The online drop-in sessions are consistently attended throughout the year by students, though engagement slows down as we enter the summer period.

We find that familiar faces attend regular drop-in sessions, utilising the resources, making connections and developing a rapport with placement staff—a good precursor to the workplace. However, we are eager for more students who we have not spoken to before to join these sessions so that we understand where they are at with their placement search and what support they need. It also helps us to notice if there are any patterns/gaps in knowledge between students so that we can directly address these to look at ways to improve the service we offer. We actively encourage attendance by inviting each student every day via email, signposting them to drop-in sessions in our conversations with them and publicising them at lectures. We sometimes utilise the direct message-selling of course and module leaders to encourage students, particularly if a certain cohort is not engaging or if a slew of similar positions are available with inadequate student engagement.

Online drop-in sessions are extremely accessible to students because they can join using any device that supports Microsoft Teams (previously Zoom). This technical functionality and willingness to engage with video conferencing technology has been noticeably more effective and used since the lock-down-learning that we all had to move to over the COVID years. This means students can join a drop-in from wherever—for example, they may be at home or somewhere on campus, travelling, between lectures (or even on an unofficial holiday) when they speak to us. Overall, the highest quality individual support that the Placement Team offers is via these drop-in sessions.

16.4.2 Curriculum Vitae (CV) Writing

CV writing is a critical part of the placement search. Without a professional-looking version, several students would want to send the CV they developed during school/college or when they've applied for part-time jobs. This would severely impact the likelihood of them receiving an interview, which is ultimately the main purpose, aside from selling themselves.

During Term 1 the Placement Team give a lecture focussed on CV writing to teach students how to be effective in their self-promotion. This is when we task them to update their CV using the guidance taught in the lecture, alongside the resources available on our Virtual Learning Environment (VLE) before they submit their CV to us for feedback.

There is a significant minority of students who do not engage with the resource we provide before submitting their ‘best first attempt’. For example, the layout of their CV may be completely against the advice that we provide, it may lack detail or professionalism. This proves challenging to mitigate each year. If a CV is submitted without using the provided CV Checklist, then the CV is sent back to the student and they are directed to explore and utilise the resources on our VLE before resubmitting.

Generally, students are slow at making improvements based on the feedback provided to them. The most engaged students will use drop-in sessions in order to get immediate feedback before resubmitting their CV. The least engaged students will send their first draft in semester 2, after the initial deadline in semester 1. Often the reasoning for this is because:

- they were busy with their studies and assignment deadlines;
- they were not sure if they wanted to do a placement; or
- they have just seen a job advert we have emailed in [January onwards] that has sparked their interest.

16.4.3 Placement Database Engagement: ePlacements

Some students are heavily reliant on our ePlacements jobs database when searching for a position. This software application helps staff and students filter through and manage a journey through both a generalised and more-specific job search, via the many hundreds of positions entered by the Placement Team. Some will only use it rather than expanding their search through other sources such as Gradcracker (www.gradcracker.com), RateMyPlacement (www.ratemyplacement.co.uk), Indeed (uk.indeed.com), LinkedIn (www.linkedin.com) and other more-speculative approaches. Note that web addresses are relative to a UK location.

The ePlacements application is beneficial because it allows the team to support students in their placement search by directly advertising opportunities to them, so they are more visible. This is another advantage of students having high levels of engagement and developing a relationship with staff.

It is common for students to misunderstand how to apply for opportunities on ePlacements by not carefully reading the guidance we provide. For example, some students neglect to read the ‘how to apply’ section carefully in a vacancy, often presuming the Placement Team will send their CV to the company even though the vacancy may require them to apply direct to the company’s site, or that they are required to upload a covering letter.

16.4.4 Mock Interviews

The Placement Team offer mock interviews to students who have secured a formal placement interview. These are to better prepare students by providing constructive feedback on how they interview with a Placement Advisor. The feedback is varied between students dependent on the way they present themselves and answer questions.

Initially, a number of students are weary of, and don't recognise the value of a mock interview until they have completed one. After they have attended their formal interview, comments such as "I am so glad I had prepared through having a mock because I felt more confident and better prepared for the questions the company asked me". Some student's reluctance to book a mock is influenced by their level of confidence or apprehensiveness to be evaluated. However, this perspective does change after receiving a mock interview. The Placement Team commonly receive feedback from students expressing the usefulness of mock interviews, especially those who have never done a formal interview before, not knowing what to expect and how to be during an interview. Hence their apprehensive nature before booking a mock interview. Alternatively, there are students who have familiarity with interview processes and answering questions. These students are less likely to book a mock interview as they have confidence in their capabilities. Though, this confidence doesn't necessarily translate in the student being well-versed in suitably answering questions relevant to placements, as the Placement Team have encountered many experienced students who, despite their confidence, fall for common 'mistakes' when answering interview questions. Typically, their previous interview experience relates to part-time work or other informal work-experience, so they benefit considerably from the mock interview. Even students with experience of formal interviews will book a mock as a 'warm-up' or 'practice run', often allowing them to gain confidence and learn new ways in which to better prepare for a position.

Overall, students leave the mock interview with (further) confidence for their upcoming formal interview and consequently tend to perform better in their formal interview. This is noted by many employers who note that our students are better prepared compared to graduates who haven't been through the formal interview process before or haven't undertaken a placement year prior to graduating. This demonstrates the effectiveness of mock interviews in preparing Year 2 students at a level beyond a majority of graduates.

It is however not uncommon for employers to feedback that some students were ill-prepared, having not researched the company adequately. Typical comments are:

- "... not understanding what service we provide";
- "... not aware of how our company functions";
- "... unaware of what our goals/strategies are and how this relates to being successful in the role".

Employers want to know how a student's research has inspired their interest in applying for the placement other than a standard response of "I would like to gain

professional industry experience to help me develop myself into a career within your industry”. For many undertaking inadequate research, and/or not engaging with the mock interview process, employers have noted that:

- some gave weak answers;
- some didn’t understand the question and gave an irrelevant answer;
- some were focused too much on one example; and
- for some, their technical (hard-skill) understanding was a weakness.

16.4.5 Company Interviews

Where the Placement Team have a partnership with an employer, we offer a service of coordinating and organising interviews and contacting and managing student attendance. The majority of students who have been invited to an interview will confirm they would like to accept and then attend the interview itself. If this is the student’s first interview, they will often reach out to book a mock interview to help prepare themselves.

Students usually do attend their agreed interview. However, when this does not happen, they will be notified by the Placement Unit that they need to speak to a Placement Advisor before they can attend any other interview. This approach helps to ensure that we are aware of the reasoning behind their non-attendance and allows us to manage employers’ perception of our students and service. Repeated unprofessional actions (or inactions) by students risk diluting the links and commitment an employer has to us and our service. Therefore, we must have a serious discussion with students who do not attend their interviews (in particular, who do not communicate in advance) so that they understand their responsibilities and accountability in the recruitment process.

16.5 Employer Observations

The Placement Team need to do two things well: attract, manage and maintain employers’ willingness to provide placement positions; and to manage a wide range of students to engage responsibly (and hopefully professionally) with the placement process. For the student, this cycle of engagement is between 18 months and two years. For employers, this is hopefully an ongoing yearly relationship where we provide students who meet or exceed their expectations and part-manage the placement student whilst in-situ with the company.

For many companies open to the employment of students, the year in industry sits comfortably between strategies of years-long commitment to apprentices and the contractual employment of new talent on a full-time basis, probationary periods notwithstanding. The 12-month placement period is ideal for many—long enough to get some productive and creative outputs from the student and short (and contractually

final) enough to take a chance on raw talent. Here are some of the benefits, as well as barriers that companies new to the placement student concept, can have.

16.5.1 Company Benefits

Many companies and organisations of all sizes and structures recognise the value of offering sandwich placement opportunities to undergraduate students. In the short term, students can get involved with delivering projects and tasks that contribute to the work that the organisation delivers for their customers and stakeholders. In the medium-term, many placement students can pick up the lead and/or management of tasks and projects. This can free-up their teams and supervisors to continue with other work.

In the long-term, placements are valuable to employers because it is used as a recruitment timeline. That is, they treat the placement as a year-long job interview and often maintain commercial contact with part-time hours during final year. Additionally, where everyone is agreeable, they often recruit the student on graduation. Compare this with the challenges of graduate recruitment in the open market (competition from other companies, salary inflation and struggling to reach the right candidates), and we start to understand the valuable role that student placements play in early graduate-role recruitment. One of the hidden advantages for companies is that students want the placement experience so much and feel appreciation for the knowledge, trust, training and responsibility gained, that they become very loyal, thereby reducing employment churn.

16.5.2 Barriers to Placements

We support hundreds of companies each year and many express a frustration in recruiting enough of the right quality of students for placement roles. Companies know that they want more staff similar to their best employees, however, Year 2 undergraduates can be very naive and unsure of themselves—raw talent ready for shaping and mentoring. Indeed, many employers value the commercially-unformed and creative ‘learning sponges’ in which to train placement students in the way that they operate. Many see the opportunity of ensuring that students do not learn bad habits, such as not writing adequate comments into their programming code. However, whilst finding the right student is a challenge, companies present their own barriers to attracting students. Whilst we place students with Blue Chip companies, Non-Government Organisations (NGOs) such as charities, Colleges and Universities, there are perhaps more barriers for placements in public sector companies and small enterprises, so we cover these two in greater detail here.

One of the barriers to placement is not having had some or adequate previous work experience. Many students join HE and take on part-time jobs to help with living

costs. Whilst many jobs can be seen as part of the no-contract gig economy, they do afford much in the way of: customer interfacing; learning new skills; taking on responsibility; time-keeping; self-management; accountability; and responsibility (**Whittard, Drew and Ritchie**). Unfortunately, many of these positions were not available during Covid-19 lockdown and have been slow to re-open for students, affecting, for some, two consecutive years of HE study.

16.5.3 Public Sector

Public sector employers are in a strong position to offer students placements that will enrich their careers, build their skills and support students through a process of professional and personal development. They tend to offer structure and security as employers and patience when supporting students to improve their skills.

However, students on technology-focussed courses frequently perceive public sector employers as stuffy and unimaginative. They are nervous that the cutting-edge technology and products that they are eager to work with won't be found in health authorities, local councils or in schools and universities. Often, students are keen to work with either blue-chip companies or start-ups. Failing those, small agile companies where a variety of projects seem to be offered is more attractive than the prospect of working in the perceived inertia of the public sector.

Public sector organisations are frequently tied to a formal recruitment system and procedure. Commonly, this will be an online portal where students will have to submit lengthy examples of skills required for the role—competency-based recruitment. Frequently, students say that they don't have the time to complete such applications. They need to balance the return on investment (ROI) of a recruitment activity and may feel that the investment of significant time is not balanced by the chance of securing an interview. Our experience of working with health authorities and local councils, for example, is that of frustration from the recruiters that they are tied to cumbersome recruitment portals that systematically filter applicants out of the race via attrition, rather than encouraging participation and applications.

16.5.4 Small to Medium Enterprises (SMEs)

SMEs face significant challenges in recruiting students with the requisite technical and professional skills for their organisations. During the academic year 2022–2023, SMEs have been reporting that students are interviewing less effectively than they have in previous years. Fewer students are articulating well-structured and detailed examples to interview questions and an increased number don't seem to understand the formality and good quality preparation required for a job interview. This seems to be an impact of the Covid-19 pandemic. Students applying for placements in 2022 are the same cohort who missed out on their traditional (and sometimes first) part-time

jobs at pubs, restaurants, shops etc. So they have not had the practical experience of interviewing, working and gaining various soft-skills and confidence. These first jobs are where the rough edges get smoothed away and, in the absence of that experience, students appear to be less capable and/or less confident.

In SMEs there is less room for error in recruitment because the student directly and immediately contributes to paid projects for clients. SMEs are more cautious in the recruitment because they perceive that they must get it right first time. This caution, in addition to interviewing students with less work experience, is causing companies to be hesitant in their recruitment.

In recent years there has been a shift in attitude from students who are now more savvy at manipulating their placement search to serve their own ambitions. There is an increase in the willingness of students to play companies off against each other for improved salary or conditions, to keep companies waiting on placement offers in the hope that something better comes along or reject placements they have already accepted to pursue better options. There are often conversations with students around responsibility, reputation and acting professionally, not least for the Placement Team to maintain good working relationships with these companies for future cohorts.

Generally, SMEs advertise their placements in a comparable calendar window of March–June. With the end of the tax year approaching, companies become more confident on their financial standing and whether they can recruit a placement student for the following year. Companies then place their adverts at a similar time and the same students can end up applying for multiple placements. The challenge then is to manage those companies wanting to interview and recruit the same students at the same time. Recruitment in this window is challenging and can sometimes be frustrating for colleagues at SMEs.

A general reflection is that the best (employable and motivated) students tend to secure their placements early in the cycle, resulting in increasing uncertainty for those remaining, as they scramble for a reduced frequency of positions over the summer with the ever-increasing possibility (and anxiety) of returning direct to final year study in September.

16.6 Graduate Attributes—Employment Websites

Our sandwich students tend to perform better in final year than non-placed students and many (typically around 30%) pass through final year studies with the knowledge and security of having a graduate job offer from that company. Clearly the job-related hard-skills as well as the personal development (soft-skills) play a part here. It would therefore be useful to know what graduate employment attributes are desirable from an employer's viewpoint and to assess how the placement year contributes. This Section seeks to do so.

There is an additional layer of employability-training open to Huddersfield students via a new initiative in the form of our GPA. This is a three-year provision of training materials and experience to enhance employability: leadership,

emotional resilience, entrepreneurship, positiveness, wellbeing as well as increase global and social awareness (**University of Huddersfield: Global Professional Award**). The initiative is accredited with the CMI, achieving Level 5 status on successful completion and is assessed in more detail below (**CMI**).

A review of comments from some returning placement students (2018, pre-Covid-19) is presented in the following section, but before that, a short desk-top study of several employment and job-advice websites is reviewed in terms of graduate hard- and soft-skills. A web search was conducted (access date: 1st November 2022) with the intention of reviewing the types of skills desired in graduates, across all sizes of companies in a wide range of employment sectors. It seems a fair assumption that placement student skills will be quite close to graduate desired skills and may stand up to close scrutiny and comparison.

16.6.1 Hard Skills

There were some interesting results for hard-skills across job and employment sites such as Indeed (**Indeed: Hard Skills versus Soft Skills: Definition and Examples**), Valamis (**Valamis: Hard Skills versus Soft Skills**), Resume Genius (**Resume Genius: Hard Skills versus Soft Skills: Definitions and Lists of Examples**), Zety (**Zety: Hard Skills versus Soft Skills: List of Examples for Your Resume**) and Asana (**Asana: The difference between hard skills and soft skills: Examples from 14 Asana team members**). Many hard-skills are repeated through the articles; here are some of the most common categories and relatable attributes:

Computing/Technology—MS Office, computer skills, video editing, gathering software requirements, IT skills, cloud computing, artificial intelligence, machine learning;

Data—data analysis, data mining, interpreting data;

Marketing—copywriting, search engine optimisation (SEO), marketing skills, affiliate marketing;

Project Management (mentioned several times but no specific sub-skills);

Finance—financial planning, blockchain, financial modelling, business analysis; and others mentioned such as **Trouble-Shooting, Languages, Analytical Reasoning and Design**.

We believe that many students approach the placement year focussed on learning the hard-skill content and experience of their chosen sector, such as:

- industry-standard procedures and workflows;
- using industry-standard software and equipment; and
- undertaking real-life work for real clients.

Whilst some of the websites seem to be somewhat technology-focussed, we were surprised by the degree and specificity of computer-related skills such as video editing, artificial intelligence, data-related skills, SEO and blockchain in, what appeared to be, non-specific sector write-ups. There does seem to be a shift towards using computing applications to assist in everyday business life across a wide range of industries, such as data analytics and an acceptance that Artificial Intelligence (AI) and machine learning will in some way impact on current or near-future working lives. Digital literacy is clearly an essential and ever-evolving area.

16.6.2 Soft Skills—Personal

The list of soft-skills mentioned in online websites (referenced in some of the above articles: **Indeed**, **Valamis**, **Resume Genius**, and **Zety**) and augmented with others (**LinkedIn: Soft Skills between your success and failure** and **Making Business Matter: 8 Soft Skills and Hard Skills Research Conclusions**), list far greater numbers of attributes than hard-skills. However, very little is mentioned of the difference of personal/individual skills (such as optimism, tenacity, creativity and critical thinking) and those involving other people—interpersonal (such as negotiation, communication, teamwork and leadership). The following attempts to sift through the combined soft-skills into these two categories: personal; and interpersonal.

Over 30 soft skills were pulled from articles in the above-mentioned websites (there are many more!). When similar attributes are aggregated (e.g. 'communication' with 'communication skills' and 'time management' with 'time keeping' etc.) and split between personal and interpersonal, it becomes clear that many of the skills expectations are related to working with other people. Table 16.1 shows the personal/interpersonal split for 33 soft skills attributes.

Whilst many personal soft skills are given, some could be considered in both a personal and interpersonal context. Clearly, ambition, emotional intelligence, optimism, patience and self-awareness are personal attributes, but those of accountability, adaptability, critical thinking, flexibility, organisation, problem solving etc. could easily reside in both personal and interpersonal.

What is apparent from those writing the online articles (above), is that graduates should be positive and optimistic people who can solve problems, think critically, be able to be accountable and responsible whilst being emotionally well-grounded. Some of these attributes can be learnt from an early age and not easily taught from an HE viewpoint, however, if not experienced by family and friendship role models, they are certainly impacted on by their managers, mentors and peers in the placement workplace. As we'll see in the next Section, responsibility, friendships and being mentored feature often in their returning feedback debrief.

Table 16.1 A review of online graduate soft-skills (Indeed, Valamis, Resume Genius, Zety, LinkedIn and Making Business Matter)

| Soft skills attributes for graduates | | | |
|--------------------------------------|-------|----------|---------------|
| Soft skill | Count | Personal | Interpersonal |
| Accountability | 1 | 1 | |
| Active listening | 2 | | 2 |
| Adaptability | 2 | 2 | |
| Ambition | 1 | 1 | |
| Attention to detail | 1 | 1 | |
| Collaboration | 2 | | 2 |
| Communication | 7 | | 7 |
| Conflict resolution | 2 | | 2 |
| Creativity | 2 | 2 | |
| Critical thinking | 2 | 2 | |
| Emotional intelligence | 2 | 2 | |
| Empathy | 1 | | 1 |
| Entrepreneurial | 1 | 1 | |
| Flexibility | 1 | 1 | |
| Initiative | 1 | 1 | |
| Innovation | 1 | 1 | |
| Interpersonal skills | 1 | | 1 |
| Leadership | 6 | | 6 |
| Negotiation | 1 | | 1 |
| Optimism | 1 | 1 | |
| Organisation | 2 | 2 | |
| Patience | 1 | 1 | |
| People skills | 1 | | 1 |
| Positivity | 1 | 1 | |
| Problem solving | 4 | 4 | |
| Professional | 1 | | 1 |
| Public speaking | 1 | | 1 |
| Responsibility | 1 | | 1 |
| Self-awareness | 1 | 1 | |
| Strategic thinking | 1 | 1 | |
| Teamwork | 3 | | 3 |
| Time management | 2 | 2 | |
| Work ethic | 1 | 1 | |

16.6.3 Soft Skills—Interpersonal

Perhaps the surprise in Table 16.1 is not the categories of interpersonal soft-skill attributes (communication, leadership, teamwork, collaboration, conflict resolution) but the number of times these attributes are repeated across different online articles: Communication—7; Leadership—6; Teamwork—3). Figure 16.1 demonstrates this (based on Table 16.1).

At the University of Huddersfield, our computing-related courses tend to have teamworking modules across all courses at all years, so the opportunity to engage with a variety of peers over different tasks is great. However, we tend not to teach the principals of leadership or conflict resolution. Although students are well-versed in the use of technological applications to engage in communication and peer-assessment, they don’t always engage or engage appropriately/adequately. Indeed, peer assessment (scoring their fellow members on aspects such as teamwork, communications, quality of work, timeliness etc.) can be used by some to elevate or penalise certain team members.

Whilst we don’t judge students by the minutiae of their team working practices, they do learn what good and poor practices look like. However, in an academic environment, students learn how to do things... or they learn how to ‘play the game’ and so they don’t all necessarily know what true team working is about, when reputation, accountability and career prospects are in play.

Students work differently in the employment environment...because they must. In HE, their ambition is perhaps to get an A-grade, with a responsibility and accountability only to themselves. Whilst in the work environment, student output standards are management- and customer-related and responsibility and accountability are spread among fellow employees, management and the customer/client. This is a

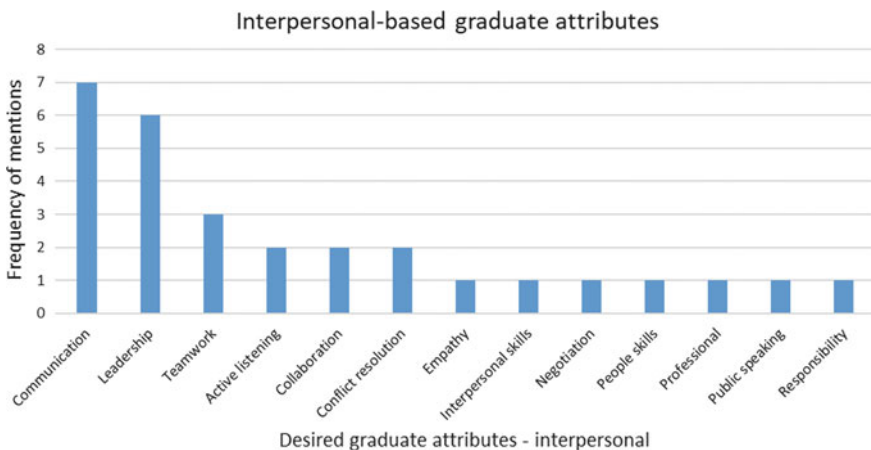


Fig. 16.1 Frequency of mentions of desirable graduate interpersonal attributes (derived from online articles, see Table 16.1)

significant step up in the quality and timing of the work effort. The consequences of poor performance in academia, is a lower mark; in the workplace it is unemployment.

16.6.4 GPA and the eight Huddersfield Graduate Attributes (HGA)

Much of the GPA work involves writing reflective reports on a series of topics related to eight key employability attributes. Known as the Huddersfield Graduate Attributes, they attempt to fill the employment skills gap between HE academia and the skills businesses would like to have in their new graduates. The HGA Employability and Enterprise Mapping matrix (**HGA: Develop your graduate attributes**) covers the following eight categories, some are personal soft-skills, some interpersonal and others relate to leadership:

- **Self-motivation**—explore the topic at hand, self-directed learning, use of management techniques, personal goals, strengths and weaknesses;
- **Commercially aware**—links between teaching and work, stakeholders in curriculum delivery, how teaching aligns with industry standards, work experience and live projects, self-promotion externally;
- **Enterprising**—creativity and innovation practice, inter-disciplinary learning, evaluate opportunities, work with internal and external stakeholders, personal development reflection, sell new ideas, awareness of new technology;
- **Resilient**—be constructive with failure, develop wellbeing and confidence, use feedback;
- **Effective collaborator**—group work and collaboration, engage with public and wider community, work positively with difference (personalities, views);
- **Confident leader**—selling ideas, opportunities to lead, managing group processes;
- **Globally and socially aware**—consider ethics, global, political and social aspects, critically situate self, international perspectives;
- **Personal development and growth**—define life goals and values, graduate opportunities, career plan and implementation.

The HGA scheme does seem to plug the gaps with respect to preparation for leadership, emotional resilience and career planning aspects, as expected from newly qualified graduates. The next Section therefore assesses feedback from returning placement students to determine if any of these desirable attributes are learnt/experienced before graduation.

16.7 Placement Year—Student Feedback

Our placement year at Huddersfield is 12 months of work experience and should not be confused with Internships which are generally viewed as short-term placements.

As part of the management of placements, we survey returning students before they start their final year of study using a debrief questionnaire. In this Section, we assess pertinent elements of just one year’s comments of returning students: 2017–2018, a pre-Covid-19 cohort ($n_{\text{students}} = 112$, with 11 not providing any comments, $n_{\text{comments}} = 101$). In the debrief questionnaire, students are asked for observations and reflections on their placement in the form of a free-form text box. They had limited time to complete this, so the spelling and grammar are not always correct. However, we have decided to leave the comments verbatim. Their reflections provide a rich and deep view of their year’s highlights: things that really stand out for them.

What follows is a shallow-dive into some of these comments, across Engineering (Electrical, Electronic and Mechanical), Music Technology, Computing (Computer Science, Information Systems, Games and Multimedia/Web). These students commenced placement in summer 2017 and finished around August 2018. We compare stated experiences with the desired industry graduate requirements as well as the additional HGA attributes; principally the wide range of soft-skills—personal and interpersonal. There were a few common themes expressed by the students, some of the prominent ones are captured below.

16.7.1 Inclusion—Friendly and Welcoming

Reading through the comments, it is apparent that there is as much to be gleaned about their initial reservations and fears towards the placement year from comments relating to how ‘friendly and welcoming’ their new employment peers were. For many of us reading this, familiar with the workplace, it might seem unusual, but it registers as quite an ‘eye opener’ for many students. Some 67 comments (31%) mentioned something positive about their fellow workers. Some were briefly mentioned, such as:

“I was apart of a multi-discipline team which gave me experience with a number of different engineers....”

Engineering student

“Learnt skills, worked with other people in a team....”

Computing Student

Many were more effusive in describing their former colleagues and the working environment:

“Very welcoming staff who offered lots of support throughout the year.”

Computing student

“...My boss was really supportive and my colleagues made coming to work fun.”

Music student

“It was a good year because the work place atmosphere was really good. Ability to participate in loads of non work related activities with colleagues. Provided the opportunities to work with a variety of different people from the top bosses to the Engineers on the shopfloor...”

Engineering student

Some comments were more reflective, outlining specific benefits of teamworking that they appreciated:

“... I got a great insight into working in a team for an external period of time and how the social skills that come with it, are as if not more important than the technical skills that are most obvious.”

Computing student

“Gained knowledge of what it is like working in a real life working environment. Gained independence and assertiveness was able to work for a professional working team.”

Computing student

“I developed not only sound engineering skills but also social skills dealing with bands, management and other engineers. I was always made to feel welcome and valued as a team member. My supervisor couldn’t have been better; he was patient, helpful and never got frustrated if I had to ask twice for help.”

Music student

16.7.2 Experience—Employment Versus Study

Students also became aware of the large shifts in personal experience and soft-skills that exposure to a working environment provide and how they often seamlessly pick up these skills along the way. Most students would avoid presentations and similar personal-intensive events in an academic setting, but they have little or no choice in industry to engage in such team and social settings. The difference in work focus, responsibility and accountability in the workplace are highlighted in these comments:

“Because I was pushed to new lengths and learnt a lot more at work than university.”

Engineering student

“I learnt more skills and techniques than I did in my first and second year...”

Computing student

“Because it was real, engaging work; More technical and social skills developed in my first week than at first two years of university.”

Computing student

16.7.3 Leadership Experience

By virtue of their role, managing a project is probably the only leadership-related activity many placement students would be able to engage in; none reported that they managed other employees. However, several comments (approximately 9%

of students) related directly or indirectly to leadership-related attributes during the placement year, such as: communication; accountability; resilience; vision; influence; positivity; confidence; problem-solving and decision-making. Some of the indirect comments were:

"Self initiated a project with the other interns and followed an agile fashion."

Computing student

"Given a chance to lead project which were inspected with a international inspectors."

Engineering student

Some were expressive of a greater degree of autonomy and leadership for their own projects:

"My placement was good because was given the responsibility of managing my own projects and improving process to help not only the team but the whole company."

Computing student

"Learnt a huge amount, worked in a fantastic team, provided me with lots of opportunities to work independantly on projects."

Computing student

16.7.4 Prestige, Responsibility and Personal Growth

It is clear that some students were given high levels of autonomy and responsibility to undertake work and also to interact with clients. Anecdotally, staff are aware that many students are typically given a two-to-three-month settling-in period and then given greater amounts of supervised/mentored work to do, as appropriate. Many are also moved through different departments to maximise work experience and/or to see where the students' passions and talents lie

Students often report that they were treated as regular employees; recognising the unspoken transition from student to entrusted employee. For the employer, this is a desirable change; for the student this realisation is often life-changing and career-affirming, resulting in feelings of loyalty, career-direction satisfaction and an easing of personal-development stress and anxiety. They can often be too busy 'growing their skills' to be anxious about new situations and experiences

"I was given the same role and responsibilities as permanent staff, and was given great opportunity to share my own input on company projects ..."

Music student

"I was also treated more like an employee than a placement student which was good. Did work that produced lasting effects which will still be in place after I leave."

Engineering student

"I was entrusted to be the backline tech, alone at [national UK] stadium."

Music student

16.7.5 *Enterprise Placement Year—Leadership*

A second (and less common) version of the Placement Year allows students to work for themselves, either as an individual or in a group/team and to follow a more enterprising and entrepreneurial path. We run the Enterprise Placement Year (EPY), popular with Games Design and Games Programming (both part of Computing) and Music/Music Technology students.

This is a non-academic year (i.e. no academic credits are available) and the focus is on business training, Business Plan development and the business-pitching aspects of work. Students are expected to engage with: branding; promotion and marketing; networking; product development; financial planning; funding applications where available; and ideally the sale of products.

Students are expected to experience leadership roles, either for themselves or the team, as well as in their individual spheres of operation. They are expected to market their creations and network to grow a willing band of followers and potential customers. Many of the EPY student comments are directly related to these leadership qualities with the added opportunities for interpersonal activities such as: empathy; delegation; sub-contract management.

“Learnt about how to run a business, gave me the opportunity to learn at my pace and the freedom to choose the areas of work to develop, had access to business advisors and an industry mentor. Could apply for funding to develop my business.”

Music student

The following student started a Games EPY in a team, which split up during the year. Since the student continued, we can infer a good level of emotional resilience, motivation and adaptability.

“Spent the time working on my own projects for self employment. Whilst also working on freelance projects for clients. However the team I started EPY with split up so I made a game by myself.”

Games student

Every placement position and every student are different, however, those companies that take placements from us each year tend to offer the same roles, managed by the same people/team, so there is great continuity on the employer’s side. Students differ greatly among themselves, but by studying the same courses and going through fairly demanding selection procedures, the same types of students attain a placement position, year after year. Feedback is overwhelmingly positive, often with no negative comments at all. Of course, the challenge is to ensure that every student is willing, motivated and capable of achieving a placement position, which isn’t the case. This will require more research in the future.

16.8 Conclusions

Part of the motivation for this work was to lay out, on the one hand, the barriers to students gaining a placement, and on the other, to begin to outline the skills and attributes that the placement experience brings to HE sandwich/ placement students. As we progressed the work, it became clear that many of the placement student feedback comments were expressing a high level of soft-skill employment attributes, with many gaining graduate-level attributes, as identified by the CMI and online careers/jobs websites. Without the placement year, the main way that students would graduate with attributes such as leadership, communication, responsibility and emotional resilience etc. would to have been, for example, Head Boy/Girl at a School or College, to have completed the Duke of Edinburgh's Award or to have had a high level of engagement via a personal/team/sport/group activity or via employment. It takes a certain type (or calibre) of person to want to engage with these types of advanced activities, so clearly, the route to a placement role starts with home role-models and/or schooling.

16.8.1 *Barriers to Placement Employment*

Of the many barriers to obtaining a placement (availability, locality, course-related, employer willingness and role requirements etc.), arguably the largest is the student's own aptitude, confidence, expectations and motivations.

Students can be their own biggest critic, potentially leading to: a lack of drive, desire or confidence in the process; not appreciating the value of company research; giving up on the extra effort an application requires; nervousness for the CV/interview process; an unease of fitting in to the working environment; and the lack of confidence and uncertainty of living and working during the placement year. Recent cohorts have not been helped with the UK leaving the European Union (2019 Brexit), the Covid-19 lock-down (starting March 2020), Russia's invasion of Ukraine (March 2022) and the current world-wide economic down-turn. The cost of living, including food, fuel, accommodation costs and mortgages, have risen considerably over this time and so the trend for many current students is to prefer to stay close to home. The unwillingness to travel is another potential barrier to a placement position.

16.8.2 *Work Experience and Employment Attributes*

The placement year is overwhelmingly a positive experience, judged by the wide range of positive comments and consistent lack of negative ones. Many students seem surprised that they are treated as normal employees and express a relish for the hard-skills learning, responsibility, contacts and satisfaction from being fully

immersed and engaged in normal working activities. They have improved on their social skills, have realised that they can start and finish complex tasks and for many, have found their ‘true calling’ (or their *passion*) within the sector.

For many who attain a placement, the promotion and support in Year 2 was very much ‘preaching to the converted’. There is a high degree of motivation for the placement year which seems to emanate from a combination of parental role models and/or one or more part-time jobs prior to or during their first two years of HE study. The more confident they are on starting a placement, the more experience and responsibility they are given, the more they enjoy the year. Several of the comments (not stated above) relate to meeting famous people and industry role models during the course of their work, as well as working with highly experienced professionals who are settled in their own careers and can advise and mentor the placement student. Add to this a typical salary for the year of £14–£17k (sometimes over £20k—2022 student cohort) and we get a good sense of the positives.

Many students grow long-term relationships with industry professionals during this year, who they can call on for advice in final year and beyond. They invariably leave with their manager willing to provide a glowing reference in a future job search. They learn considerable amounts of hard-skills within their job role/company structure. Whilst some students will undertake the same role for the full 12 months, many spend several months in different departments or locations, in part to see where their skills and interests lie, as well as to spread the benefits of their youth/cultural knowledge and their innovative creativity to different parts of the company.

It comes as no surprise to regular placement employers that students grow their soft-skills considerably during the year. For relatively naive students, their self-awareness and the amount of knowledge, experience and confidence from the daily work interactions is considerable over the year. They reflect on their new networking and communications skills, (inferred here in part as a reduction in shyness) and measured in the high esteem they hold for their new contacts. These broad aspects form part of *Communication* and are one of the highest set of graduate attributes desired by industry and management accreditation bodies.

Many students reflect positively on the trust imparted to them by their employer to undertake important tasks and projects. Some are allowed access to Department Heads, Heads of Service, fellow staff and customers to develop products and services. Whether leading on product/service development projects or on problem-solving exercises, the nature of having responsibility for the work, being accountable for the quality and timeliness of the results and being able to lead on decision-making are essential components of leadership; another highly desirable graduate employment attribute.

Our placement students, collectively, are clearly gaining recognised employment sector-based hard-skills as well as a wide variety of personal and interpersonal soft-skills. When cross-referenced with the: Chartered Management Institute Level 5 requirements; our University’s Huddersfield Graduate Attributes and online job-advice articles, it is apparent that a good proportion are attaining the desired graduate skills and attributes.

16.8.3 *Future Work*

The Placement Team deal with up to 200 students each year. We have over 500 individual comments available for analysis within a rolling five-year rolling General Data Protection Regulations (GDPR) window and plan a project of Machine Learning Text Analysis. This has been commenced to investigate aspects such as: sentiment; saliency; attribute relationships and others. This project is expected give a great insight to what our placement students actually achieve.

Whilst our School manages a large yearly proportion of placement students, there are many more who are either not interested or who don't engage, and we really don't know why. We want to draw out from them aspects of confidence, ambition, expectation etc. and to see how we can improve their outlooks through regular module content and specialised lectures/training/support. This will be done through a combination of personal interviews and questionnaires, reaching out to current students and those on staff LinkedIn profiles.

References

- Asana: The difference between hard skills and soft skills: examples from 14 Asana team members. <https://asana.com/resources/hard-skills-vs-soft-skills>. Accessed 27 Nov 2022
- Chartered Management Institute: Management and Leadership Qualifications (09/12/2022) <https://www.managers.org.uk/education-and-learning/qualifications/>
- Gradcracker—Where the STEM jobs are (09/12/2022). <https://www.gradcracker.com/>
- High Fliers Research Limited (2018) The graduate market in 2018: annual review of graduate vacancies & starting salaries at the UK's leading employers. https://www.highfliers.co.uk/download/2018/graduate_market/GMReport18.pdf. Accessed 27 Nov 2022
- Hora MT, Parrott E, Her P (2020) How do students conceptualise the college internship experience? Towards a student centred approach to designing and implementing internships. *J Educ Work* 5. <https://doi.org/10.1080/13639080.2019.1708869>
- Indeed (09/12/2022). <https://uk.indeed.com/>
- Indeed: hard skills versus soft skills: definition and examples. <https://uk.indeed.com/career-advice/career-development/hard-skills-vs-soft-skills>. Accessed 27 Nov 2022
- LinkedIn (09/12/2022). <https://www.linkedin.com>
- LinkedIn: Soft Skills between your success and failure. <https://www.linkedin.com/pulse/soft-skills-between-your-success-failure-blackbucks/>. Accessed 27 Nov 2022
- Making Business Matter: 8 soft skills and hard skills research conclusions. <https://www.makingbusinessmatter.co.uk/soft-skills-hard-skills/>. Accessed 27 Nov 2022
- Rate My Placement—Find paid placements and internships (09/12/2022). <https://www.ratemyplacement.co.uk/>
- Resume Genius: Hard skills vs soft skills: definitions and lists of examples. <https://resume-genius.com/blog/resume-help/hard-skills-vs-soft-skills>. Accessed 27 Nov 2022
- Rimmer F, Everson D, Crampton A (2019) Supporting students to be global graduates: opportunities and challenges in providing an international work-based learning experience. In: Carter J, Rosen C (eds) *Transnational higher education in computing courses*. Springer, Cham. https://doi.org/10.1007/978-3-030-28251-6_10
- The University of Huddersfield: Global Professional Award. <https://students.hud.ac.uk/studies/gpa/>. Accessed 9 Dec 2022

- The University of Huddersfield: Global Professional Award: develop your graduate attributes. <https://students.hud.ac.uk/studies/gpa/>. Accessed 9 Dec 2022
- Valamis: Hard Skills versus Soft Skills. <https://www.valamis.com/hub/hard-skills-vs-soft-skills>. Accessed 27 Nov 2022
- Whittard D, Drew H, Ritchie F (2022) Not just arms and legs: employer perspectives on student workers. *J Educ Work* 35(6–7):751–765. <https://doi.org/10.1080/13639080.2022.2126972>
- Zety: Hard skills vs soft skills: list of examples for your resume. <https://zety.com/blog/hard-skills-soft-skills>. Accessed 27 Nov 2022

Chapter 17

Baseline Skills—Scaffolding Soft Skills Development Within the Curriculum



Sue Beckingham

Abstract To enable Computer Science students to develop employability and 'work-ready' skills it is important to consider both the technical skills aligned to their discipline and the soft skills desired by employers. Research has identified that students in Computer Science would benefit from further support to develop the latter. This chapter considers how these skills can be developed through a variety of work experience opportunities including work-based learning and work-related learning; in class activities and alternative teaching approaches such as project, inquiry and problem-based learning; and through scaffolding both soft skills development and reflective practice, how students can become more confident in articulating these skills when applying for graduate work.

Keywords Employability · Soft skills · Problem based learning · Reflective practice

Over recent years there have been numerous accounts in the news proclaiming that graduates are not work ready as they don't possess the baseline soft skills required by employers. Despite the growth in Computer Science and IT-related degrees and their subsequent graduates, unemployment was running at just over 10% (Shadbolt 2015). As a result the 'Shadbolt Review of Computer Sciences Degree Accreditation and Graduate Employability' was commissioned, and the research published in 2016.

This chapter will consider the recommendations made within the Shadbolt Review and other research to highlight:

- what elusive soft skills employers are looking for;
- provide guidance and recommendations on how soft skills development can be integrated effectively within the curriculum; and
- how students can go on to apply and showcase these skills confidently and effectively through a professional online presence

S. Beckingham (✉)
Sheffield Hallam University, Sheffield, UK
e-mail: S.Beckingham@shu.ac.uk

17.1 Employability

Over two decades ago Dearing (1997: 133) recommended that higher education focus on key skills which were the ‘key to the future success of graduates whatever they intend to do in later life’ (p. 133). Skills identified included: communication skills, numeracy, information technology, learning how to learn/personal development planning, problem solving and team-working. Leckey and McGuigan (1997) lamented on the allegations of a gap between the generic skills fostered by higher education, and those that the labour market need. Whilst subject specific knowledge and skills are strong, the transferable knowledge, skills and attitudes essential for the world of work are weak. They go on to cite the European Commission (1991: 44) who argue “One feature of current skills shortage is the widespread lack of important generic skills and social skills such as quality assurance skills, problem-solving skills, learning efficiency, flexibility and communication skills”.

To understand what makes individuals employable Knight and Yorke (2004: 5) defined employability as “A set of achievements—skills, understandings and personal attributes - that make individuals more likely to gain employment and be successful in their chosen occupations, which benefits themselves, the workforce, the community and the economy”. Cole and Tibby (2013: 5) add that employability is about “supporting students to develop a range of knowledge, skills, behaviours, attributes and attitudes which will enable them to be successful not just in employment but in life”.

In 2016 the Shadbolt Review was commissioned by Ministers from the Department for Business, Innovation and Skills (BIS), where one of the key strategic priorities was to support and develop science and engineering talent coming through the education system and ensure that the UK has access to ‘the skills and knowledge that it needs to drive economic growth and the development of a more innovative, productive and information-driven economy.’ (Shadbolt 2016: 13). To realise this ambition, it is vital to align the skills graduates have with those required by employers. In order to do this and prepare work ready graduates, a recommendation for employers and higher education providers to work more closely together was made.

The Shadbolt Review (2016: 52) highlighted the following as the main issues impacting on graduate employability:

- Graduates lacking ‘softer skills’
- Graduates lacking specific knowledge
- Graduates lacking computer programming skills of specific programming languages
- Graduates lacking business/commercial awareness
- Graduates lacking work experience.

17.2 Skills

The terminology used to describe skills most frequently refers to soft and hard skills. Generally speaking hard skills tend to be those that have been or can be tested; may be recognised with a professional, technical or academic qualification; and are quantifiable. Soft skills are personal life skills and tend to be more subjective. However there are variations as can be seen in Fig. 17.1.

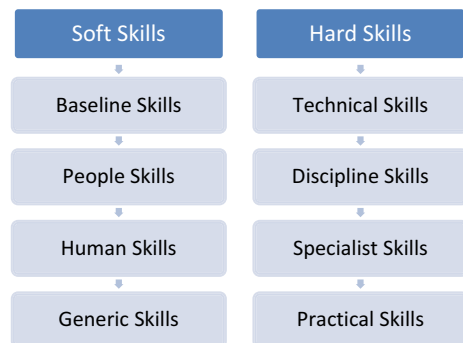
In 2004 the Pedagogy for Employability Group (p. 5) as cited by Dacre Poole and Sewell (2007) collated the following skills, advocating that these were the skills that employers expect graduates to have:

- imagination/creativity
- adaptability/flexibility
- willingness to learn
- independent working/autonomy
- working in a team
- ability to manage others
- ability to work under pressure
- good oral communication
- communication in writing for varied purposes/audiences
- numeracy
- attention to detail
- time management
- assumption of responsibility and for making decisions
- planning, coordinating and organising ability

A series of scoping interviews with employers carried out by Lowden et al. (2011: 12) recognised the need for skills and knowledge for specific roles (which would include technical skills); however all agreed the following *transferable* skills were considered most relevant:

- team working
- problem solving

Fig. 17.1 Alternative names for soft and hard skills



- self-management
- knowledge of the business
- literacy and numeracy relevant to the post
- ICT knowledge
- good interpersonal and communication skills
- ability to use initiative but also to follow instruction

The UK Commission for Employment and Skills (UKCES) a publicly funded, industry-led organisation provides leadership on skills and employment issues across the UK. UKCES carry out an Employer Skills Survey at regular intervals, the last conducted in 2015. Within this report it states that skills can be defined within two groups and named these: 'people and personal skills' and 'technical and practical skills' (UKCES 2015a: 44).

Hawkins (1999) splits soft skills into three groups: people skills, self-reliance skills, and generalist skills; then includes a fourth group as technical skills. People skills include team working, leadership, interpersonal skills, and customer orientation. Self-reliance skills include self-awareness, confidence, self-promotion, initiative, proactivity, networking, willing to learn and action planning. Generalist skills include problem solving, IT/computer literacy, flexibility, numeracy, business acumen and commitment (Hawkins 1999: 12).

Kaplan (Pedley-Smith 2014) published a white paper on graduate recruitment, learning and development in which a survey compiled a list of competencies. These were grouped as Knowledge (e.g. numeracy, literacy, technical knowledge), Skills (e.g. effective communication, analytical, problem solving) and Attitude (e.g. team player, confidence, positive mental attitude). Thurner et al. (2012) consulted software companies (potential employers of graduates) to identify desired skills and these are grouped as Self-Competencies (e.g., openness to constructive criticism, striving for life-long learning), Practical and Cognitive Competencies (e.g., analytical thinking, diligent and accurate work style) and Social Competencies (e.g., empathy and understanding of others, ability to work and cooperate in a team).

The World Economic Forum (WEF) report (2016: 21) categorises core work-related skills as abilities, basic skills, and cross-functional skills. Within these groupings there are subgroups:

- Abilities: cognitive abilities and physical abilities
- Basic Skills: content skills and process skills
- Cross Functional Skills: social skills, systems skills, complex problem-solving skills, resource management skills and technical skills

Technical skills that involve coding, AI, data science and cyber security are in great demand due to the massive digital acceleration that is pervading our person and working lives (Marr 2022). What is quite clear is that not only students but also graduates and professionals in the IT industry will need to continue to engage in lifelong learning to keep up with new and emerging technologies. This period is often referred to as the fourth industrial revolution or Industry 4.0.

An area of concern is the ability for students and graduates to recognise and evidence vital soft skills. In a recent publication Cater (2021) refers to these as professional skills. Being able to articulate transferable skills such as problem solving with clear examples requires the students to catalogue a collection of examples and to develop the confidence to sell themselves (Shadbolt 2016). One of the issues that students and graduates appear to struggle with is being able to provide clear examples of when and how a given skill has been applied in an authentic situation. However given the complexity of how skills are referred to, it should not come as a surprise that students and educators trying are struggling to know what skills they should focus on and the language to use to describe them.

17.3 Skills Gaps

A crucial problem faced is gaining a clear understanding about *what* key skills employers are looking for. The Council for Industry and Higher Education (CIHE) (2010) highlights the importance of individuals possessing a ‘fusion’ of technology, business, creative and interpersonal skills. The Association for Computing Machinery (ACM) states that each computing discipline must “articulate its own identity, recognize the identities of the other disciplines, and contribute to the shared identity of computing” (Association for Computing Machinery 2005: 8). In addition to having foundational knowledge in Computer Science, the Burgess Report (2016: 70) also highlights the significance of ‘soft’ or ‘work readiness’ skills. CBI’s report (2017) refers to work-related attitudes, with communication, teamworking and a positive attitude to work being critical to attaining career opening.

It is therefore important for students to not only build and develop soft skills, but to be applying these skills and knowledge gained in a professional and business context. The Office of Students (2018) an independent regulator of higher education in England states within its strategy that as an outcome of the higher education experience, students will be able to progress into employment or further study. Universities UK (2016: 30) argue that “while a degree may be a baseline requirement for attaining a job, an applicant’s ability to demonstrate skills and competencies that they can bring to the workplace will allow them to stand out from their graduate counterparts”.

The UKCES Employer Skills Survey (2015a: 45) acknowledging there is a need to understand what skills are in poor supply, looks to identify skills lacking in the labour market. Employers taking part in the survey selected from a list of skill descriptors presented within the two aforementioned groups:

1. people and personal skills
2. technical and practical skills

To some extent these groupings could be seen to complicate matters. For example making speeches or presentations are under people and personal skills and yet basic numerical skills and writing instructions, reports etc. are placed within technical and practical skills. There is an overlap here if communication skills were used to

describe a typical soft skill. That aside it still provides an indication of skills gaps and areas to focus on.

The 2017 Tenth CBI Education and Skills Survey (run in partnership with Pearson) received responses from 340 UK organisations. Within this research a similar set of skills were identified. Areas of particular weakness included international cultural awareness (39%), business and customer awareness (40%), and attitudes/behaviours e.g., resilience and self-management (32%) (CBI 2017: 93). Furthermore being able to demonstrate personal qualities such as resilience, attitude and confidence is inextricably linked to soft skills, and therefore also of importance (Fincher and Finlay 2016).

Looking to the future the anticipated skills needs are even harder to predict. On average, by 2020, more than a third of the desired core skill sets of most occupations will be comprised of skills that are not yet considered crucial to the job today (World Economic Forum 2016: 20). Jung (2022) highlights the disruption of the world of work by the ever-increasing use of new technologies, globalisation and hybrid working. This presents new roles and opportunities in industries that may previously have not been considered by students. The World Economic Forum 2020 report stated:

“In the mid-term, job destruction will most likely be offset by job growth in the ‘jobs of tomorrow’—the surging demand for workers who can fill green economy jobs, roles at the forefront of the data and AI economy, as well as new roles in engineering, cloud computing and product development.” (World Economic Forum 2020: 8).

In the Council of Professors and Heads of Computing (CPHC/HEA 2015) report on Computing Graduate Employability, it refers to an employer invited to speak to students about expectations and is cited as stating that those students graduating with a first or 2:1 demonstrate the aptitude to learn so any additional technical skills required can be taught. However what they cannot teach is the soft skills.

The employers consulted during the Shadbolt Review indicated that whilst a range of soft skills were desired, the top two were **communication** and **project management skills**, stating that these were “crucial for working in teams, developing successful working relationships and contributing positively to an employer’s strategic vision” (Shadbolt 2016: 55). Interestingly similar to the findings two decades prior by Leckey and McGuigan (1997: 368) who refer to the importance of ‘personal transferable skills’ and categorise these as communication skills, problem analysis and solving, interactional skills, initiative and efficiency.

Overall there is evidence that students benefit from authentic work experiences, ideally in an organisation that provides them with the opportunities to develop these soft skills and of particular importance communication as this is a fundamental overarching skill that will enable students to articulate the range of skills and experience, they have in subsequent job interviews. Being able to communicate is clearly a vital skill, but to do so confidently needs development, practice and ongoing feedback.

17.4 Work Experience

One of the key recommendations from the Shadbolt Report (2016) is to extend and promote work experience. Placements and internships can provide rich opportunities to apply knowledge and current skills, as well as developing new skills (Carter 2021). Such authentic work experience can provide a context for learning (Pegg et al. 2012). The report also suggests that the National Centre for Universities and Business (NCUB), the Council for Professors and Heads of Computing (CPHC), and the National Union of Students (NUS) should work closely to try to identify what barriers Computer Sciences students are facing when trying to gain work experience.

There is a correlation between those that have undertaken a placement (or other work experience) and being employed once graduated. It is therefore very important to prepare and encourage students to ensure they are in the best possible position to apply for posts. However ‘work awareness’ is also valued. This is where employers have an expectation that new employees will have “a useful awareness of the world of work... a feel for the market, and what’s going on in the world” (Bennet et al. 2000: 101). Other terms used are experiential learning where Roberts (2015) includes active learning, integrated learning, project-based learning and community-based learning. These include opportunities where workplace engagement includes *experiences* where students can apply learning outside the classroom (Roberts 2018).

Work experience can be referred to as work-based learning (in the workplace) or work-related learning (workplace and learning space, and also simulated space). Work-based learning (WBL) is the term used to describe a class of university programmes that brings together universities and work organisations to create new learning opportunities in workplaces (Strachan et al. 2011: 134). The Department for Children, Schools and Families (DCSF 2009) define work-related learning as: “Planned activity that uses the context of work to develop knowledge, skills and understanding useful in work, including learning through the experience of work, learning about work and working practices, and learning the skills for work”. This is further described as:

- learning *for* work by developing skills for enterprise and employability (e.g., problem-solving activities, work simulations and mock interviews)
- learning *about* work by providing opportunities to develop knowledge and understanding of work and enterprise (e.g., careers education)
- learning *through* work by providing opportunities for young people to learn from direct experiences of work (e.g., work experience or enterprise activities)

Work experience can be both paid and unpaid. Opportunities to work within an organisation closely related to the subject discipline can provide access to develop the broadest set of skills. However much can be gained through engaging with part-time work, volunteering and extra-curricular activities as important transferable soft skills can be developed and evidenced. Wilson (2012: 37) stated that in addition to the course/programme studied, personal skills are developed as “a consequence of social and family background, the environment within which study is undertaken

Table 17.1 Skills development opportunities

| Examples of where skills can be developed | |
|---|---|
| Sandwich placement | Year-long industry placements typically in the third or final year |
| Semester placement | Sort internships could be as little as a day a week or in a block |
| Work shadowing | An opportunity to get a feel for the working environment |
| Summer internships | Taken over the summer break and include working abroad |
| Part-time work | Typically unrelated to subject discipline |
| Course and university initiatives | Hackathons and competitions, student led conferences, students as researchers' projects |
| Volunteering | Charities, computer clubs, peer assisted learning schemes, course rep and other university committees |
| Extra-curricular activities | Clubs, societies, special interest groups |

and the extracurricular activities of the student". This might include voluntary work, community work, or part-time work unrelated to the degree subject (Table 17.1).

Cross-discipline work-related learning opportunities can provide two-way learning. For example an institution led initiative provided an IT service desk to nursing students. The Nursing students gained IT skills, and the Computing students were given feedback on their communication skills. Peer assisted learning (PAL) schemes help to develop leadership, communication and mentoring skills, whereby students mentor students in the year(s) below them.

17.5 Developing Opportunities for Work Experience

The development of an Industrial Advisory Board linking Computer Science academics with industry professional can open up a forum to discuss both work experience prospects and the preparation needed to apply for such posts; as well as forging links for graduate jobs (Universities UK and UKCES 2014, UKCES 2015b). Partnerships with businesses can also provide opportunities for guest lectures, industry visits and projects. Students on placement often go on to focus their final year project or dissertation on a topic related to the organisation they worked at. Employer engagement can include inviting employers to provide information, advice and guidance; or to contribute case studies or work-based scenarios to the curriculum (University Alliance 2015).

Cole and Tibby's (2013: 10) paper 'Defining and developing your approach to employability' is a useful planning tool and comes with an action plan for course teams. It considers four stages:

1. Discussion and reflection—creating and defining a shared point of reference
2. Review/mapping—what are we doing/not doing?

3. Action—how do we share and enhance existing practice? How do we address gaps in provision?
4. Evaluate—What does success look like and how is it measured? How can we enhance practice further?

There is much to be gained through academic cross-institutional visits, developing communities of practice between institutions to discuss and share good practice about employability and skills development. A CPHC-funded (Council of Professors and Heads of Computing) initiative focussed on employability took place during 2016/17 called ‘GECCO Building a Graduate Employability Community in Computing’. A series of three events took place in London, Manchester and Edinburgh. The evaluation report (CPHC 2017) shared that participants valued:

- Time and space share/discuss practice
- Discussions between practitioners for ideas and reflection
- Networking (mentioned three times)
- Exchanging ideas/establishing new contacts.
- Contacts. Ratification that we are not alone!!
- New contacts at other institutions.
- The networking opportunities
- Talking to colleagues about employability across the sector

Maintaining alumni networks is key, as when students graduate and maintain connections with their alma mater, they are more likely to let past Tutors know about job offerings and placements opportunities in their workplace. Equally it enables the academics to stay in touch and invite graduates back to give talks to inspire students to take a placement year, undertake extracurricular activities and explain how these develop the skills employers are seeking. Utilising LinkedIn, (regarded as ‘professional’ networking) to create course alumni groups, provides a useful way to keep in touch with graduates and follow their progression. Encouraging students to also join industry groups and follow Company pages can give an insight into the culture and specialisms within organisations.

17.6 Approaches to Teaching Soft Skills

Teaching soft skills can be challenging. Some computing programmes have specific stand-alone soft skills modules which focus solely on professionalism and communication. Whilst often aligned to the skills identified by employers as important, students do not always value the module. For some it is seen as a stand-alone bolted on addition to the course, containing activities the students feel they are already competent in. Criticism of this approach suggests that activities are not sufficiently contextualised in the computing specialism students are taking.

Knight and Yorke (2004: 199) note the following ways to include employability skills development in the curricula:

- employability through the whole curriculum
- employability in the core curriculum
- work-based learning or work-related learning incorporated as one or more components within the curriculum
- employability-related module(s) within the curriculum
- work-based or work-related learning in parallel with the curriculum

The British Computer Society's (BCS) best practice model considers the legal, social, ethical and professional issues in computing (LSEPI). Healey (2014) argues that providing opportunities to work through ethical issues can help students to develop critical thinking skills. Connecting current examples of these issues that align with specific computing specialisms are more likely to engage students than providing generic examples. An inquiry-based learning approach can provide students to research exemplars and then link the activity with an exercise to identify the skills they have gained as a result. There is further scope for BCS to collate Computer Science case studies of best practice implementing effective approaches to embed LSEPI in the curriculum; and to liaise with Association of Graduate Recruiters (AGR) and the Association of Graduate Careers Advisory Services (AGCAS) to develop a model for accrediting careers advice provision within the curriculum (Shadbolt 2016).

The capstone or service-learning module is an alternative approach whereby students engage in project based learning and key skills are purposefully integrated. For example, written and verbal communication, teamwork and organisation skills (Carter 2011). This could involve real data and role play where students play the role of the Client and set the scene for the project (Vogler et al. 2017). Schonell and Macklin (2019) stress the importance of creating authentic learning activities as work integrated learning using live cases. This is where real businesses are involved and provide a valuable work-related learning experience. The application of communication skills is demonstrated through meetings with the Client face to face, online, by phone, via email and through presentations and a final report; along with a vast array of other skills needed to undertake and complete the project. When working with a real business, students will experience client management, meeting client expectations and potentially conflict management (Gonzalez-Morales et al. 2011). Jackson (2014) refers to work-integrated learning and opportunities for service learning where students apply their professional skills through participating in an authentic activity that benefits the community. There are also opportunities to cross disciplinary boundaries, involving students from other courses. Hazzan and Har-Shai (2014) ask students to reflect on one stage in the lifecycle of a company, stage of a project or specific department, and describe the skills required. Yu and Adaikkalavan (2016) expound upon the value of problem-based learning where time is given to coach students and evaluate performance.

Another approach is involving students in the planning of the curriculum in relation to soft skills development. This can be done by asking the students what skills they would like to develop on the course and to suggest or contribute to the design of activities to develop these. Using a card sort exercise, students can engage in an activity to rank skills. For example:

- group activity—identify the ones they feel employers most value
- individual activity—identify the skills they are most/least confident in
- group activity—design activities to develop skills
- Individual activity—articulate what soft skills are and why they are of value

Innovative approaches take a constructionist approach (Papert and Harel 1991) and use Lego Serious Play to engage students in discussions around topics such as barriers and enablers of effective teams, and skills development. Students use the bricks to build metaphorical representations and share stories based on these. It provides an effective way for students to express themselves in a non-threatening way and to learn from each other (Peabody and Noyes 2017).

Scott et al. (2019) introduced a specific employability module which runs in the first few weeks of the final undergraduate year. This included the use of case studies to review jobs for suitability and skill alignment, the practice of assessment centres and psychometric tests, and workshops on CV writing and preparing for interviews. Career development learning (CDL) should be student centred, inclusive and context specific relating to the discipline and associated careers students may pursue (Dollinger et al. 2022).

Key to all of these activities is helping students identify and then articulate the skills they are developing (as well as recognise those that need further development). Often what can be missing is a portfolio that is owned and valued by the student to capture this information.

17.7 Personal Development Planning (PDP)

The integration of personal development planning (PDP) can be an effective way to engage in reflective practice. PDP is defined by the QAA (2009: 2) as a "structured and supported process undertaken by a learner to reflect upon their own learning, performance and/or achievement and to plan for their personal, educational and career development." Furthermore the QAA states PDP supports the idea that learning is a lifelong and life-wide activity. To achieve skills enhancement requires self-awareness of strengths and weaknesses from personal skills profiles (Wilson 2012).

The 2004 Burgess report 'Measuring and Recording Student Achievement' concluded that whilst advocating the use of personal development planning, further work should be supported to ensure research extended knowledge of the most effective strategies and evaluation of impact. This was followed in 2007 by the Burgess Group Final Report 'Beyond the honours degree classification' which introduced the Higher Education Achievement Report (HEAR). The HEAR (2008) is described as being "designed to encourage a more sophisticated approach to recording student achievement, which acknowledges fully the range of opportunities that higher education institutions in the UK offer to their students".

The recommendation was that students evidence additional skills attained at university which is intended to provide employers with an additional report to

their transcript of academic results, providing a more comprehensive record of achievement. For example the HEAR (2008) can capture:

- additional awards—accredited performance in non-academic contexts and individual units/modules studies in addition to the main degree programme
- additional recognised activities—roles and activities undertaken by students which demonstrate achievement but for which no recognition is given in terms of academic credit e.g. volunteering, student union representative roles, representation at national level sport or training course run internally
- university, professional and departmental prizes

Another way of capturing attainment is through 'Progress Files'. These can provide a way for students to record, reflect and review the skills and experience gained from the curriculum, work related learning, work-based learning as well as casual and voluntary work. In doing so they will develop the confidence to articulate and evidence their skills and knowledge (QAA 2001). In Pegg et al. (2012: 27) a case study by Waldock advocates the use of weekly e-Progress files throughout their time at university, whereby students develop a culture of engagement, value supported learning through regular feedback, and provides staff with regular ongoing feedback on their teaching.

PDP is seen both as a set of process and through a portfolio a valuable product in its own right. Within the portfolio, students are expected to review achievements, identify learning needs, plan how to address these needs, and present achievements (Knight and York 2004). However for this to be achieved it is important that guidance and support is given to students (Beard 2018). It should also be acknowledged that reflection takes practice and encouragement. To move from surface reflection and acceptant thinking to deep reflection and question thinking, students need to learn how to effectively undertake self-analysis and achieve self-awareness. They need to value the process and the outcomes (Carter 2011). When amotivated, there is a risk that they will just go through the motions. Fung (2017) advocates the integration of research and enquiry-based pedagogies, where activities and skills develop over time.

Dacre-Pool and Sewell (2007: 280) created the 'The Key to Employability' a metaphorical model. The CareerEDGE Employability Development Plan aims to develop increased levels of self-efficacy, self-confidence and self-esteem through reflection and evaluation. CareerEDGE is a mnemonic for the five components of the model.

1. **Career** Development Learning
2. **Experience** (work and life)
3. **Degree** Subject Knowledge, Understanding and Skills
4. **Generic** Skills
5. **Emotional** intelligence

The Centre for Recording Achievement (nd) is a national network organisation and a registered charity which seeks to “promote the awareness of recording achievement and action planning process as an important element to improving learning and progression throughout the world of education training and employment”. It provides a useful collection of case studies and CPD opportunities.

17.8 Scaffolding Reflective Practice

Learning how to reflect on experiences and developing a habit of doing so is an important life skill. It can have a profound impact on learning. Boud et al. (1985) consider the journey of turning experience into learning in three stages: What? (experience), So what? (reflection) and Now what? (learning). Developing reflective skills can in itself help individuals with the art of knowing how to learn (Helyer 2015).

The concept and pedagogy supporting the use of PDP has been written about extensively. As practioners we know ourselves it is valuable as part of our own CPD. However it is clear that reflecting effectively for many does not come easily. One of the barriers is finding the language to articulate what needs to be said. ‘Blank canvas syndrome’ is rife, with students declaring “I don’t know what to write!”.

There is a tendency to simply write about the ‘what I did’ but not expand upon what might be done differently, what was learned, what areas of development are needed and a recognition of skills that have been developed. It is therefore helpful to scaffold the process of reflection. This can be done by first of all explaining what reflection is and providing examples of how reflective practice is used outside of academia.

For example reflection is common practice in sport, where after a game the players will review their performance. Reflective practice in sport appraises what was on form and what can be learned from mistakes made. In the military After-Action Reviews (AARs) are conducted during or immediately after each event. They use open ended questions, determine strengths and weaknesses, and link performance to training. These consider: what did we set out to do, what actually happened, why did it happen and what will we do next time.

Then in the context of computing the use of Scrum has been used since the 1990s. Schwaber and Sutherland (2017) define Scrum as “a framework within which people can address complex adaptive problems, while productively and creatively delivering products of the highest possible value”. Scrum prescribes four formal events for inspection and adaptation:

1. **Sprint Planning**—The work to be performed in the Sprint is planned at the Sprint Planning. This plan is created by the collaborative work of the entire Scrum Team
2. **Daily Scrum**—This is a 15-min time-boxed event for the Development Team and is held every day of the Sprint. At it, the Development Team plans work for the next 24 h.

3. **Sprint Review**—This is an informal meeting, not a status meeting, and the presentation of the Increment is intended to elicit feedback and foster collaboration.
4. **Sprint Retrospective**—This is an opportunity for the Scrum Team to inspect itself and create a plan for improvements to be enacted during the next Sprint.

Whilst the language used is different the concept of teams 'inspecting itself', can be aligned with reflection. It can be valuable to invite professionals in to describe how Scrum is used in the context of their own organisation.

When it comes to capturing reflective practice, rather than asking students to go away and write 200–300 reflective words, providing aide-mémoires to guide students can be a very helpful and supportive approach. Building on the work of Gibb's (1988) reflective cycle which commences with a description of what happened and then considers feelings, evaluation, analysis, a conclusion and an action plan; it can be helpful provide suggestive prompts for each of the six stages of reflection. These can be presented in the form of a check list to encourage students to reflect beyond *the doing* and reach a point where they can start to identify skills they are developing and those that need further attention. Furthermore they can contextualise them in authentic scenarios, providing examples of how they have been able to apply the skills.

1. **Description: What happened?**

Begin by describing in detail the activity you are going to reflect upon. Think about including who you were with, where you were, what did you do/read/see; what were your responsibilities and what did you contribute, what others contributed; what were the outcomes.

2. **Feelings: What were you thinking and feeling?**

Now consider what you were thinking about. Capture what you were feeling at the beginning of the activity. What did you feel when you completed and how do you feel now? Did your feelings change? Consider how others made you feel.

3. **Evaluation: What was good and bad about the experience?**

The next step is to evaluate your experience. Think about what went well and what didn't go as well as expected. Record both the positive and negative aspects of each stage of the experience. Were there any difficulties? What/who was helpful/unhelpful?

4. **Analysis: What sense can you make out of the situation?**

Look more closely at why you think aspects went well or didn't go so well. What contributed to things going well? Where things went badly think about how this might have happened. Think about your contribution and how others contributed. How does it compare to other experiences?

5. **Conclusion: What else could you have done?**

Now you need to consider what you have learned from your experience. It is important to be honest with yourself and think about how you could have done anything in a different way. Have you learned anything from other people's approaches or behaviours?

6. **Action plan: If it rose again what would you do?**

Finally if you were to find yourself doing this or a similar activity again, how would you approach it? What would you do differently? Think about the skills you may need to develop to ensure it went better next time. Plan how and when you will undertake any skills development.

A further consideration is encouraging the students to explore a variety of approaches to capture their reflections. For example multimedia reflective blogs can contain text, but also photos, audio and video, sketch notes and mind maps. For example students learning in groups to programme a Lego Mindstorm in preparation for a race, could capture maths calculations as a photo, and a video of the robot in action. The visuals can aid recall and provide a focal point for reflection on the activity.

17.9 Scaffolding the Articulation of Soft Skills

It is important that student value the broad range of opportunities experienced whilst at university (Knight and Yorke 2004) and learn how to reflect confidently on the skills that they are developing; become able to articulate these; and also realise where their skills development is weak and understand how to develop an action plan to overcome any weaknesses.

As with reflective writing it can be challenging finding the right words to demonstrate skills. Hawkins (1999: 12) presents a ‘Skills Portfolio’ which captures a useful collection of adjectives for each of the skills listed. Providing students with such a list can act as a prompt as they identify and build their own portfolio of skills. For example team working might include supportive, facilitator, organised, co-ordinator, deliverer, imaginative, delegator, open-minded or willing to learn: motivated, adaptable, enthusiastic, active, keen learner, inquisitive, continual improver (Hawkins 1999: 12).

This could also be developed into a group activity where students in groups are tasked to identify relevant adjectives to describe a skill and the class shares and critiques the outputs. These can then be compared to Hawkin’s Skills Portfolio.

In Australia Ellis and Han (2020: 521–522) state the Quality Indicators for Learning and Teaching (QILT) look to measure collaboration as a valued skill on graduation through a questionnaire. This looks at how their course prepared them for work asking specifically the following:

- Working well in a team
- Getting on well with others in the workplace
- Working collaboratively with colleagues to complete tasks
- Understanding of different points of view

- Ability to interact with co-workers from different or multicultural backgrounds

These questions could easily be adapted and used for student reflection on their experience of groupwork throughout their time at university.

17.10 Peer Support

Students can learn about skills development and the value from their peers through peer assisted learning and mentoring schemes. Students as mentors develop leadership, communication and mentoring skills and students as mentees can develop a range of skills influenced by the activities, they are able to engage with (Pegg et al. 2012).

Another useful approach is either inviting students currently on placement, final years returning from placement or graduates who have experienced placements, to be involved in group or individual speed dating conversations or poster presentations to highlight the skills required for placements and the skills developed whilst out on placement.

17.11 Careers Support

Many universities have a Careers unit within the university. Those working in this area can provide students with one-to-one personal support or as group activities to further help them prepare for placements and graduate jobs. For example:

- Job applications
 - Skills
 - Competencies
 - Attributes
- CVs and cover letters
 - Skills and STAR statements
 - Technical skills
- Interviews
 - Phone interview
 - Skype video interview
 - Face to Face interview
- Assessment Centres
 - Group exercises
 - Role play

- Logic tests
- Tests
 - Psychometric testing
 - Numerical testing
 - In tray test (task organisation)

By engaging in these mock activities, students can practice articulating the skills they have and demonstrate these to receive feedback. This can be further enhanced where there is support through mentoring schemes, Academic Advisors, Personal Tutors, Employability Advisors and PAL (peer assisted learning) schemes.

Other class activities can support the development of presentation skills. This might include the use of PowerPoint (or similar), infographic posters or interactive digital poster, or screencasts.

17.12 Recording Skills

A useful approach to describe any skill is to develop a S.T.A.R. statement for each, whereby a scenario considers each of the four elements: situation, task, action, and result. Using the template in Fig. 17.2 a specific skill is chosen and then evidenced by considering an example of where it was applied. This allows for reflection on examples where things went well and also where they didn't. By considering the result there is an opportunity to further reflect on what was learned from a situation and what might be done differently next time. An interview question for example might be 'Describe a situation where you were working in a team and things didn't go as planned.'

Hawkins (1999: 14–15) developed an exercise to record tasks undertaken and skills gained in any type of work experience. The examples given include full-time work, voluntary work, part-time work and community work. The objective is to list all relevant tasks undertaken in the selected role and then for each task consider what skills were developed. To further aid the process skills are to be considered under the following four headings: working with data and information, working with people, working with practical things, or working with ideas. The exercise can be repeated for each role an individual has had. The outcome demonstrating the broad range of skills an individual has when looking at this from a wider perspective.

17.13 Showcasing Skills

In addition to the traditional CV, students applying for placements or graduate jobs can benefit greatly from having a professional online presence to showcase achievements and demonstrate their skills. This can be done by creating a profile on LinkedIn,

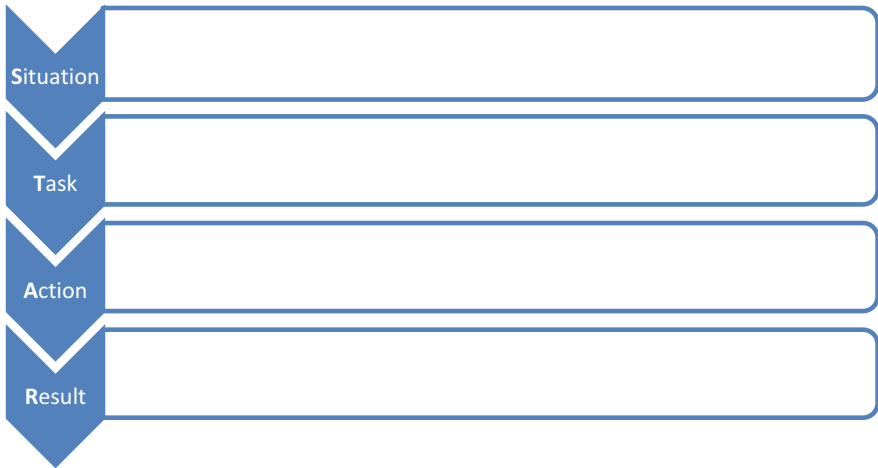


Fig. 17.2 Template for a STAR statement

which is both a professional networking site and a tool for recruiters to find talent. In addition the use of blogs, wikis, websites and e-portfolios can provide spaces to share skills, achievements and work experience (See Fig. 17.3). Engagement will increase if students can see the purpose (Barrett 2005) of digital portfolios. Clear signposts can help prospective employers find further information. This can easily be done by cross-referencing the online spaces using hyperlinks.

Students need to take ownership of these spaces and value what is presented publicly. It is common practice for employers and recruitment consultants to use search engines like Google to screen applicants. Digital footprints left in social media spaces where open to the public can reflect on the individual positively, but also negatively. Taking ownership of a personal profile is therefore important. The act of 'googling' yourself can be quite revealing. Not only does it help users see what others might see, it can provide a timely prompt to revisit social media profiles to

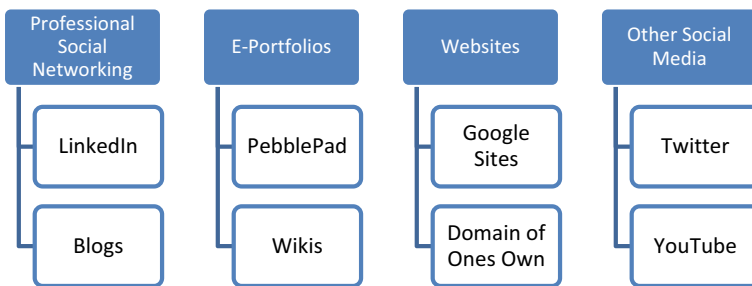


Fig. 17.3 Examples of tools to capture reflect on and showcase learning

check privacy settings where information is not intended to be openly public, and also to tidy up profiles that are public.

Hundreds of millions of professionals now have a LinkedIn profile to which they add their education, skills and information about their employment. As mentioned, employers will actively search for potential candidates using LinkedIn. Considering what search terms they might use to do this is another useful activity. These keywords might include placement or graduate. Where these words are included in a profile header can increase the chance of being found. For example: '*BSC Computing Science graduates looking for employment*'. Adding skills to a profile can help in this search process as well as providing the reader with a clear picture of the strengths of the individual.

Self-promoting a professional online presence can be done in a number of ways.

- email signature—adding the URL of a LinkedIn page, blog or website
- business cards—as above and converting the link into a QR code
- paper CVs—it is now common to see a LinkedIn URL included at the top of a CV alongside an email address and telephone number
- social media—links to any digital portfolios can be shared via Twitter and LinkedIn

17.14 Summary

The range of soft skills can be seen as a minefield. However building partnerships with industry can help to highlight what is required as work-ready skills, as this will differ in different specialisms within the sector and is likely to change over time in relation to what is a current priority. It is crucial that graduate skills are aligned with the expectations and needs of employers. Underpinning the wide spectrum of skills is the ability to confidently reflect and communicate examples when faced with a job application and interview. The development of all skills is enhanced greatly when students engage in work experience.

References

- Association for Computing Machinery (2005) Computing curricula 2005: the overview report. Available at: <https://www.acm.org/binaries/content/assets/education/curricula-recommendations/cc2005-march06final.pdf>
- Barrett HC (2005) Researching electronic portfolios and learning engagement. Available at: <http://www.electronicportfolios.org/reflect/whitepaper.pdf>
- Beard C (2018) Dewey in the world of experiential education. *New Dir Adult Continuing Educ* 158:27–37. <https://doi.org/10.1002/ace.20276>
- Bennett N, Dunne E, Carre C (2000) Skills development in higher education and employment. SRHE/Open University Press, Buckingham

- Boud DE, Keough RE, Walker DE (1985) *Reflection: turning experiences into learning*. Kogan Page, London
- Burgess R (2004) *Measuring and recording student achievement: Report of the Scoping Group*. Available at: <https://www.universitiesuk.ac.uk/policy-and-analysis/reports/Documents/2005/measuring-and-recording-student-achievement.pdf>
- Burgess R (2007) *Beyond the honours degree classification: Burgess Group Final Report*. Available at: http://www.hear.ac.uk/sites/default/files/Burgess_final2007.pdf
- Carter L (2011) Ideas for adding soft skills education to service learning and capstone courses for computer science students. In: SIGCSE '11 Proceedings of the 42nd ACM technical symposium on computer science education, pp 517–522. <https://dl.acm.org/citation.cfm?doid=1953163.1953312>
- Carter J (2021) *Work placements, internships and applied social research*. Sage
- CBI (2017) *Helping the UK thrive: CBI/Pearson employment and skills survey 2017*. <http://www.cbi.org.uk/insight-and-analysis/helping-the-uk-thrive/>
- Cole D, Tibby M (2013) Defining and developing your approach to employability: a framework for higher education institutions. Available at: <https://www.heacademy.ac.uk/knowledge-hub/defining-and-developing-your-approach-employability-framework-higher-education>
- CPHC (2017) *GECCO evaluation report: building a graduate employability community in computing*. <https://cphcuk.files.wordpress.com/2017/08/geccoevaluationreport.pdf>
- CPHC/HEA (2015) *Computing graduate employability: sharing practice*. Available at: <https://cphc.ac.uk/2016/01/08/cphchea-report-computing-graduate-employability-sharing-practice-published/>
- CRA (n.d.) *Centre for Recording Achievement*. Available at: <https://www.recordingachievement.ac.uk>
- Dacre-Pool L, Sewell P (2007) The key to employability: developing a practical model of graduate employability. *Educ Training* 49(4):277–289. <https://doi.org/10.1108/00400910710754435>
- DCSF (2009) *The work-related learning guide (second edition): a guidance document for employers, schools, colleges, students and their parents and carers*. DCSF, London
- Dearing R (1997) *Higher education in the learning society. Report of the national committee of enquiry into higher education*. HMSO, London
- Dollinger, M., Brown, J. L. and D'Angelo, B. (2022). *Co-Creating end-to-end student experiences in career development learning*. In: O'Shea S, Groves O, Austin K, Lamanna J (eds) *Career development learning and sustainability goals. sustainable development goals series*. Springer, Singapore. https://doi.org/10.1007/978-981-19-6637-8_3
- Ellis R, Han F (2020) Assessing university student collaboration in new ways. *Assess Eval High Educ* 46(4):509–524. <https://doi.org/10.1080/02602938.2020.1788504>
- Fincher S, Finlay J (2016) *Computing graduate employability: sharing practice*. Project report. Council of Professors and Heads of Computing, Kent, UK. <https://cphcuk.files.wordpress.com/2016/01/computinggraduateemployabilitysharingpractice.pdf>
- Fung D (2017) *A connected curriculum for higher education*. UCL Press, London, England
- Gibbs G (1988) *Learning by doing: a guide to teaching and learning methods*. Oxford Brookes Further Education Unit, Oxford
- González-Morales D, Moreno de Antonio LM, Roda García JL (2011) Teaching “soft” skills in Software Engineering. *IEEE global engineering education conference*. In: (EDUCON)—*Learning environments and ecosystems in engineering education*, pp630–637. ieeexplore.ieee.org/document/5773204/
- Hawkins P (1999) *The art of building windmills: career tactics for the 21st century*. Liverpool: Graduate into Employment Unit
- Hazzan O, Har-Shai G (2014) Teaching and learning computer science soft skills using soft skills: the students' perspective. In: *IGCSE '14 proceedings of the 45th ACM technical symposium on computer science education*, pp 567–572. <https://dl.acm.org/citation.cfm?id=2538885>

- Healey RL (2014) How engaged are undergraduate students in ethics and ethical thinking? An analysis of the ethical development of undergraduates by discipline. *Student Engagement Experience J3* (2). <https://doi.org/10.7190/seej.v3i2.93>
- HEAR (2008) Higher education achievement report. <https://www.hear.ac.uk>
- Helyer R (2015) Learning through reflection: the critical role of reflection in work-based learning (WBL). *J Work-Appl Manag* 7(1):15–27. Available at: www.emeraldinsight.com/2205-2062.htm
- Jackson D (2014) Employability skill development in work-integrated learning: barriers and best practice. *Stud High Educ*, 40(2):350–367. Available at: <https://doi.org/10.1080/03075079.2013.842221>
- Jung J (2022) Working to learn and learning to work: research on higher education and the world of work. *High Educ Res Dev* 41(1):92–106. <https://doi.org/10.1080/07294360.2021.2002274>
- Knight P, Yorke M (2004) *Learning, curriculum and employability in higher education*. Routledge Falmer, London
- Leckey JF, McGuigan MA (1997) Right tracks-wrong rails: the development of generic skills. *High Educ Res Hig Educ* 38(3):365–378. Available at: <https://doi.org/10.1023/A:1024902207836>
- Lowden K, Hall S, Elliot D, Lewin J (2011) *Employers' perceptions of the employability skills of new graduates*. Project Report. Edge Foundation, London, UK
- Marr B (2022) *Future skills*. Wiley
- Office for Students (2018) *Office for students strategy 2018 to 2021*. Available online at: <https://www.officeforstudents.org.uk/media/1435/ofs-strategy-2018-to-2021.pdf>
- Papert S, Harel S (1991) *Constructionism*. Ablex Publishing Corporation
- Peabody MA, Noyes S (2017) Reflective boot camp: adapting Lego Serious Play in higher education. *Reflective Pract* 18(2):232–243. Available at: <https://doi.org/10.1080/14623943.2016.1268117>
- Pedley-Smith S (2014) *Graduate recruitment, learning and development white paper*. Kaplan UK. https://kaplan.co.uk/docs/default-source/pdfs/kaplan_graduate_recruitment_whi_tepaper40FEB3E55577.pdf
- Pegg A, Waldock J, Hendy-Isaac S, Lawton R (2012) *Pedagogy for Employability*. Available at: <https://www.heacademy.ac.uk/knowledge-hub/defining-and-developing-your-approach-employability-framework-higher-education>
- QAA (2001) *Guidelines for HE progress files, digital education research archive*. Available at: <http://dera.ioe.ac.uk/8480/1/progfile2001.pdf>
- QAA (2009) *Personal development planning: guidance for institutional policy and practice in higher education*. Available at: <http://www.qaa.ac.uk/en/Publications/Documents/Personal-development-planning-guidance-for-institutional-policy-and-practice-in-higher-education.pdf>
- Roberts J (2015) *Experiential education in the college context*. Routledge
- Roberts J (2018) From the editor: the possibilities and limitations of experiential learning in higher education. *J Experiential Learn* 41(1):3–7. <https://doi.org/10.1177/1053825917751457>
- Schonell S, Macklin R (2019) *Work integrated learning initiatives: live case studies as a mainstream WIL assessment*. *Stud High Educ* 44(7):1197–1208. <https://doi.org/10.1080/03075079.2018.1425986>
- Schwaber K, Sutherland J (2017) *The scrum guide. The definitive guide to scrum: the rules of the game*. Available at: <https://www.scrumguides.org/>
- Scott FJ, Connell P, Thomson LA, Willison D (2019) Empowering students by enhancing their employability skills. *J Furth High Educ* 43(5):692–707. <https://doi.org/10.1080/0309877X.2017.1394989>
- Shadbolt N (2015) *Unemployment among computer science graduates—what does the data say?* HEFCE Blog. <http://blog.hefce.ac.uk/2015/07/08/unemployment-among-computer-science-graduates-what-does-the-data-say/>
- Shadbolt N (2016) *Shadbolt review of computer sciences degree accreditation and graduate employability*. https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/518575/ind-16-5-shadbolt-review-computer-science-graduate-employability.pdf

- Strachan R, Liyanage L, Casselden B, Penlington R (2011) Effectiveness of technology to support work-based learning: the stakeholders' perspective. *Res Learn Technol* 19:134–145. Available at: <https://journal.alt.ac.uk/index.php/rlt/article/view/729/961>
- Turner V, Böttcher A, Winter V (2012) Soft skill development along the education path. Evaluating expectations on and perceptions of student competencies in software engineering education. *Int J Eng Pedagogy* 2(3). <http://www.online-journals.org/index.php/i-jep/article/view/2148/2269>
- UKCES (2015a) UK Employer Skills Survey 2015a: UK results <https://www.gov.uk/government/publications/ukces-employer-skills-survey-2015a-uk-report>
- UKCES (2015b) Sector insights: skills and performance challenges in the digital and creative sector. <https://www.gov.uk/government/publications/sector-insights-skills-and-performance-challenges-in-the-digital-and-creative-sector>
- Universities UK and UKCES (2014) Forging Futures: building higher level skills through university and employer collaboration. https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/356749/FF_FinalReport_Digital_190914.pdf
- Universities UK (2016) Higher Education in England: provision, skills and graduates. <http://www.universitiesuk.ac.uk/policy-and-analysis/reports/Documents/2016/higher-education-in-england-provision-skills-and-graduates.pdf>
- University Alliance (2015) Mind the gap: Engaging employers to secure the future of STEM in higher education. <https://www.unialliance.ac.uk/2015/10/09/mind-the-gap-engaging-employers-to-secure-the-future-of-stem-in-higher-education-2/>
- Vogler JS, Thompson P, Davis DW, Mayfield BE, Finley PM, Yasseri D (2017) The hard work of soft skills: augmenting the project-based learning experience with interdisciplinary teamwork. *Instr Sci* 46(3):457–488. <https://doi.org/10.1007/s11251-017-9438-9>
- Wilson T (2012) The Wilson review: a review of business–university collaboration. Available at: https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/32383/12-610-wilson-review-business-university-collaboration.pdf
- World Economic Forum (2016) The future of jobs employment, skills and workforce strategy for the fourth industrial revolution. [Online] http://www3.weforum.org/docs/WEF_Future_of_Jobs.pdf
- World Economic Forum (2020) The future of jobs report 2020. <https://www.weforum.org/reports/the-future-of-jobs-report-2020/>
- Yu L, Adaikkalavan R (2016) Developing soft skills by applying problem-based learning in software engineering education. In Railean E, Walker G, Elci A, Jackson L (eds) *Handbook of research on applied learning theory and design in modern education*. IGI Global, Hershey, PA

Chapter 18

Professionalism and Online Presence



Sue Beckingham

Abstract Building and maintaining networks is a crucial part of being a professional, and increasingly this is taking place online as well as in person. Understanding how to behave in these different spaces is vital, as the implications of an unprofessional online presence can have a detrimental effect for anyone. Changes in the way employers recruit include the ease of searching and identifying suitable candidates via their LinkedIn or Handshake profiles. Having an active and visible presence is therefore important for students seeking placements and for recent graduates. This chapter provides tips on developing social media profiles to enhance employability and guidance on professional social networking, social media policy, copyright and data protection.

Keywords Online presence · Social media · Digital footprint · Employability

18.1 Introduction

Working-age graduates and postgraduates continue to have higher employment rates than non-graduates. However the findings of the Graduate Labour Market Statistics (GLMS) in 2021 found those 65.2% of undergraduates and 77.4% postgraduates were in high-skilled employment (Gov.UK, 2022). In 2021 57% of 25–34 year-olds in the UK were educated to tertiary level (OECD 2022), there are concerns that there are too many graduates looking for far too few jobs. However Universities UK (2022) have identified that this is not the case. They found that data from the Institute of Student Employers (2022) shows that the graduate market has recovered and that graduate job vacancies are 20% higher than pre-pandemic. They go on to report that compared to 2019, graduate vacancies have risen by 67% in the built environment, 42% in digital and IT, and 24% in health and pharmaceuticals. Overall in the period June to August 2022 the number of job vacancies was 1.2 million (Office for National Statistics 2022).

S. Beckingham (✉)
Sheffield Hallam University, Sheffield, England
e-mail: S.Beckingham@shu.ac.uk

What is significant in the Institute of Student Employers report is that 18% of employers surveyed felt that the quality of graduates had decreased and that they lacked application readiness. In a study conducted by Handshake 59% of employers felt that the impact of Covid-19 has resulted in graduates being prevented from gaining work experience through placements and internships and that as a result they are less likely to have the skills needed in the workplace (Handshake 2022a). Furthermore student confidence has declined, and they are less likely to engage with the services Careers teams offer (Handshake 2022b). Handshake is a career network and career management platform connecting over 1400 universities in the UK and US with 750 K + employers who are actively looking for interns and graduates (Handshake 2022c).

The digital divide and digital poverty became ever more evident during Covid-19 with students attempting to undertake their online learning using a mobile phone as they did not have access to a laptop that in turn could provide them with access to the software and hardware they needed to engage with digital learning. Jisc (2021:9) undertook a survey on student digital experiences which found the following problems were experienced when learning online:

- 63% poor Wi-Fi connection
- 30% access to online platforms/services
- 24% mobile data costs
- 22% need specialist software
- 21% no safe, private area to work
- 15% no suitable computer/device

Whilst for many teaching has now returned to the classroom, there is still work to do to ensure our students are best prepared to apply for placements and graduate roles. Key to this is being able to identify what skills employers are looking for and how to evidence these in job applications. Secondly it is important for students to learn how to use this information to develop a professional online presence (Lancaster 2014) and how social media can help them to research job openings and how to network. Sutherland and Ho (2017) propose that proficient use of social media can enhance the transferable skills that are associated with professionalism and valued by employers.

Students also need to participate in activities both within and outside of the curriculum to develop confident skills, networks and communities. In the Student Futures Manifesto 2022 the UPP Foundation cite Professor Sir Chris Husbands, Vice Chancellor, Sheffield Hallam University who stated:

The real jolt of the pandemic wasn't to curriculum delivery... the real jolt of the pandemic was to the more 'immersive' elements of university – the engagement in extended learning and activities which widened horizons, developed skills and so on. I've always felt this is a social justice issue – some students come into HE with those things already in their kitbag of cultural and social capital. The pandemic ripped into this. We have to rebuild it.

(UPP Foundation 2022:58)

Arnedillo-Sanchez and Tseloudi (2018:3) expand on this and argue that there are four dimensions of capital. In addition to cultural capital (lived experiences

of an individual and social capital (an individual's social network), an individual must demonstrate their human capital (subject knowledge and attributes needed to function in the workplace, and psychological capital (the ability to provide optimal performance on the job).

Jackson (2016) reminds us that communication, teamworking and self-management continue to be crucial employability skills, but also highlight that it is vital students are able to construct their pre-professional identity (PPI). This is described as “an understanding of and connection with the skills, qualities, conduct, culture and ideology of a student's intended profession.” (Jackson 2016:926).

18.2 Knowing What Graduate Schemes and Jobs Are Available

Every university will have a valued Careers and Employability team who are there to support and guide students to attain a graduate job with access to one-to-one meetings and group workshops for CV checks, mock interviews and more. Many will have access to a university database of jobs. Increasingly universities are subscribing to Handshake where students and graduates can choose to connect with employers and message them within Handshake. This opens up the opportunity for them to say a little about themselves and learn more about a specific role.

The Prospects website <https://www.prospects.ac.uk/> is also a useful source of information providing careers advice, information about graduate schemes and suggestions for careers with specific ‘what can I do with me degree in [add degree]’. Examples of job profiles include computer science, games design, information systems and software engineering. There are of course a multitude of recruitment sites promoting jobs.

18.3 Social Media

There are assumptions made that our tech savvy computing students are aware of the role that social media can play to develop a professional online presence, however this is not always the case. Whilst many may have experience of using it socially, consideration of how it may be used in a professional context and in relation to preparing for placement or graduate job applications may be lacking or not fully understood. Bridgstock (2019) states that students are not all conversant with how to use social media for professional purposes and lack the confidence and skills to utilise these spaces for career management. It is incumbent upon us as educators to support students to learn how these channels can be used effectively and raise awareness of how organisations use social media in the context of employability. Furthermore

research undertaken by Davis et al (2020) highlights that there are networking benefits for postgraduate students who whilst already may be in employment were able to benefit from career sponsorship, social support, job search assistance, business assistance, protection and political guidance and information and new ideas.

On the flip side it is equally important for students to be aware of the implications of public posts which openly share aspects of their social lives and how these might adversely affect their future employment. Employers can easily undertake a Google search on applicants, and this can quickly present any open social media profiles. Research undertaken by Darr and King (2018) highlight a range of examples where students are openly sharing unprofessional content. These included inappropriate and offensive language, drunkenness and overt sexuality. Holt (2014) highlights that deleting online postings does not always mean they cannot still be found. Students need to be reminded that online activity can leave digital footprints and to be mindful what they share. It may not be ethical but the use of social media as a recruitment screening tool can influence hiring decisions (Roth et al. 2016; Arnedillo-Sanchez and Tseloudi 2017) and in this context exclude potential candidates from ever reaching an interview or job offer.

18.3.1 *Careers Advice*

In addition to a website, organisations such as Prospects and Handshake also make use of their own social media channels to share tips. Students can access suggestions that will help them develop an online presence, careers advice, job openings and more. These can provide regular updates and where optimised provide helpful notifications of new content to a smart phone. Table 18.1 provides some examples of different social media use.

Table 18.1 Examples of social media use

| | |
|-----------|---|
| Handshake | https://www.instagram.com/joinhandshake/ https://www.tiktok.com/joinhandshake https://twitter.com/joinhandshake https://facebook.com/joinhandshake https://www.linkedin.com/showcase/handshake-students/about/ |
| Prospects | https://www.instagram.com/prospects_ac_uk/ https://www.twitter.com/prospects https://www.facebook.com/prospects.ac.uk/ https://www.linkedin.com/company/prospects-ac-uk/ https://www.youtube.com/ProspectsJisc |

18.3.2 *Companies Offering Employment*

Companies looking to recruit students for placements, graduate schemes and other roles now make use of LinkedIn to post jobs but can also use this space to headhunt and seek out prospective candidates that meet their criteria. LinkedIn is the world's largest professional network with 875 million members in more than 200 countries and territories worldwide (LinkedIn 2022). Those responsible for recruitment can choose to upgrade and subscribe to LinkedIn Recruiter giving them access to search and review profiles beyond their own personal connections and 'InMail' individuals through LinkedIn's internal messaging service.

It is important to note that the use of social media has had an impact on recruitment globally. Koch et al (2018) for example state the use of social media in South Africa has levelled the playing field, enabling them to find a much wider pool of candidates. Ismail and Koshy (2017) undertook research that looked at Abu Dhabi and Dubai, and found that this form of online recruitment was more efficient and was not limited by a geographical area, enabling a worldwide reach.

Many organisations will have their own company LinkedIn page. LinkedIn users can follow these pages. In addition organisations may choose to also use other social media channels as a marketing tool to promote openings and share information about their staff and the culture of the organisation (Dutta 2014). Here are some examples:

IBM make use of Twitter, Instagram and YouTube to showcase role profiles of their staff.

<https://twitter.com/lifeatibm>

<https://www.instagram.com/lifeatibm/>

<https://www.youtube.com/LifeAtIBM>

Cisco share company information on Facebook, YouTube, Twitter, Instagram and Snapchat.

<https://www.facebook.com/WeAreCisco/>

<http://youtube.com/wearecisco>

<https://twitter.com/WeAreCisco>

<https://www.instagram.com/wearecisco/>

<https://www.snapchat.com/add/wearecisco>

Dell Technologies provides company insights and careers opportunities using Facebook, Twitter, Instagram, and YouTube.

<https://www.facebook.com/DellTechCareers/>

<https://twitter.com/delltechcareers>

<https://www.instagram.com/delltechcareers/>

<https://www.youtube.com/user/DellCareers>

Students can follow these social media sites to learn about company culture and open job opportunities.

18.3.3 Professional Social Networking

Social media sites can provide access to promoted careers events and job opportunities, including formal placements, internships, and entrepreneurship schemes. LinkedIn for example is used by professionals for both knowledge finding and sharing, a space to both raise and answer questions, engage in discussions, and give or receive feedback on projects. The space provides opportunities to connect and network with other professionals and learn about their companies. Bridgstock and Jackson (2019) suggest that there are three types of employability strategies, and these are programmes that aim to develop short term employment outcomes, those that prepare students for a profession and thirdly activities that focus on life-long learning and career development. When aligning this to professional social networking it is important to remember that developing an online presence and network is an iterative process, serving different purposes dependant on the stage of an individual's career. Short term employment could be work experience, volunteering, internships and placements. Students approaching graduation may be considering graduate schemes. Postgraduate students may already have a professional role but seek new opportunities. All will benefit from professional social networking.

There tends to be three groups of people when it comes to online professional social networking (OPSN): the advocates, the dabbler and the unengaged (Beckingham 2019).

- Advocates: well developed profile, active user, connected, sharing and collaborating.
- Dabblers: sporadic visits, low contribution, profile may only be partially set up.
- Unengaged: unaware of potential, not yet part of the conversation.

The most worrying are the dabblers. An incomplete profile which may not have been proofread, can look very unprofessional. It is important to remember that once an individual creates an online profile, this forms part of their online presence, irrespective of whether the social media space is used or not. Whilst the unengaged are missing out on potential opportunities at least there is no digital trail that might come back to bite them.

For anyone new to OPSN, then it can be beneficial to look at how professionals in your field use these spaces. Habets et al. (2021) make the case for training sessions on the use of social media to maintain current networks and resources, as well as learning how to develop new professional contacts. Whilst most profiles begin by

making connections with strong ties, those people you already know, there is much to be gained by extending your professional network to connect with others who share interests but also offer additional perspectives (Benson et al. 2014). Extending your network enables you to take advantage of the experiences Granovetter (1973) called ‘weak ties’.

Making connections with potential employers is a useful way to show that you have an interest in their organisation. Whilst this may seem quite daunting to start with there are opportunities that will present themselves. For example when attending a careers fair and engaging in conversation, capturing the name of the person you spoke to is a must. This opens the chance to send a LinkedIn invite to connect with that person and follow-up with any further questions. Course Leaders can consider creating a LinkedIn group for their course and invite students and alumni to this. Not only does this create a social network to share news and job opportunities, but it is also useful to track career trajectories of alumni (Li et al. 2016). Furthermore current students can learn from alumni. Their profiles can be used as case studies and evaluates on their strengths and weaknesses. This will help students when completing their own profiles (Case et al. 2016) and can be useful to learn what skills alumni use in their profiles as employers and recruitment companies will search for candidates using skills as keywords (Zide et al. 2014).

18.4 Practical Tips

In this final section there are some suggestions on how to make the most of your professional social media profiles. Employers find potential candidates based on the career interests placed in user profiles. It is helpful to be clear what kind of job you are looking for. For example:

- Job type: Do you want an internship, part-time job, or full-time job?
- Location: State the towns/cities you want to work in
- Job roles: Select the types of jobs you are interested in, e.g. software engineer, data analyst, web designer.

In Table 18.2 there are tips on how to maximise the sections within a typical LinkedIn profile. Aspects from these can also be applied to a Handshake profile. For other online professional social networking sites it is very important to complete the bio section of your user profile. This should provide information about who you are—for example current BSc [add degree] student or graduate and what your current interests are—for example looking for a placement or graduate role in [add interest].

Recommendations from Marr (2022: 194–195) include:

- Cleaning up your profiles by deleting any content that you wouldn’t want potential employers to see.
- Be yourself. While you want to cultivate a personal brand, its important to let your personality shine through in your social media posts.
- Share what you are learning.

Table 18.2 Making the most of the sections in a LinkedIn profile (Beckingham 2022)

Headline

This appears under your name. Use this to say what you do (name of your course) and what you want to do (get a placement/graduate role in...). Your headline is likely to be the first thing a recruiter or employer will see

Photo

Make sure you have a professional looking ‘head and shoulders’ photo. This can be easily taken by a friend against a white or light-coloured wall. Make sure you are dressed smartly, and you smile. Don’t be tempted to use a cropped version of your favourite holiday photo!

Summary

This section is your opportunity to share your experience (at university, part-time work or volunteering), the skills you have developed and what your career aspirations are. Tell your story! What projects have you worked on, are you a member of a club, society or do you engage in other extracurricular activities? What skills have you gained through working on university projects?

When writing this you may identify there is a gap in your experience. Take the opportunity to fix this by looking for volunteering or paid part-time roles. This could be becoming a Course Rep, Student Ambassador, joining a Society or taking a part-time job

Imagine this is your elevator pitch with your future employer – you have literally a few minutes to tell them about yourself. Write your summary in first person. Use the STAR technique (Situation, Task, Action, Result) to share real examples to demonstrate your skills and attributes. In short say what the situation was, explain the task and what the goal was, what action did you take to achieve this, and what was the result of the action

State here that you are open for placement or graduate opportunities, and that you are happy to be contacted

Skills

Adding specific skills to your LinkedIn profile allows you to showcase your abilities to other LinkedIn members, such as peers, colleagues, managers and prospective employers. Use this section to add both the technical and soft skills you feel you can demonstrate. Be strategic and add the ones you know are relevant to the industry you wish to work in. Over time you will need to revisit this section and add new skills

Enable the #OpenToWork feature from your LinkedIn profile

This tells recruiters and employers that you are actively looking for work

Recommendations

If you have work experience, reach out to your colleagues and ask them if they would be happy to give you a recommendation. This is a valuable way to endorse your experience

Make Connections

This may seem obvious, but over time you should aim to develop your connections beyond those you immediately know and work with. For example you may have attended a Careers Fair at university and talked to an employee representing a company you are interested in working for. You could follow this up by sending a personalised connection request saying for example “It was good to meet you at the recent Careers event... Thank you for the advice.”

You can also follow Companies on LinkedIn. This will give you updates on information shared by those organisations

Discussion groups

Join professional discussion groups. These might include discipline and interest groups, course groups which include alumni, as well as other industry networking opportunities

18.5 Summary

It is vital that we support students and those that have graduated and not yet attained employment in the different ways they can look for job opportunities but also how to prepare themselves to showcase their skills in the most effective ways. Encouraging students to meet with a Careers/Employability Adviser to get feedback on their professional online profiles is important but should also be supplemented with examples from tutors and industry experts. Inviting alumni back to share with current students how they have utilised social media and professional online profiles can be very powerful.

Finally it is important to remember that

- building professional networks is an ongoing process,
- updating our professional online profiles is an ongoing process,
- learning from others is an ongoing process.

References

- Arnedillo-Sanchez I, Tseloudi C (2017) Escort: employability skills content curation tool for social media profiles. In: European conference on technology enhanced learning. Springer. https://doi.org/10.1007/978-3-319-66610-5_55
- Arnedillo-Sanchez I, de Aldama C, Tseloudi C (2018) rESSuME: employability skills social media survey. *Int J Manpow* 39(8):1080–1096. <https://doi.org/10.1108/IJM-10-2018-0333>
- Beckingham S (2019) Developing a professional online presence and effective network. In: Rowell C (ed) *Social media in higher education: case studies, reflections and analysis*. Open Book Publishers
- Beckingham S (2022) LinkedIn tips for students looking for placements or graduate job vacancies. *Social Media for Learning*. <https://socialmediaforlearning.com/2022/02/13/linkedin-tips-for-students-looking-for-placements-or-graduate-job-vacancies/>
- Benson V, Morgan SJ, Filippaios F (2014) Social career management: social media and employability skills gap. *Comput Hum Behav* 30:519–525. <https://doi.org/10.1016/j.chb.2013.06.015>
- Bridgstock R (2019) Employability and career development learning through social media: exploring the potential of linkedin. In: Higgs J, Cork S, Horsfall D (eds) *Challenging future practice possibilities*. Sense-Brill Publishers
- Bridgstock R, Jackson D (2019) Strategic institutional approaches to graduate employability: navigating meanings, measurements and what really matters. *J High Educ Policy Manag* 41(5):468–484. <https://doi.org/10.1080/1360080x.2019.1646378>
- Case TL, Han H-J, Rimes E (2016) Career paths of computing program graduates: a linkedin analysis. In: SAIS 2016 Proceedings. pp 21. <http://aisel.aisnet.org/sais2016/21>
- Darr T, King SP (2018) To post or not to post: perceptions of appropriateness on social media. *Modern Psychol Stud* 24(1):5. <https://scholar.utc.edu/mps/vol24/iss1/5>
- Davis J, Wolff H, Forret ML, Sullivan SE (2020) Networking via linkedin: an examination of usage and career benefits. *J Vocational Behav* 118:1–15. <https://doi.org/10.1016/j.jvb.2020.103396>
- Dutta D (2014) Tweet your tune—social media, the new pied piper in talent acquisition. *Vikalpa* 39(3):93–104. <https://doi.org/10.1177/0256090920140307>
- Gov.UK (2022) Graduate labour market statistics: 2021. <https://www.gov.uk/government/statistics/graduate-labour-market-statistics-2021>

- Granovetter MS (1973) The strength of weak ties. *Am J Sociol* 78(6):1360–1380. <https://doi.org/10.1086/225469>stron
- Guangzhi Zheng L, Peltzverger S, Zhang C (2016) Career trajectory analysis of information technology alumni: a linkedin perspective. In: SIGITE '16: proceedings of the 17th annual conference on information technology education. <https://doi.org/10.1145/2978192.2978221>
- Habets O, van der Heijden B, Ramzy O, Stoffers J, Peters P (2021) Employable through social media: an intervention study. *Sustainability* 13(9):5093. <https://doi.org/10.3390/su13095093>
- Handshake (2022a) Graduate employer priorities 2023 report. <https://www.joinhandshake.co.uk/gep>
- Handshake (2022b) Careers2032 Report. <https://www.joinhandshake.co.uk/careers2032>
- Handshake (2022c) About. <https://joinhandshake.com/about/>
- Holt J (2014) Legal aspects of social media. Internet Specialist Group and the Law Specialist Group of the BCS. <https://www.bcs.org/media/3416/legal-aspects-of-social-media-151214.pdf>
- Institute of Student Employers (2022) ISE vacancy survey. <https://ise.org.uk/page/graduate-job-vacancies-20-higher-than-pre-pandemic>
- Ismail I, Koshy S (2017) The use of linkedin as a recruitment tool in the UAE: an evaluation. In: International conference on education, humanities and management (ICEHM-17), pp 113–119. <https://ro.uow.edu.au/dubaipapers/874>
- Jackson D (2016) Re-conceptualising graduate employability: the importance of pre-professional Identity. *High Educ Res Dev* 35(5):925–939. <https://doi.org/10.1080/07294360.2016.1139551>
- Jisc (2021) Student digital experience insights survey 2020/21: UK higher education (HE) survey findings. <https://repository.jisc.ac.uk/8487/1/Student%20DEI%20HE%20report%2021%20Final.pdf>
- Koch T, Gerber C, de Klerk JJ (2018) The impact of social media on recruitment: are you LinkedIn? *SA J Human Resour Managem.* <https://doi.org/10.4102/sajhrm.v16i0.861>
- Lancaster T (2014) Teaching students about online professionalism: Enhancing student employability through social media. In: Management Association I (ed) Digital arts and entertainment: concepts, methodologies, tools, and applications. IGI Global. <https://doi.org/10.4018/978-1-4666-6114-1>
- LinkedIn (2022) About LinkedIn. <https://about.linkedin.com/>
- Li L, Zheng G, Peltzverger S, Zhang C (2016) Career trajectory analysis of information technology alumni: A LinkedIn perspective. In: SIGITE '16. Proceedings of the 17th Annual Conference on Information Technology Education, September 2016, pp 2–6. <https://doi.org/10.1145/2978192.2978221>
- Marr B (2022) Future skills. Wiley
- OECD (2022) Education at a Glance 2022: OECD indicators. OECD Publishing, Paris. <https://doi.org/10.1787/3197152b-en>
- Office for National Statistics (2022) Labour market overview, UK: September 2022. <https://www.ons.gov.uk/employmentandlabourmarket/peopleinwork/employmentandemployeetypes/bulletins/uklabourmarket/september2022>
- Roth PL, Bobka P, Van Iddekinge CH, Thatcher JB (2013) Social media in employee-selection-related decisions: a research agenda for uncharted territory. *J Manag* 42(1):269–298. <https://doi.org/10.1177/0149206313503018>
- Roth PL, Bobko P, Van Iddekinge CH, Thatcher JB (2016) Social media in employee-selection-related decisions. *J Manag* 42(1):269–298. <https://doi.org/10.1177/0149206313503018>
- Sutherland KE, Ho S (2017) Undergraduate perceptions of social media proficiency and graduate employability—a pilot study. *Higher Educ Skills and Work-Based Learn* 7(3):261–274. <https://doi.org/10.1108/HESWBL-02-2017-0018>

- Universities UK (2022) Busting graduate job myths. <https://www.universitiesuk.ac.uk/what-we-do/policy-and-research/publications/busting-graduate-job-myths>
- UPP Foundation (2022) A student futures manifesto. <https://upp-foundation.org/wp-content/uploads/2022/02/A-Student-Futures-Manifesto-Final-Report-of-the-Student-Futures-Commission.pdf>
- Zide J, Elman B, Shahani-Denning C (2014) LinkedIn and recruitment: how profiles differ across occupations. *Empl Relat* 36(5):583–604. <https://doi.org/10.1108/ER-07-2013-0086>

Postscript

Clive Rosen

When friends or acquaintances ask me what I do for a living, I usually say “I teach computing”. That usually ends that line of conversation and we move on to something more interesting like politics or religion. I don’t try to expand and say “I teach programming”, “software design”, “programme design” or “software systems development” because most people glaze over and seem to think “computing” is akin to quantum mechanics and as equally unfathomable.

This is both a blessing and a curse; a blessing because it affords us cognoscenti the status of shamans playing with the bones of dead animals and creating a future few could dream of; a curse because it excludes us of the real world and creates the impression that we’re doing something impossible for everyday folk to understand. This inaccessibility puts many people off and consigns interest in the subject area to nerds and weirdos nicely characterised in the film *Ready Player One*.

But “computing” is a mongrel subject as I think this book demonstrates. It borrows concepts from many other subject areas, from maths and statistics, through business and engineering to psychology and philosophy, and this too is a blessing and a curse. It’s a blessing because it provides us with limitless concepts from which we can draw, and a curse because neither we, the institutions we work in, nor the Dewey classification system have any idea where we truly belong.

During my time as an academic, my faculty was bounced round the university from one department to another like a beach ball in a football crowd. We started off as an independent school, then lumped in with engineering and technology, next on to business (to which law was then appended) and finally back to a school of computing and mathematics. (I never quite made it into social sciences faculty, but assimilating us with psychology was discussed.) This is not only a problem because of the lack of continuity and changes in ethos and direction it creates, but because it inhibits the development of a recognised computing profession.

Professional status is important because it is a first step towards the establishment of recognised standards, inhibiting the adoption of quack remedies.¹ Ultimately standards help to prevent catastrophic failure. When IT systems are at the heart of almost every working system in the modern world, from trade and finance, to international security, social welfare and health, education, communications, travel and media, the lack of standards make for a disaster waiting to happen. The consequences of a failure of any one of these systems would be unpredictable, but almost certainly not good.²

Traditionally, academics have been central to establishing standards and responsible for passing them on to their students. In programming, we have earlier generations of academics to thank for the demise of the GOTO command and principles of modularisation, encapsulation etc. Higher education has also been the place where students become inculcated with the ethical standards of the profession. When students can enter professional life without going through this process, these principles are undermined. So the fact that anyone can practice in any field of computing with a qualification in any subject area or even without a professional qualification of any kind, it should be and is disturbing.

However, whilst academics in computing might express concern in this regard, they have largely not helped themselves. The concentration on practical skills³ to the detriment of theoretical principles has resulted in theory⁴ being underdeveloped. Without having core (and unique) principles that neophyte practitioners must demonstrate command of, it is impossible to prevent entrants without this knowledge entering the profession (Abbott 1988). A problem therefore of our own making. Problem-based learning, as promoted through this book, is all very well, but if it does not lead to students reflecting on the principles they used to solve the problem, it will leave them without the ability to transfer their learning to other scenarios. A number of conceptual constructs exist within each sub domain of computing and these are often well articulated, but what of the discipline as a whole? Can we identify some universal principles that graduates should not leave home without? I would propose three to begin with, namely:-

- The principle of user led development
- The concept of systems thinking and systems integration.
- The capability to design appropriate systems level testing

¹ Remedies that have little or no scientific basis.

² We have seen in other areas that when standards are not maintained the consequences for ordinary people are calamitous, but generally localised. Failure in an IT system are potentially boundless.

³ Vis-A-vis programming, OOD, database development, web design etc.

⁴ Theory in this sense is a number of models of how the world works from the perspective of the discipline.

The Principle of User Led Development

I would like to draw a distinction between “user led development” and the “principle of user led development”. The former implies a (usually brief) consultation with a client or perhaps in some circumstances, some market research followed by the production of a requirements specification. (In Agile this might be equated to the project backlog (Sedano et al. 2019)). The latter is a frame of mind. In the former, developers might consult users. In ideal circumstances, this would be on a regular basis. They may even consult a wide range of stakeholders (Kotonya and Sommerville 1998; Van Lamsweerde 2009) and follow a recognised process, although anecdotal evidence suggests that for non-mission critical developments all these activities are uncommon. Requirements elicitation and evaluation are self-evidently essential and all too often not given the time, resources and attention they should have. However, even when sufficient effort is expended, there is a tendency to concentrate on product functionality that is relatively straightforward to specify compared with the non-functional, quality aspects of the development (Inayat et al. 2015; Alsaqaf et al. 2017; Olsson et al. 2022). These requirements are difficult to identify, almost impossible to specify in precise terms and critical to the success or failure of the development. This is well known, but all too often ignored especially by developers who have entered the profession without a computing qualification. And this is why the principle of user led development is so important.

The principle of user led development is about thinking about how users will use the product. Who are the users in all their variations and diversity? What are their problems? How can we capture their experience? These thoughts should be at the forefront of a developers mind throughout the project, whether they be senior analysts, product designers, coders or testers. It is about walking the miles in the users’ shoes, empathising with them and offering solutions that they will be happy with.⁵ Sadly, the traditional software engineering approach is as many miles away from the principle of user led development. It tends towards a function led perspective as opposed to a person led perspective. In software systems development, form is as important as function and needs to be recognised as such. Inculcating students into this way of thinking should be one of the prime functions of computer science educators and provide a distinct differentiation between a graduate with a computer science related degree and anyone coming from a different background. Sadly, at present, all too often, it is people coming from non-computing backgrounds that have better skills of empathy and understanding than computing students. Whilst this remains the case, users will continue to be dissatisfied with the products of our labours.

⁵ Shuraida and Barki (2013) Explore the issues associated with the relationships between systems analysts and user very comprehensively.

The Principle of Systems Thinking

All too often software developers focus on the solution of the problem they are confronted with without looking at the bigger picture. They may even be encouraged to do this by commissioning agents who tell us ‘this is the problem, please fix it’. The problem is that, as no man is an island, no software development exists without interfacing with a wider system. This is especially so today as we are confronted with the problems of legacy systems integration. Yet, whilst the concept of systems thinking has been around a considerable time (Beer 1985; Checkland 1999)⁶ it still seems to be regarded as the quirky domain of social workers and pseudo-scientists. Granted it is hard to grasp the concepts that:-

- Every (viable) system is a member of a system chain, receiving inputs from earlier chain members and providing outputs to later chain members
- Every system/subsystem is both parent to a subsystem with which it interacts and child of a super system with which it also interacts
- That the systems/subsystems can be both technological and social
- That the developers themselves are amongst the said systems

Nevertheless, every developer should be aware of this complexity and that their intervention has consequences beyond the changes they are making. They should recognise the responsibility that this places on them to investigate the intended and unintended consequences of this intervention. Granted also that the infinite recursion inherent in this model makes this tricky, but it does not absolve them from trying to understand. The AI community are already aware of the dilemmas created by technological innovation, but the consequences of the development of social media I think have probably caught us all by surprise and should be regarded as a cautionary tale. This is not to argue against technological innovation, which is inevitable, but to at least consider its effects in advance. Working out what might be the wider connotations of an upgrade to a web site might be less obvious and less significant, but nevertheless worth thinking about.

The other side of this coin should also be considered; how the interactions with the wider environment might affect the operation of the software system. In particular, how do the social systems affect the technological systems? This goes beyond the user interface to an appreciation of the ecological environment in which the system will operate. Should computing students qualify with these skills and knowledge, it would provide them with a significant advantage over competing students from other domains. It would however require a slackening in the strict adherence to the “scientific” model of computing related courses loved by many faculties at present.⁷

⁶ Beer’s thinking in fact goes back thirty years before this Beer (1984).

⁷ Being aware of systems theory, I am also aware of how difficult such a transition might be in practice.

The Requirement for Systems Level Testing

Whilst the previous two points might seem to some highly esoteric, this point might seem profoundly prosaic, but it is in fact a reiteration of those two points. As with “test driven software”, a technique recommended for agile development, the test should be written before the software to ensure it meets the specification. The principle of system level testing is that the completed software should satisfy the end user(s). Too often in software development, the development time and cost exceed the budget (Johnson 2020).⁸ To recover some of this overspend and schedule slippage, there is a tendency for the testing to be curtailed. This probably wouldn’t happen in the development of any other product and may even be illegal for some, so the question arises, why is this tolerated in software development? I would contend that the main reason is the failure to specify the product properly in the first place, and here I am referring to the non-functional requirements. The failure to understand users’ needs and pay sufficient attention to them leads to the attitude that as long as the product does what it says on the tin, everything is ok. But, for software this may be a necessity, but it is not sufficient. The requirement is not only what it does, but how well it does it, in the environment in which it will operate; that is at the systems level. So, it is easy to say “we’ve tested it and no faults were found”, but when it comes to the operation of the product in the real world, the whole thing falls apart. Recriminations then fly around between consumer and producer, the later arguing that they produced what you asked for, but not what was needed.

Good testing is an art and requires imagination, the imagination to think what could happen and what might go wrong. Systems testing requires this imagination at the systems level thus requiring the tester to engage in systems thinking, to understand the user and the users’ environments and to model these potential scenarios; combining the skill of being able to empathise with the user and understand the complexity and interconnectivity of the system.

So, what should we require of our students and hence what do we need to teach them? In a nutshell, I would suggest: the ability to construct a conceptual model of the system that includes the quality requirements and how the system fits in and interacts with its environment. They also need the ability to share this model accurately and with as little ambiguity as possible, with other developers. In order to achieve these objectives, they need to be able to:-

1. Communicate with stakeholders at all levels of authority and influence.
2. Communicate effectively with other developers both as a listener and interpreter of their conceptual model, and to be capable of articulating their own conceptual model.
3. Imagine what ifs and what if nots.
4. Explore interconnectivities down the rabbit hole and come back up safely.

⁸ The basis of figures for project failure / challenge that the Standish group report in their series of Chaos reports have long been disputed but remain one of the most reliable sources for this information.

5. Be comfortable with big picture thinking.
6. Enter the world view (wettanschauung) of others.

Not much to ask.

These are not skills usually associated with the weirdos and nerds of popular culture but of open minded, creative and sensitive people with the added capability to think logically. Enabling people to develop these skills though is not so much a matter of changing the curriculum as of adapting the process of curriculum delivery to encourage creative thinking, team working and problem solving. This book has reviewed a number of these methods; multitasking (Liz Coulter-Smith), active learning (Diane Kitchen), flipped classroom (Michael O'Grady) and many more throughout the book, but there are a couple more things that need to happen too. One is a desire to embrace these approaches by instructors in HE. This might take some time, but it is hoped that this book will contribute towards facilitating this change. A second change might be even more difficult to engineer. We need to rebrand computing courses and update the sales pitch to prospective students to attract students with a greater interest in people than in technology. Given the discipline's current attachment to the scientific paradigm and narrow focus on practical skills, it is unlikely that this will happen quickly or soon. For those of us who think this is the way forward, it may be possible to introduce some marginal change and hope we get there in the end. I suspect though that it will be up to academics with exceptional communication skills to initiate this change. As the French may say, bon chance et bon courage.

References

- Abbott AD (1988) *The system of professions: an essay on the division of expert labor*. University of Chicago Press, London
- Alsaqaf W, Maya D, Wieringa R (2017) Quality requirements in large-scale distributed agile projects—a systematic literature review. In: Grünbacher P, Perini A (eds) *Requirements engineering: foundation for software quality*. REFSQ 2017. Lecture notes in computer science, vol 10153. Springer, Cham, pp 219–234
- Beer S (1984) The viable system model: its provenance, development, methodology and pathology. *J Oper Res* 35(1):7–25
- Beer S (1985) *Diagnosing the system for organisations*. Wiley, Chichester
- Checkland P (1999) *Systems thinking, systems practice*. Wiley, Chichester
- Eveleens JL, Verhoef C (2010) The rise and fall of the Chaos report figures. *IEEE Softw* 27(1):30–36
- Inayat I, Salim SS, Marczak S, Daneva M, Shamshirband S (2015) A systematic literature review on agile requirements engineering practices and challenges. *Comput Hum Behav* 51:915–929
- Johnson J (2020) *CHAOS2020 beyond infinity*, Standish Group
- Kotonya G, Sommerville I (1998) *Requirements engineering*. Wiley, Chichester
- Olsson T, Sentilles S, Papatheocharous E (2022) A systematic literature review of empirical research on quality requirements. *Requirements Eng* 27(2):249–271

- Rosen C (2020) Guide to software systems development; connecting novel theory to current practice. Springer, Cham
- Sedano T, Ralph P, Péraire C (2019) The product backlog. In: 2019 IEEE/ACM 41st international conference on software engineering (ICSE), Montreal, QC, Canada, IEEE
- Shuraida S, Barki H (2013) The influence of analyst communication in IS projects. *J Assoc Inf Syst* 14(9):482–520
- Van Lamsweerde A (2009) Requirements engineering. Wiley, Chichester

Index

A

Academic integrity, 72–74
Academic misconduct *see*, Academic Integrity
Academic performance, 4
Accountability, 225, 235, 237, 240, 242, 245, 246
Action plan, 197, 211, 266, 267
Activity diagram, 120, 121
Alumni, 261, 281–283
Animation, 21, 23, 24, 88, 103, 104, 106
Assessment, 17, 19, 34, 47, 65–67, 69, 71, 72, 74, 75, 77–79, 89, 118, 125, 149, 150, 155–157, 159–164, 166, 174, 177, 179, 184, 189, 198, 203, 204, 211, 212, 218, 242
 automated, 162
 board, 229
 centre, 195, 212, 217
 course, 207
 coursework, 75, 78, 156, 174, 210
 criteria, 17, 163, 212
 formative, 76
 group, 78
 group project, 189, 198, 200, 201
 mechanisms, 193
 methodology, 201
 module, 106
 patchwork, 213, 214, 216, 218
 peer, 66, 184, 199, 244
 personal skills, 214
 portfolio, 125, 161
 process, 74, 126, 199, 200, 212, 215
 promote Learning, 210
 reflection on, 214
 regime, 187
 self, 214, 215

skills, 195

task, 71

Assignments, 31, 35, 67, 69, 74, 80, 110, 229

 programming, 162

Attendance, 29, 31, 37–43, 45, 120, 235

B

BCS, 68, 160, 172, 262

Behaviour, 5, 9, 11, 12, 31, 56, 71, 72, 105, 108, 117, 122, 151, 165

Blackboard, 41, 65, 143

Blogs, 66, 267, 270

C

Career, 268

Career development, 264

Challenges for lecturers, 17

Cognitive

 Disconnect, 35

 theory, 51

Cognitive control, 3

Cognitive noise, 51

Collaboration, 33, 74, 83, 136, 180, 181, 183, 243, 266, 267

Competencies, 204–207, 212, 216–218, 256, 257

 student, 207

Conceptual, 8, 87, 102, 119, 120, 124, 126, 135, 137, 142, 172, 209

Confidence, 21, 59, 110, 118, 126, 155, 161, 190, 199, 203, 213–218, 224, 228, 230, 234, 238, 243, 246, 248, 250, 256, 258, 264, 276, 277

Conflict management, 262

Constructivism
 theory, 19
 Constructivist, 209, 213, 217
 theory, 19
 Constructs, 52, 53, 104
 personal, 188
 Contact time, 30, 44, 48
 Coursework, 75, 78, 109, 110, 118, 124,
 153, 154, 231
 portfolio, 125
 COVID, 69, 91, 92, 180
 Creativity, 69, 135, 173, 227, 243, 249, 255
 Critical analysis *see*, Critical thinking
 Critical evaluation, 190
 Critical thinking, 94, 162, 240, 262
 Curriculum, 19, 71, 78, 86, 89, 90, 95, 100,
 161, 174, 183, 187–189, 191, 192,
 201, 203–208, 213, 218, 243, 253,
 260, 264, 276

D

Data science, 85–91, 256
 Digital literacy, 32
 Digital profile, 211
 Distance learning, 63

E

Education
 changes, 17, 160
 Computer Science, 76, 79, 130
 computing, 17
 contextualisation, 85
 culture, 76
 environment, 93
 experience, 18, 78
 formal, 131
 infrastructure, 91
 interdisciplinary, 86
 open, 11
 process, 59
 purpose, 190
 research, 172, 189
 standards, 161
 strategy, 131
 traditional, 11
 E-learning, 69
 Employability, 13, 64, 78, 83, 99, 100, 111,
 187–189, 192, 193, 195, 198, 200,
 201, 203–210, 212–218, 225, 238,
 243, 253–255, 258–264, 275, 277,
 280, 283
 graduate, 225

Employers, 6, 9, 13, 172, 183, 189,
 195–197, 204–208, 210, 212, 213,
 215–218, 223–229, 231, 232,
 234–238, 246, 249, 253–259, 261,
 263, 270, 271, 275–278, 281, 282
 Employment, 67, 69, 195, 205–207, 213,
 215, 217, 224, 235, 236, 238, 239,
 242–245, 247–249, 254, 256, 257,
 265, 271, 275, 278, 280, 283
 attributes, 248
 Engagement, 10, 12, 20, 22, 23, 29, 31, 32,
 34, 36, 37, 39, 41–47, 49, 65, 93, 94,
 115, 129, 139, 140, 146, 147, 156,
 159, 160, 182, 196, 199, 206–210,
 212, 217, 224, 230, 231, 233, 235,
 248, 259, 260, 264, 270, 276
 Evaluation, *see* methodology

F

Facilitated Peer Assessment, 196, 198
 Feedback, 25, 32, 39, 49, 65, 69, 76, 111,
 115, 125, 127, 136, 144, 154, 157,
 159, 161, 162, 165, 166, 171, 172,
 178–180, 182, 195, 198, 204, 210,
 212, 214, 215, 224, 227, 230, 233,
 234, 240, 243, 247, 258, 260, 264,
 266, 269, 280, 283
 assessment, 216
 assignment, 39
 constructive, 234
 employer, 234
 expert, 216
 formative, 65, 144, 230
 individual, 211
 ongoing, 264
 peer, 198
 project, 280
 student, 115, 150, 160, 171, 182, 217,
 244, 248
 student (from), 218, 234
 summative, 32, 161, 212
 verbal, 178
 Flipped Classroom, 29, 30, 32–34, 46, 47
 Fuzzy Logic, 65, 149–151, 154–157

G

Government, 66, 204, 218
 Graduate attributes, 205, 238, 249
 Group process, 187, 192, 199, 243
 Group project, 171, 188–193, 195, 196,
 198, 199

Group work, 18, 171, 178, 179, 181, 182, 230, 243

H

HEFCE, 204

Higher Education Academy (HEA), 17, 161, 196, 206

I

IFT, 4, 5

Industrial *see*, industry

Industry, 31, 68, 72, 77, 78, 88, 89, 91, 162, 172, 183, 190, 212, 223, 226, 227, 229–231, 235, 239, 243–245, 247, 249, 256, 257, 260, 261, 271, 282, 283

Inheritance, 19, 56, 94, 123, 162, 165

Interdisciplinary, 86, 91, 93, 95, 173, 174, 179, 183

Internships, 207, 244, 276, 280

Interviewing *see*, interviews

Interviews, 78, 100, 183, 197, 208, 215, 217, 224, 226, 230, 234, 235, 250, 255, 258, 259, 268, 277

face to face, 208

job, 208, 224, 231, 236, 237, 258

placement, 183

process, 195, 234, 235, 248

ISR, 63

J

Java, 19, 24, 100, 101, 103, 109–112, 159, 162, 163, 166

JISC, 162

K

Knowledge

abstract knowledge, 189

background knowledge, 126

block, 141

computer science, 82

construction, 129, 137, 138

core, 136

cultural, 249

employability, 206

knowledgebase, 149

map, 129, 131, 138–141, 143, 144

programming, 100

specialist, 37

structure, 19, 21

technical, 256

transfer, 59, 183, 226

transferable, 254

Knowledge map, 131

L

Language

body, 212

computer, 102

English, 75

natural, 25, 151

pattern, 115, 116, 125–127

processing, 101–103, 108

Leadership, 196, 238, 247, 248

Learning

action-based learning, 6

active, 7, 17, 19–22, 25–27, 216, 259

barriers, 19, 52, 60, 100, 101, 135, 263, 265

blended, 49, 180

collaborative, 20, 94, 155, 181

distance, 32, 180

distributed, 65

environment, 14, 17, 32, 88, 93

experience, 18, 48, 89–91, 144, 159, 196, 197

face-to-face, 92

independent, 89

integrated, 259, 262

inter-disciplinary, 243

lifelong, 256, 280

machine, 74, 82, 92, 176, 239

material, 27, 29, 32, 35, 41, 45, 47, 49, 66, 161

networked, 66

objective, 56, 94, 95, 101, 102, 110, 111, 150, 154–156

online, 32, 276

outcomes, 19, 22, 59, 89, 93, 143, 182, 187–189, 191, 192, 200, 201

peer assisted, 269

personal experience, 196

problem-based, 51, 89

process, 33, 47, 51, 87, 100, 101, 138, 188, 190

project-based, 89, 136, 181, 259

quality, 156

resources, 33, 34, 36, 46, 72

rote, 59

self-directed, 243

skills, 190

style, 136, 161

surface, 150
 two-way, 260
 work-related, 259, 260, 262
 Lego, 267
 Lego Mindstorms *see*, Lego
 Lego robot *see*, Lego
 LinkedIn, 211, 227, 233, 240, 241, 250,
 261, 269, 271, 275, 279, 281, 282

M
 Marking, 65, 126, 160–162, 164, 165
 MATLAB, 154, 156
 Mentor *see*, mentoring
 Mentoring, 90, 236, 260, 268
 Metaphors, 51, 54, 56–58, 60
 Metrics, 37, 49, 150, 157, 204
 Model
 cognitive, 51, 53, 55, 59
 cognitive engineering, 5
 conceptual, 51, 53–55, 59, 125
 mental, 20, 56
 Moodle, 181
 Motivation, 3, 18, 21, 64, 88, 93, 110, 161,
 207, 230, 247–249
 Multitasking, 3–5, 12

N
 Networking, 29, 67, 68, 134, 205, 224, 247,
 249, 256, 261, 270, 275, 278,
 280–282

O
 Office of Students (OFS), The, 257
 On-line *see*, Online
 Online, 269
 articles, 240, 242
 code, 77
 delivery, 75
 examinations, 74
 feedback, 212
 interviews, 208
 learning, 32, 33
 meetings, 180, 230, 262
 module delivery, 64
 opportunities, 247
 placement opportunities, 231, 236
 plagiarism, 72
 presence, 253, 269, 271, 275–278, 280
 profile, 280, 283
 recruitment, 279
 resources, 72, 143, 232

shopping, 114
 teaching, 92
 teaching and learning, 180
 tools, 33
 websites, 240
 Outcomes
 employment, 280
 graduate, 205, 224, 225
 module, 25
 placement, 226
 process, 264
 project, 144, 162
 project work, 136

P
 Participation, 86, 155, 182, 207
 active, 21
 collective, 155
 stakeholder, 129
 Pedagogy, 150
 ABL, 7
 online, 92
 Pedagogy for Employability, 206
 Pedagogy for Employability Group, 255
 Personal
 appearance, 229
 attributes, 240, 254
 brand, 281
 credentials, 206
 development, 187, 191–193, 207
 experience, 245
 growth, 246
 learning, 216
 profile, 270
 reflection, 48, 75, 191, 197
 skills profile, 240, 263
 soft skills, 240, 242, 243
 Personal Development Planning (PDP),
 207, 263
 Placement, 37, 118, 183, 196, 197, 207,
 214, 215, 223–236, 238–240, 243,
 244, 246–250, 259–261, 268, 269,
 271, 275–277, 279–282
 barriers, 223, 236, 248, 259
 employers, 249
 feedback, 223
 roles, 236
 students, 249
 Plagiarism, 71, 72, 74, 75, 80, 81, 160, 173
 Postgraduate, 65, 115, 212, 275
 Presentations, 257
 poster, 268

student, 66, 76, 245
 Problem-based, 27, 57, 59, 89, 262
 Problem solving, 9, 59, 65, 138, 172, 189,
 190, 240, 254–257
 process, 138
 Professional development, 172, 210, 211,
 213, 214
 Professional standard, 30, 37
 Programming
 language, 19, 51, 55, 56, 80, 94,
 100–102, 109, 111, 116, 154, 159,
 190, 254
 Project work, 18, 67, 89, 136, 171, 173,
 183, 189

Q

Quality Assurance Agency (QAA), 65, 66,
 68, 161, 263

R

Research, 3–5, 8, 11–14, 18, 19, 63, 64,
 67–69, 71–75, 82, 83, 86–88, 91, 92,
 95, 117, 131, 136, 154, 177, 179,
 190, 203, 206–208, 215, 217, 218,
 226, 228–230, 234, 235, 240, 247,
 248, 253, 258, 262–264, 276, 278,
 279
 methodology, 74
 pedagogic, 86
 quantitative, 218
 Responsibility, 68, 71, 73, 195, 198–201,
 209, 224, 225, 236–238, 240, 242,
 245, 246, 248, 249, 255
 Robot, 51, 55, 58, 267

S

Self-awareness, 189, 190, 240, 249, 264
 Self-reflection, 188
 Skills
 cognitive, 9
 communication, 189, 228, 240, 254,
 256–258, 260, 262
 employability, 189
 gap, 257
 hard, 239, 240, 255
 interpersonal, 171, 256, 257
 personal, 256, 257
 problem-solving, 254, 256
 self reliance, 256
social, 245, 249, 254, 256

soft, 93, 172, 174, 228, 239, 240, 242,
 253, 255, 257–259, 261–263, 267,
 271, 282
 work-related, 256
 Social capital, 206, 210, 218, 277
 Social media, 7, 9–11, 31, 80, 91, 211, 270,
 271, 275–281, 283
 footprint, 270
 profile, 270, 275, 277–279
 Social network, 7, 277, 280
 Stakeholders, 122, 129, 131, 134–137, 139,
 147, 225, 227, 231, 236, 243
 sponsors, 122
 Student-centred, 209, 263
 Students
 failing, 99
 international, 18
 Support
 personal, 268

T

Teaching and learning, 4, 72, 174, 180, 216,
 227
 Teaching quality, 17
 Teams, 172, 228, 236, 263, 266
 bias (in), 198
 careers, 276
 communication, 179
 interdisciplinary, 174, 179, 183
 multidisciplinary, 171, 172
 selection, 199, 229
 staff, 187, 264
 student, 228, 262
 teamwork, 93, 145, 172, 174, 179, 181,
 183, 185, 205, 212, 213, 217, 240,
 242, 257, 262
 working, 77, 171, 191, 201, 242, 255,
 256, 267
 Technological Pedagogic Context
 Knowledge (TPCK), 89
 Theory, 88, 189
 and practice, 90
 application, 189
 cognitive, 51, 56
 construct, 51–53, 55, 60
 Constructivist, 19
 development, 46
 foraging, 3
 graph, 21
 group, 199
 link to practice, 214
 momentum, 101

set, [18](#), [19](#), [22](#), [151–153](#)
social independence, [162](#)
threshold concepts, [203](#), [208](#), [209](#)
Tutorial, [22–26](#), [29](#), [31](#), [34–36](#), [41](#), [43–49](#),
[65](#), [181](#), [200](#)

Tutors
personal, [269](#)

U

UK Commission For Employment And
Skills (UKCES), [256](#)

Undergraduate, [159](#)

Unified Modelling Language, [115](#), [117](#),
[119](#), [120](#), [123](#), [163](#)

Universal Design for Learning

UDL, [66](#), [156](#), [161](#)

V

Virtual learning environment, [34](#), [64](#), [143](#),
[233](#)

Virtual meetings, [32](#), [181](#)

VLE *see*, Virtual Learning Environment

W

Webinar *see*, virtual meeting

Work-Based Learning (WBL), [264](#)

World Economic Forum (WEF), [258](#)