George Wells
Monelo Nxozi
Bobby Tait (Eds.)

Communications in Computer and Information Science 1518

ICT Education

49th Annual Conference of the Southern African Computer Lecturers' Association, SACLA 2020
Virtual Event, July 6–9, 2020
Revised Selected Papers

Springer

Communications in Computer and Information Science 1518

More information about this series at https://link.springer.com/bookseries/7899

George Wells · Monelo Nxozi · Bobby Tait (Eds.)

ICT Education

49th Annual Conference of the Southern African Computer Lecturers' Association, SACLA 2020
Virtual Event, July 6–9, 2020
Revised Selected Papers

Springer

Editors
George Wells
Rhodes University
Grahamstown, South Africa

Monelo Nxozi
Rhodes University
Grahamstown, South Africa

Bobby Tait
University of South Africa
Johannesburg, South Africa

ISSN 1865-0929 ISSN 1865-0937 (electronic)
Communications in Computer and Information Science
ISBN 978-3-030-92857-5 ISBN 978-3-030-92858-2 (eBook)
https://doi.org/10.1007/978-3-030-92858-2

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

Preface

SACLA 2020, the 49th Annual Conference of the Southern African Computer Lecturers' Association, provided a forum for the presentation and discussion of original research and practical experiences in the teaching and learning of Computer Science, Information Systems, Informatics, and other related disciplines. The conference theme selected for 2020 was *TL;DR: Teaching the New Generation!* Due to the restrictions and safety concerns introduced by the COVID-19 pandemic, the conference was held online, as a Zoom Webinar, during July 6–9, 2020.

Submissions included full research papers, short papers, and discussion papers. In addition, the conference also featured two keynote addresses (by Helen Purchase from the University of Glasgow and Charles Young of Rhodes University) and two presentations by sponsors (IITPSA and Pearson). The focus of the primary track of the conference was on practical experiences in computing education at the tertiary/post-school level, particularly classroom innovations, novel tools for learning and/or assessment, and research investigating aspects of computing education. In addition, submissions were also invited for a secondary track focusing on general research in Computer Science and Information Systems, which was intended to assist early-career academics with a welcoming platform to present their research.

All papers, for both tracks, went through the same double-blind reviewing process, and each paper received three reviews. The Program Committee comprised 33 international and national researchers (see Organization).We wish to record our thanks to the members of the Program Committee who willingly gave their time and expertise to provide very helpful and insightful reviews of the submissions.

A total of 53 papers were submitted. After reviewing, 20 papers were accepted for presentation at the conference as full research papers. After the conference, and following a careful review of the full papers in the main track, the 13 best papers were selected for publication in this Springer volume (giving an acceptance rate of 24.5%). Prior to submission to Springer, these papers were reworked by the authors, taking into account the reviewers' comments and feedback from delegates at the conference. The remainder of the full papers presented at the conference appear in a separate locally published proceedings of the conference.

This year, the award for the best paper was awarded to Kudzai Katsidzira and Lisa Seymour for their paper entitled "Factors Impacting Using the Internet for Learning: The Digital Divide in South African Secondary Schools". Congratulations to the authors on research to be proud of!

Our thanks go to all the authors and researchers who entrusted us with the results of their research and writing. Any conference is only as good as the work that is submitted for presentation, and we are deeply grateful to all who contributed, under the difficult circumstances of a global pandemic and national lockdown, to make SACLA 2020 a great success.

November 2020

George Wells
Monelo Nxozi
Bobby Tait

Message from the Chairpersons

SACLA 2020 had originally been planned to take place during July 6–8, 2020, at the Mpekweni Beach Resort in Eastern Cape. We were quite far into our preparations with the venue, when the COVID-19 pandemic struck South Africa. By early April it was becoming apparent that holding an in-person conference might not be possible, although at that stage none of us had any idea of the disastrous effect the virus would have on our country and its people. On April 18, 2020, before participant registration was opened, the organizing committee made the final decision to hold a virtual conference instead. This was both a relief having actually made the decision, but also the start of a fairly stressful period in the sense that we had to restart the planning for a virtual conference, and familiarise ourselves rapidly with technology to support the decision.

We were most grateful to our three sponsors, the Institute of Information Technology Professionals, Rhodes University, and Pearson South Africa, who continued to support our virtual endeavour.

Given the worsening circumstances in our country, going virtual was obviously the right decision. We sincerely hope that all the participants felt that it was a worthwhile alternative, although we realize that the socializing and networking aspects of a face-to-face conference were sorely missed. We are, however, confident that the dissemination of some excellent research was achieved through the online sessions.

Having to deal with organizing a virtual conference certainly had its challenges. However, there were always solutions on offer, especially related to technical logistics, which was crucial to the success of the conference. We wish to thank the following committee members for their active involvement throughout:

Program chairs:	George Wells and Monelo Nxozi
Technical logistics:	Billy Morgan
Sponsorship:	Stephen Flowerday
Webmaster:	Greg Foster
Proceedings:	Philip Machanick and Karl van der Schyff
Treasurer:	Caro Watkins
Secretary:	Michelle Coupe
Keynotes:	James Connan and Dane Brown
Springer volume:	Bobby Tait

Thanks are also due to the keynote speakers, all the participants who joined the various sessions, and of course the presenters without whom the conference could not have happened. In terms of engagement at the conference, 110 participants registered, with the maximum attendance at any one session being just over 40, and the average attendance being in the mid 30s. We look forward to hopefully meeting in person once again at SACLA 2021, which will be hosted by the University of Johannesburg.

November 2020

Karen Bradshaw
Ingrid Siebörger

Organization

Conference Chairs

Karen Bradshaw	Rhodes University, South Africa
Ingrid Siebörger	Rhodes University, South Africa

Program Committee Chairs

George Wells	Rhodes University, South Africa
Monelo Nxozi	Rhodes University, South Africa

Springer CCIS Post-proceedings Co-editor

Bobby Tait	UNISA, South Africa

Proceedings Editors

Philip Machanick	Rhodes University, South Africa
Karl van der Schyff	Rhodes University, South Africa

SACLA 2020 Program Committee

Shirley Atkinson	University of Plymouth, UK
Shaun Bangay	Deakin University, Australia
Sonia Berman	University of Cape Town, South Africa
Alan Chalmers	University of Warwick, UK
Charmain Cilliers	Nelson Mandela University, South Africa
Thomas Clemen	Hamburg University of Applied Sciences, Germany
Sue Conger	University of Dallas, USA
Stephen Flowerday	Rhodes University, South Africa
Patricia Gouws	UNISA, South Africa
Jean Greyling	Nelson Mandela University, South Africa
Barry Irwin	Noroff University College, Norway
Nega Lakew	Bowie State University, USA
Janet Liebenberg	North-West University, South Africa
Hugo Lotriet	UNISA, South Africa
Anthony Maeder	Flinders University, Australia

Jan Mentz	UNISA, South Africa
Francois Mouton	Noroff University College, Norway
Liezel Nel	University of the Freestate, South Africa
Christian Omlin	University of Agder, Norway
Jacques Ophoff	Abertay University, UK
Helen Purchase	University of Glasgow, UK
Rayne Reid	Noroff University College, Norway
Karen Renaud	Abertay University, UK
Aslam Safla	University of Cape Town, South Africa
Ian Sanders	UNISA, South Africa
Chris Scogings	Massey University, New Zealand
Hussein Suleman	University of Cape Town, South Africa
Alfredo Terzoli	Rhodes University, South Africa
Hannah Thinyane	United Nations University, China
Mamello Thinyane	United Nations University, China
Mac van der Merwe	UNISA, South Africa
Dieter Vogts	Nelson Mandela University, South Africa
Janet Wesson	Nelson Mandela University, South Africa

Additional Reviewers

James Connan
Lynette Barnard
Sune Van der Linde

Contents

Factors Impacting Using the Internet for Learning: The Digital Divide in South African Secondary Schools

Kudzai Katsidzira and Lisa F. Seymour(✉)

University of Cape Town, Cape Town, WC, South Africa
lisa.seymour@uct.ac.za

Abstract. Globally governments are providing the Internet to schools to improve the quality of teaching and learning. In South Africa, the Western Cape Education Department recently embarked on an e-learning smart schools project to provide broadband to schools in the province. Yet this project has had challenges and not all schools have embraced e-learning. Through a case study of eight schools, this research explains how relevant factors impact the use of the Internet for teaching and learning in Cape Town schools. The inductively derived model explains how contextual conditions such as the high levels of inequality in South Africa impact usage of the Internet for learning. The actual characteristics of learners and teachers, the facilities being provided by the WCED and the relevant schools and the practices of teachers also have an impact. This study should assist local governmental organisations, schools and teachers trying to increase and reduce inequalities in e-learning in schools. The study should also interest university lecturers who need to be aware of the changing teaching practices in schools and the expectations of students arriving on campus.

Keywords: e-learning · Digital divide · ICT in teaching

1 Introduction

As Information Communication Technologies (ICTs) develop rapidly, we are experiencing what has been termed as the "digital revolution". New technologies and information are emerging to enhance our everyday lives, yet people still face barriers to participate in this revolution. Social inequalities arise as the digital revolution progresses, due to everyone not being equally empowered, thus increasing the digital divide [1]. The inequalities that arise affect different areas of life; one of them being education. The Internet has vast capabilities that can empower individuals in different ways. Implementing broadband internet has been supported by global organisations such as the United Nations Educational, Scientific and Cultural Organisation as well as the International Telecommunications Union which came together to form the Broadband Commission for Sustainable Development [2]. This organisation was formed to offer

G. Wells et al. (Eds.): SACLA 2020, CCIS 1518, pp. 1–18, 2021.
https://doi.org/10.1007/978-3-030-92858-2_1

insights on providing broadband globally to help reach the United Nation's Sustainable Development Goals. While it is believed that broadband can aid in digitally transforming education, the effects of broadband once implemented, are not fully understood [3,4]. Lack of understanding does not only affect current and future policies towards broadband provision and implementation but how teaching takes place, as well as the digital divide. Bill Gates recently stated "The value of computers in classrooms is virtually nil. So that's good news, we didn't accentuate the digital divide; the schools with computers are just as bad as the ones without them" [5], which raises the question of What influences successful use of technology for teaching? The role of teachers surely is critical as they are aware of learners, particularly how they learn, remember and process information [6]. Yet there must also be other factors. Hence this research will explain how relevant factors impact the use of the Internet for teaching and learning in a school environment.

South Africa continues to have one of the most unequal distributions of income and wealth in the world [7]. This inequality has been attributed to the disparate quality of school education in the country [7]. The quality of schooling for most black learners is poor [8]. Historically, under apartheid, South Africa had three school types, Model A private schools, Model B state schools, and Model C, state-aided schools [9]. These inequalities have persisted with the public school system being divided into a smaller group of well performing (state-aided or fee paying) schools serving the wealthiest 20–25% of learners, while the majority of schools are fee-free, poorly resourced and cater for the poorer learners [8,9].

To try to bridge the digital divide, many governments have supported or subsidized the implementation of broadband, particularly in schools [1,10]. In 2015, the Western Cape Education Department (WCED) announced their plans for the e-learning smart schools project with some of the following goals: The linking of schools through a high-speed, real-time Wide Area Network (WAN) and the provision of Local Area Networks (LANs) in different schools throughout the Western Cape [11]. This research seeks to answer the question: How has the use of the Internet for learning been influenced in secondary schools in Cape Town, South Africa (SA)? Cape Town is an ideal region to select schools from as it is the region with one of the highest uses of broadcast spectrum in South Africa [12] and it is part of the WCED e-learning project. The following section reviews the literature on this topic in more detail. This is followed by the research method, the findings and lastly the conclusion.

2 Literature Review

In trying to understand the factors impacting Internet usage for learning in Cape Town secondary schools, this literature review will first look at internet in developing regions and will then cover government support of the Internet. It will then describe and explore factors identified in the literature that impact Internet access and usage by teachers and learners for education.

In developed regions such as North America and parts of Europe internet access is easily accessible [13] with access in public areas. Yet accessing the Internet in less developed countries proves to be harder [1]. Developing countries in regions such as Africa, Asia, and Latin America have poor infrastructure supporting Internet access [13,14]. This leads to poor Internet speed which affects the reliability of the Internet and hinders individuals and schools from using it effectively. Some areas have no internet access at all. Where connections are available people may not be able to afford to pay for internet subscriptions [13]. A study done in Nigeria identified that telecommunication companies in Africa have been trying to increase Internet penetration in Africa [15]. Unfortunately, as most of the population in Africa resides in rural areas this is challenging and impacts the schools in those areas greatly.

2.1 Government Support of the Internet

Many countries have created broadband policies as ICT is seen as a key driver for social and economic development [16]. Much of the support comes from the presumed relationship between broadband and economic growth [10,17]. The Broad-band Commission for Sustainable Development has set targets for 2025; one of them being "all countries should have a funded national broadband plan or strategy or include broadband in their universal access and services definition" [2]. Various countries who are a part of the UN have adopted broadband policies over the last 15 years [2,18]. There is an expectation that the use of ICTs and access to broadband will have a positive effect on pedagogic outcomes [4,18]. Consequently connecting schools to the Internet has become a top priority of many governments [4,10,19] and in some cases, such as in Portugal, broadband implementation is the sole decision of the central government in which schools do not have a say on whether they would like the broadband or not [3].

The broadband adoption initiative is being supported by some African countries such as Tanzania and Nigeria [14,15]. In South Africa, the government supports broadband adoption and policies due to the potential in skill development and job growth [12]. The WCED e-learning project seeks to improve the quality of teaching and learning in the province [11] through implementing broadband in schools. As of 6 August 2019, 1273 schools were connected to the WAN [20]. In addition, another goal was to provide teacher training and development in ICT and the use of e-learning in schools [20].

In North America; one of the reasons for supporting broadband adoption is to try address the digital divide and have everyone involved in the digital revolution [21]. In developing countries, broadband has also been promoted to reduce the digital divide, such as in Nepal where it is used to distribute free and open digital educational con-tent [22]. Other reasons for support are the benefits that can arise from broadband implementation such as expanding access to education [4] connecting people, particularly in developing countries and increasing digital skills [2]. Studies in Portugal and North America have been done to access whether broadband implementation in schools can spill-over into households [19,23,24]. The research did show that inter-net access in schools encouraged Internet adoption in households.

2.2 The Usage of the Internet by Teachers and Learners

Some countries such as China and South Africa are moving towards developing and implementing smart classrooms [11,25] in which e-learning technologies are used by teachers to enhance learner's learning experiences and support their learning activities [25,26]. In the developmental psychology of children, it is agreed that the development of cognitive behaviour in learners is achieved by teachers [6]. Teachers are a primary resource to schools and teachers may not be too sure how to adapt to the new technologies and therefore they need support to avoid further inequalities [27]. Teachers who are not trained or lack understanding in new technologies tend to not use ICTs in their teaching practices [22,26,27]. This differs from teachers who have been exposed, trained and prepared to use ICTs [25].

Learners can learn in different ways and this is influenced by teacher's teaching methods and incorporation of learning theories. Some of these methods can be traditional or instructivist, or modern or constructivist [28]. Traditional methods focus on individuals focusing on their own work and learning through repetition or memorisation. In contrast, modern methods focus more on teamwork and learner's participation in discussions and presentations [29]. The methodology teachers use influences their attitude towards technology and to internet usage in classrooms [26,28]. Teachers are also less likely to incorporate ICTs in the classroom if their learners use it for off-task activities such as entertainment [3]. It has been noted that using technology for teaching content distribution and creation does not necessarily translate into better learning experiences although more research needs to be conducted [28,30]. Many studies have been done to look at the impact on learner performance but here it has also been acknowledged that more research is needed [3,4]. Broadband use can have negative effects on the classroom environment and learners [3]. Learners may become more distracted and use the Internet for entertainment or leisurely purposes if not monitored correctly. Schools can implement policies to monitor the use of the Internet such as blockage of websites [3].

Internet use has been found to differ depending on which school you attend [24]. One study found that learners from better resourced schools, with teachers that had higher technological training, preferred to use the Internet for entertainment purposes in comparison to their counterparts [24]. On the other hand, a study done in Turkey on the impact of access to ICTs for learners found that learners generally tend to not use the Internet for academic purposes [31]. Literature highlights that broadband in households and socio-economic status impacts how learners use the Internet [23,24,31] and the SA study by Sackstein et al. [28] found that learner's use of technology depends on the age of the learners. Positive effects in the teaching space are only seen when ICT is used to support other teaching practices [27]. ICTs in the learning space have the potential to enhance the learning experience through teachers using immersive teaching practices they have never used before [25] but they do require the Internet.

While internet usage in schools is seen to be a cost saver as it can reduce printing costs [28], resources available in schools is a factor impacting the use of Internet in schools. Studies have found strong correlations between having relevant devices and classroom internet use [18]. Historically and still in many schools, Internet access to learners within schools is limited to computer laboratories [32]. A study done in North America, on the barriers of integrating computing in the classroom [26] found that schools with minority and poorer learners, tend to not be able to provide wireless Internet access for the whole school and computer devices also tend to be scarce. Poorer youth are less likely to have access to quality Internet resources and this impacts how they use the Internet [24]. In many African countries, Internet access is hindered due to factors such as inadequate power supply, low computer literacy rates, lack of relevant teaching materials, high cost of computers and other technologies, and low incomes [33]. Additionally, for lower-resourced schools, it has been found that learners are only able to access the Internet at school [14] and not at home. Yet mobile technology and other technologies which can be incorporated in the learning space have increased in South Africa and globally [28]. In South Africa, only 1% of university learners do not own a mobile phone [34]. In Table 1 we summarise the main factors reported in the literature reviewed influencing the use of the internet for learning in schools.

Table 1. Literature factors influencing internet usage in schools.

Factors influencing internet usage in schools	References
Limiting internet access to computer laboratories	[32]
Quality of ICT Infrastructure, power supply and relevant other technologies	[24, 26, 32]
Teacher's methodologies and style of teaching	[26, 28]
Learner's use of the internet for entertainment	[3]
The age of learners	[28]
Printing cost reduction	[28]
Teacher computer literacy, availability of relevant teaching materials, training of teachers in use of ICTs	[22, 25–27]
Broadband in households and socio-economic status of learners	[23, 24, 31]
Availability and cost of devices	[18, 24, 26]

3 Method

This section describes how the research was performed, assumptions made and limitations. The philosophy of the research influenced the assumptions made about the research which reinforced the methods and strategy chosen [35]. Seeing as the research is explanatory, the research on the proposed topic was centred on understanding teacher's experience around the use of broadband in the teaching space. The research attempted to describe how teachers have been impacted as broadband is introduced. The philosophy is interpretive which focuses on understanding the role of humans as social actors and understanding the point of view of those being researched [35]. A multiple case study strategy was conducted with each school representing a case [36].

The researcher obtained permission from the WCED and the university's ethics approval before contacting any schools. Participant's contributions to the research were voluntary. The research was done over a few months in 2019. Eight schools were chosen across Cape Town from all three school types. The first character of the case represents the school type. There was one private school (A1), four state-aided schools (C2–5) and three fee-free schools (B6–8). Judgment sampling, a qualitative sampling technique [37] was used to select diverse teachers to interview at each school. A researcher travelled to each school to interview participants. The participants provided the researcher with necessary documentation such as relevant policies. Semi-structured interviews were performed following a loose interview protocol and interviews were recorded and later transcribed using Otter Voice notes. Table 2 lists the participants interviewed. Participant codes start with the case number and end with A if the teacher had 10 or less years teaching experience and B if they had more. In each school where multiple interviews were held, a younger and older teacher were interviewed. Four out the twelve participants had worked internationally and four had more than 10 years teaching experience. NVivo was the tool used to code and analyse the data. To identify the themes, Braun and Clarke's [38] six steps of thematic analysis was followed:

1. Become familiar with the data. The researchers made summary notes on each interview conducted and the school policies collected.
2. Generate initial codes. The coding phase was done in parts. The researchers sought common topics that arose and grouped them together. From that initial coding the coded data was then reanalysed to look for more codes.
3. Search for themes. Patterns in data that were relevant to the research question were searched for in the initial codes.
4. Review themes. Themes were reviewed to ensure that they were relevant to the re-search and to establish sub themes.
5. Define themes. The essence of each theme was established to give a proper definition and explanation of the theme.
6. Write up. This involved the writing up of the data.

Table 2. Participants interviewed.

Code	Subjects taught and other roles	Teaching years
A1A	Teach English Language, Ed Tech Coach	6
A1B	Teach Creative arts, Head of Culture	15
C2A	Teach Afrikaans, IT, Head Technology Integration	9
C2B	Teach Mathematics, Science, Geography, Head of Examinations	37
C3A	Teach Computer Literacy, Life Orientation	12
C4A	Teach Computer Literacy, Life Orientation, Computer Application Technology	4
C4B	Teach English Language, Deputy Principal	29
C5A	Teach English Language, Head English, Grade 11 Discipline, Interact, Social Media	6
B6A	Teach Economics, Accounting, Business Studies	3
B6B	Teach English Literature, Life Orientation, Creative Arts, Deputy Principal	23
B7A	Teach Creative Arts, Music, English Language, Grade Head, ICTS Coordinator Public Relations officer	10
B8A	Teach Science, Geography, Business Studies	4

4 Findings

The explanatory model derived from the interviews is shown in Fig. 1. The factors found to impact the use of the Internet for learning will now be described. While the motivation for introducing broadband into the Western Cape schools was partially to reduce the digital divide, our analysis found high levels of inequality in terms of Internet use when contrasting the different school types.

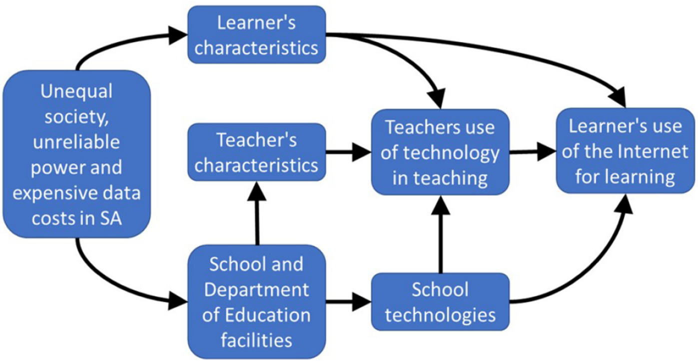


Fig. 1. Factors impacting learner's use of the Internet for learning.

Three contextual factors were mentioned which impact learners and school facilities. Firstly, participants acknowledge the problem of very high costs of data in South Africa: "Our problem in our country is the cost of the data" (P03B). As a result, learners "don't have data - they're limited" (A1A). A second issue highlighted is the disparity between households in South Africa where learners have unequal access to devices and internet at home. "A household that has a steady income where they actually have internet and computers versus a child who doesn't. I think a child who does benefits the most" (C4A). Finally, the unreliable power utility was referred to: "So there are challenges or something like Eskom [the unreliable power utility] you need to know that. Chances are that you won't have electricity and no internet" (C4B).

4.1 School and Department of Education Facilities

The education department had started to provide facilities such as internet access, training courses and online content. Many schools, particularly state-aided schools, had found these facilities to be inadequate and these schools had funded extra facilities. Private schools appeared to have been providing appropriate facilities for a while. As school fees are being used to improve facilities, fee-free schools with poorer learners and parents have less facilities. There was evidence of clear inequality in facilities between the three school types. Data evidence for these themes is provided in Table 3 and is now described. Firstly, there were found to be disparities in terms of the quality of internet at the different schools. While the private school noted that learners had Internet access over the last 10 years, generally state-aided schools were using their own internet and the WCED internet, while some fee-free schools were still installing the internet or struggling with its capacity. In many fee-paying schools Internet access was restricted to teachers only while the private school provided wi-fi to visitors.

In terms of DoE Courses and Online Content, the Department of Education has started to provide the necessary courses and online content for teachers, but this was critiqued by some as being too little too late as private schools had been setting trends in terms of online content for a while.

Private and state-aided schools had been investing in their own technology support and training. Many of these schools had a dedicated post responsible for integrating technology into teaching and training teachers. In contrast one fee-free school had not yet opened their computer lab as they had no one to manage the lab and staff had not been trained on using technology.

In terms of managing internet usage, clear policies on Internet use are needed and there is a need for management to be driving technology use. Many of the schools had set policies that prevented learners from using their phones at school and where internet was being used for teaching it was being driven from a senior level. In some fee-free schools there was lack of understanding on what rules needed to be put in place.

Table 3. Data evidence for school and department of education facilities and inequality in school technologies.

Facility themes	Data evidence
Internet Provided by the DoE and/or School	It's good and bad. We've got two lines; you've got the one that is provided by the department of education and then the one that school has always had (C3A). The WCED Wi-Fi, which is also slow sometimes so it's not the greatest quality internet (B6A). The entire school has access to Wi-Fi. How-ever, at this point, we have not formally introduced it to the learners (B8A)
DoE Courses and Online Content	I just think maybe the Department of Education says that they're moving, they're not really moving. All the schools who are massive trendsetters in this area are doing it all privately, you know (A1A)
Technology Support and Training in Schools	I basically spent the majority of my day, trying to come up with ways in which we can integrate technology into the curriculum (A1A). I'm head of technology integration, which means I'm responsible for training and sup-porting the staff in terms of how they use technology to enhance the learning (C2A). We are waiting to employ someone who will take over the entire lab. So, we don't want to open the venue until we are ourselves have obtained training to productively use that venue (B8A)
School's Management of Technology Use	The school's policy is that they can bring cell phones, but it must be off and, in their bag (C4A). So, you've got to have somebody who's going to be driving the use of technology. Now this new principal we've got he has decided everybody's going for the iPad (C2B)
Technology themes	Data evidence
Quality of Internet access and infrastructure	It's 10 gigabit fibre across the site, which is pretty nice. But we've got a heat map done... 10 days ago, as well as an extensive site survey just to identify areas of sort of lower coverage and interference (C2A). This is often an issue with the internet. It is quite slow (B6B). I find it unreliable (B6A)
Tablets and smartboards to teach using the internet	We've [teachers] also got smartboards, and now this year, we've all got tablets (C2B). There are four or five rooms with smart boards? but I think only two of us actually use them (B7A)
Devices for learners to access the internet	Our grade eights all have tablets (C2B). The only access they have is to the computer rooms which creates issues because if a whole school needs to use two rooms the only times, they can really use it is during break (B6B). We currently have a lab with 25 operating computers... Anything that they have on them of value makes them a target. So even the phones already dangerous enough so we never allow the laptop or anything bigger than a Phone (B8A)
Technology to control internet usage	We've got various blockers, and firewalls and all of that to try and keep the children safe (A1B). You can't monitor their phones; you can't have their phones on net support (C3A). With iPads It's easier because we run software that can actually monitor things and we can block certain websites more easily (C2A). I can go onto the classroom and I can then see what's on every screen. And I can lock their screens (C2B). But the problem is about control. And you know, being able to put things in place that learners won't, you know, do the wrong thing (B6B). We must have a set list of websites to consult; at the moment it's a free for all (B8A)
School e-mail and e-learning systems	As a Google school, we make use of the whole Google suites for education, Google Classroom, Google Forms, Google Docs, and it's all online. We also use other things like Flipgrid and Socrative (A1A). And we use a lot of Google classroom as well, which is an online portal, where kids can hand in tasks (C3A)

4.2 School Technologies

The digital divide is clear in South African schools as some schools have the necessary technologies needed for the use of the Internet for teaching and for learning and other schools don't have them or have challenges with them. The unequal distribution of these technologies is driven by unequal school facilities. These technologies are listed in Table 3 with supporting quotes and are now described.

The responses indicate quality of the internet and associated infrastructure varies between schools and participants have different experiences regarding Internet usage. The fee-free schools were complaining about unreliable and slow internet. Even the private and state-aided schools had their challenges but they had skilled staff who were monitoring the situation and adding network points when necessary this helped improve the internet.

Having the Internet is not sufficient. Teachers need to have technologies to teach with. Many of the schools had issued teachers with smart boards and tablets to use for teaching. In some schools the technologies were issued but they were not being used. This lack of use was driven by insufficient facilities to teach and support the teachers. The third category is devices for learners to access the internet. In poorer schools the only devices learners had access to were computers in restricted computer labs and these computers were often insufficient. In wealthier schools, learners were issued with tablets for use in the classroom. While most learners have phones, these were deemed to be insufficient. One respondent referred to crime at the school and the surrounding areas which prevents learners from having devices on them.

This leads to the category of using technology to control internet usage. Software was also being used to control what learners could do on their tablets. Yet this soft-ware could not be used on phones. This was another reason given for tablets being preferred to phones. The private and state-aided schools monitored internet usage and had blockers and firewalls installed that controlled what learners could access on the Internet. In contrast one respondent in a fee-free school noted that learners could access any content. Finally, once all these school technologies are in place, learners need access to e-mail, appropriate e-learning software and learning management systems. The private and some state-aided schools referred to the use of school e-mail accounts and software such as Google Classroom.

4.3 Teacher's Characteristics

Certain teacher's characteristics were found to impact how they use the Internet in teaching, and these are described in this section. In some cases, these teacher characteristics were influenced by the school and Department of Education facilities. These characteristics are listed in Table 4 and are now described.

Teacher's lack of computer literacy was shown to influence their use of the Internet for teaching. Respondents noted that teachers do not use the Internet in teaching because they don't have the necessary computer literacy and don't want to be embarrassed in class. Teachers want to appear to be “the smartest in the room”. Workshops and training on the use of ICT in teaching was found to change teacher’s computer literacy and their enthusiasm for using IT in teaching.

The age of teachers was referred to by many respondents. Some respondents felt that older teachers generally didn’t allow devices in their classrooms and that younger teachers were more familiar with the relevant technologies and therefore used them more easily. An older respondent confirmed how technology was a high source of frustration. However, one respondent commented that the older teachers, because of their teaching experience, ultimately use the relevant technologies to better effect. The subjects teachers teach also appeared to affect use of technology for teaching. In one case a respondent noted that language teachers don’t use the Intent in teaching, while another respondent noted how online resources were “fantastic” for teaching languages.

Teaching style appeared to have a very strong impact on using the Internet in teaching. When a teacher has a modern constructivist teaching style, they believe in inquiry based, learner led teaching then they see their role as merely a facilitator of learning. Their teaching style will then encourage group work, peer assessment, online participation and learners preparing lessons. The Internet is seen as critical for this teaching style. However, they noted that this style empowers learners and not all teachers felt comfortable with losing power in the classroom.

4.4 Learner’s Characteristics

The teachers noted certain learner characteristics which they believed were influencing how they use the Internet in teaching and how learner’s use the Internet for learning. These characteristics are listed in Table 4 and are now described. Teachers noted that for learners to benefit from Internet usage they needed to have their own devices and have internet access at home. Learners who had that were at an advantage. In certain cases, the use of e-learning systems was not embraced because learners did not have that access. Teachers who were embracing the use of the Internet in their teaching referred to learners who are attracted to using the Internet. They noted that learners were “stimulated just by its presence” Learners were said to prefer visual and auditory stimuli over reading, Of concern was the problem of controlling technology usage. Many of the teachers noted that learners were not able to control their technology usage. They found social media, videos and connecting via the Internet hard to resist. Teachers also noted that learners don’t take notes anymore and preferred typing to writing. Instead of learners taking notes they take photographs of the board with their phones.

Table 4. Teacher's and learner's characteristics.

Teacher's characteristics	Data evidence
Teacher's computer literacy and attitude towards IT	Teachers really don't like to be embarrassed... often teachers like to appear the smartest in the room (A1A). Some teachers are scared to use the internet because or because they, they're not very skilled (C4B)
Age of teachers	Our older teachers also tend to do amazing things. So, they are definitely more tentative in the beginning. But then with the years of experience, they're able to use the technology often very thoroughly, to inspire deep learning. For the young people it's definitely easier and but the result aren't always better (A1B). Technology frustrates me. And if it doesn't work immediately, I'd rather not use it. But you know... you're talking to an older person (C2B)
Style or methods of teaching	So, creating lessons where its more inquiry based, student led (C2A). Kids need to be engaged... and ICT is perfect for them. Because you can even the quiet ones, you can get involved (C4B). I quite enjoy having them prepare a lesson... but it [the Internet] has definitely impacted my style (C5A). We're in the most exciting place in education, because suddenly, you've just become a facilitator for them, having information from the best minds in the world. And I think that's why teachers are scared, because suddenly the technology is disruptive and it empowers children and so teachers have to completely change their mindset. And that's difficult to do (A1B)
Learner's characteristics	**Data evidence**
Learners owning devices and having internet access	We're well resourced, the parents have provided iPads for the kids (C2B). I've been trying for a long time to get on to Edmodo or to Google Classroom. So, problem we've had is that our children don't have internet access at home. (B7A). A lot of them don't have computers at home (B6A)
Learners are very visual and attracted to the internet	They're more likely to go and look at that Instagram page than they are to go and read a book about this person (C2A). Our kids want something visual; they want something auditory, and obviously something that they can refer back to. Like I said, it's [the internet] just a part of their lives. So, they immediately connect to it. They are stimulated just by its presence (C5A)
Learners can't control their technology usage	The problem is that they don't know how to control the usage. So instead of doing what they supposed to be doing they end up talking to the friends on WhatsApp (B6B). It's just a very distracting thing [referring to phone] for them to have (C2B)
Learner's don't take notes any more	I try to encourage them to take notes as I speak, even though I find that that's a skill that this generation really, really lacks (C5A). I'll put notes up on the board. And they will instead of actually just writing it down… they'll come forward, take a picture of it with a cell phone and go back (B7A)

4.5 Teachers Use of the Internet for Teaching

How teachers use the Internet for teaching was found to be influenced by the characteristics of learners, the teacher's own characteristics and the school's internet technologies. Many of these internet practices, listed in Table 5, were found not to be performed in certain schools and by certain teachers.

In all schools some teachers were using the Internet to prepare teaching content that learners can relate to and hence would hold their attention and hence they would learn. Examples cited include creating Facebook profiles for literature characters; using short videos in class to wake learners up; incorporating graphics in PowerPoint; including snippets of breaking news in economics; playing YouTube clips of poems to make "it stick"; and playing YouTube music as a reward. Teachers were also using the Internet to access and complete online courses to refresh their teaching practice. Free courses and Google certification were referred to.

In schools with e-learning systems, teachers were sending out all announcements to learners and parents via the Internet which helped communication. They were also distributing all learning content such as notes, memos, previous tests, videos in this way. In some of these schools, teachers noted that they no longer hand out paper. Teachers were also sending information, such as test information, to learners via WhatsApp on the weekend but if a learner didn't have a smart phone this would disadvantage them. In a fee-free school a teacher noted that content was sometimes distributed via flash drives but this did not help learners who did not have access to computers or e-mail.

The practice of allowing devices connected to the Internet in the classroom was referred to by most respondents. Schools in which learners had tablets allowed the de-vices in the classroom and were able to control usage through the relevant software. Most schools had a strict "no phones in the classroom policy" because of the inability to monitor what students are doing on these devices. However, in many cases teachers were breaking with the school policy and allowing phones to be used by learners but they then had to physically control and monitor the usage which is challenging.

In classrooms where devices were being used, teachers were designing class lessons where learners would use the Internet. Examples include assignments where learners search the Internet for answers; where teams collaborate and access and edit shared documents; online quizzes where the live results are shown on the projector and even team quizzes. These practices are driven by the teacher's style of promoting student led teaching and group work.

Table 5. Teachers and learner's use of the internet for teaching.

Teachers use	Data evidence
Preparing teaching content to interest learners	When I teach literature... we do is, like a new profile Facebook profile for the characters so that it's relatable, something they enjoy, so they get to learn about the character (B6B) When I'm teaching a poem I'll try YouTube it and find a recording of it, or something different that just makes it stick (B7A)
Refreshing practice through online courses	I've done through a company called Cloud Education. I've done training and becoming a Google certified teacher (A1B). But the nice thing is, you can just go and. . . they've got lots of free courses there that you can do (C3A)
Communicating electronically with learners and parents	We run as a Google school. . . all announcements and communications and things are sent out electronically (C5A) It's easy to send it out in a WhatsApp group over the weekend... And then I can't do it because I can't give even 26 of the 27 learners information and not the one person (C4A)
Distributing teaching content to learners	They've got a classroom drive, where they can access all my materials, additional notes and videos (C5A). So, I don't hand out paper (C2A). I do give it to them on a flash, but I find it's not helpful because a lot of them don't have computers at home to use that or email (B6A)
Allowing devices or the Internet in the classroom	The rules say no cell phones in the classroom. However, we turn a blind eye if teachers use it for learning (C4B). We see having cell phones on you during class time as a serious misdemeanour... it's very difficult to monitor, whether they're using a cell phone for research... I do think it would be good for them to have access to electronic resources within the classroom (C5A)
Designing lessons with activities, group work and assessment using the Internet	In some lessons, they collaborate on... a Google doc or a Pages document or whatever that they all have shared access to (C2A). I will do like online quiz-zes... And I can see they can see who's done (C3A). It's like a quiz that you set up. . . what they do is kids share phones, kids who have data... they will share a phone and play in teams (C4B)
Learner's use	**Data evidence**
To find information	They will make discoveries on their own (B8A). If a child doesn't know the meaning of a word, or the spelling of a word (B7A)
To access class content and catch up work	They upload TUT videos, how they explain the TUT everything in that, that helps a lot, when my learners are not like, up to date with the tut. Like, for instance, they're absent for a whole week, they miss a lot of work (C4A)
To access text-books and online teaching material	I find it quite disturbing is we are still talking about textbooks... we should have moved on - you just need a device (A1B). [iPads] lightens the bags... you can have maybe one book in there and an iPad (C2A). We encourage kids to use online platforms, for example, the WCED has a very nice portal where they put up a lot of resources for the kids... also Mindset Learn (C3A)
To submit or perform assessments	We use Google Docs throughout the school to do our digital sort of assess-ments and assignments and homework and things handed in (C2A). Kids can submit their work online (C3A)
To communicate, interact and relate to others	We've just recently done a project with learners from India, Sri Lanka, and a country called Bhutan, and South Africa. So, it was a digital platform to let the kids, had to use, and then they could communicate with learners from those countries (B6B)
To fit into a digital world	We live in a digital world. These kids are not equal. Now, they're not going to be able to fit into the world once they leave here (B6B). Digital literacy and being able to like be citizens of the 21st century (C2A). The fourth industrial revolution, you know, and equipping them with skills of collaboration, problem solving, creative thinking, all of that (A1B)

4.6 Learner's Use of the Internet for Learning

How learners use the Internet for teaching is influenced by the teacher's use of technology for teaching and the school's internet technologies. In many schools this usage was severely restricted. These usage practices are listed in Table 5 and are now described.

Firstly, learners use the Internet for finding information. This ranges from doing school projects to looking up the meaning, spelling or pronunciation of a word. In schools that use e-learning systems, learners can access all class content online. This is very useful if learners have been absent or are just behind. They also access electronic textbooks and online teaching material. Teachers referred to the content on the WCED portal and instructional videos on Mindset Learn. In many schools they were talking about moving to electronic textbooks only. This is beneficial as learners only need to carry a tablet in their bag instead of heavy textbooks and is more cost effective as there are no printing and distribution costs and the risk of misplaced textbooks is averted.

Learners were also using the Internet to submit assignment or perform online assessments. The teachers referred to how useful this was as they were so easily able to control submission deadlines and give extensions of a few hours. Learner's use the Internet to communicate with their peers and their teachers. This could be through e-mail, the e-learning systems or instant messaging such as WhatsApp. This is seen as critical to keeping up with class content and learning how to communicate, especially in group projects. One teacher referred to a school project where learners used the Internet to communicate with learners in India, Sri Lanka, and Bhutan. This is seen as important in helping students to communicate, interact and relate to others.

This feeds into the broader theme of fitting into a digital world. Teachers referred to the Internet preparing learners to have digital literacy, collaboration skills, problem solving skills, creative thinking skills, fourth industrial revolution skills and being citizens of the 21st century.

5 Conclusion

While it is believed that usage of the Internet can positively impact education, its impact, how this is enabled and is experienced is not well understood. Even Bill Gates in 2019 noted that computers have no value in the classroom. The SA WCED e-learning smart schools project in 2019 was connecting schools in Cape Town with broadband. This provided an opportunity to describe the experiences of teachers and explain how relevant factors impact the use of the Internet for teaching and learning in a school environment.

From a theoretical perspective, the resultant model explains how contextual conditions such as an unequal society in SA impacts resultant unequal usage of the Internet for learning. Other dominant factors include the actual characteristics of learners and teachers, the facilities being provided by the WCED and the relevant schools and the practices of teachers. From a practice perspective, this study will be relevant to the local governmental organisations, schools and teachers trying to increase and reduce inequalities in e-learning in schools. The study should also be useful to university lecturers who need to understand how teaching practices in schools are changing and hence the expectations of students arriving on campus.

The paper has limitations. While many documents were collected as part of this study, here we only present the analysis of interview data. Also, due to space constraints, the rich and incredibly insightful comments from teachers had to be severely reduced. With more analysis the various school types could be further contrasted to confirm differences although we argue this does not have much value. In SA we are at the stage that most learners have access to a smart phone. Hence, we recommend further research into cost-effective ways to get access to devices for poorer learners and mechanism that teachers can employ to control phone usage in the classroom.

References

1. Bartikowski, B., Laroche, M., Jamal, A., Yang, Z.: The type-of-internet-access digital divide and the well-being of ethnic minority and majority consumers. A multi-country investigation. J. Bus. Res. **82**, 373–380 (2018)
2. Broadband Commision for Sustainable Development Overview. https://www.broadbandcommission.org/about/Pages/default.aspx. Accessed 08 Mar 2020
3. Belo, R., Ferreira, P., Telang, R.: Broadband in school: impact on student performance. Manage. Sci. **60**, 265–282 (2014). https://doi.org/10.1287/mnsc.2013.1770
4. Hyland, M., Layte, R., Lyons, S., McCoy, S., Silles, M.: Are classroom internet use and academic performance higher after government broadband subsidies to primary schools? Econ. Soc. Rev. **46**, 399–428 (2015)
5. Perry, T.S.: A Crowd of Computer Scientists Lined Up for Bill Gates-But It Was Gavin Newsom That Got Them Buzzing. https://spectrum.ieee.org/view-from-the-valley/robotics/artificial-intelligence/a-crowd-of-computer-scientists-lined-up-for-bill-gatesbut-it-was-gavin-newsom-that-got-them-buzzing. Accessed 08 Mar 2020
6. Blake, B., Pope, T.: Developmental psychology: incorporating Piaget's and Vygotsky's theories in classrooms. J. Cross-Disc. Perspect. Educ. **1**, 59–67 (2008)
7. Salisbury, T.: Education and inequality in South Africa: returns to schooling in the post-apartheid era. Int. J. Educ. Dev. **46**, 43–52 (2016)
8. Alex, J., Juan, A.: Quality education for sustainable development: are we on the right track? Evidence from the TIMSS 2015 study in South Africa. Perspect. Educ. **35**, 1–15 (2017)
9. Christie, P., McKinney, C.: Decoloniality and "Model C" schools: ethos, language and the protests of 2016. Educ. Change **21**, 1–21 (2017)
10. Sutherland, E.: World-class broadband. Digit. Policy Regul. Gov. **19**, 189–209 (2017). https://doi.org/10.1108/dprg-12-2016-0055
11. Casey, B.: Media Release: WCED announces details on e-learning "smart schools" project. https://www.westerncape.gov.za/news/media-release-wced-announces-details-e-learning-%E2%80%9Csmart-schools%E2%80%9D-project. Accessed 08 Mar 2020

12. Carlson, J., et al.: Studies on the Use of Television White Spaces in South Africa: Recommendations and Learnings from the Cape Town Television White Spaces Trial. https://www.tenet.ac.za/tvws/05-cape-town-tvws-trial-learnings-and-recommendations-report. Accessed 2019
13. Kirkpatrick, K.: Bringing the Internet to the (Developing) World. ACM, New York (2018)
14. Tarimo, R., Kavishe, G.: Internet access and usage by secondary school students in Morogoro Municipality, Tanzania. Int. J. Educ. Dev. Using Inf. Commun. Technol. **13**, 56–69 (2017)
15. Opawoye, I., Faruk, N., Bello, O.W., Ayeni, A.A.: Recent trends on TV white space deployments in Africa. Niger. J. Technol. **34**, 556 (2015)
16. Castaldo, A., Fiorini, A., Maggi, B.: Measuring (in a time of crisis) the impact of broadband connections on economic growth: an OECD panel analysis. Appl. Econ. **50**, 838–854 (2018)
17. Minges, M.: Exploring the Relationship between Broadband and Economic Growth. https://openknowledge.worldbank.org/bitstream/handle/10986/23638/WDR16-BP-Exploring-the-Relationship-between-Broadband-and-Economic-Growth-Minges.pdf?sequence=1. Accessed 08 Mar 2020
18. Silles, M., Hyland, M., Layte, R., Lyons, S., McCoy, S.: Are classroom internet use and academic performance higher after government broadband subsidies to primary schools? University of Hull Business School, Hull (2020)
19. Ma, J., Huang, Q.: Does better Internet access lead to more adoption? A new empirical study using household relocation. Inf. Syst. Front. **17**(5), 1097–1110 (2014). https://doi.org/10.1007/s10796-014-9485-6
20. Botha, L.: WCED. E-learning Strategy: Western Cape Education Department briefing. https://pmg.org.za/committee-meeting/28680/. Accessed 08 Mar 2020
21. Whitacre, B., Strover, S., Gallardo, R.: How much does broadband infrastructure matter? Decomposing the metro-non-metro adoption gap with the help of the National Broadband Map. Gov. Inf. Q. **32**, 261–269 (2015)
22. Thapa, D., Sein, M.K.: An ecological model of bridging the digital divide in education: a case study of OLPC deployment in Nepal. Electron. J. Inf. Syst. Dev. Ctries. **84**, 1–9 (2018). https://doi.org/10.1002/isd2.12018
23. Belo, R., Ferreira, P., Telang, R.: Spillovers from wiring schools with broadband: the critical role of children. Manage. Sci. **62**, 3450–3471 (2016)
24. Zhang, M.: Internet use that reproduces educational inequalities: evidence from big data. Comput. Educ. **86**, 212–223 (2015). https://doi.org/10.1016/j.compedu.2015.08.007
25. Li, B., Kong, S.C., Chen, G.: Development and validation of the smart classroom inventory. Smart Learn. Environ. **2**(1), 1–18 (2015). https://doi.org/10.1186/s40561-015-0012-0
26. Makki, T.W., O'Neal, L.J., Cotten, S.R., Rikard, R.V.: When first-order barriers are high: a comparison of second- and third-order barriers to classroom computing integration. Comput. Educ. **120**, 90–97 (2018). https://doi.org/10.1016/j.compedu.2018.01.005
27. Genlott, A.A., Grönlund, A.: Closing the gaps - improving literacy and mathematics by ICT-enhanced collaboration. Comput. Educ. **99**, 68–80 (2016)
28. Sackstein, S., Spark, L., Turner, B.: Technology in the classroom: teachers' mobile technology choice in relation to content creation and distribution. Issues Inf. Sci. Inf. Technol. **16**, 235–254 (2019). https://doi.org/10.28945/4316
29. Hidalgo-Cabrillana, A., Lopez-Mayan, C.: Teaching styles and achievement: student and teacher perspectives. Econ. Educ. Rev. **67**, 184–206 (2018)

30. Selwyn, N.: Minding our language: why education and technology is full of bullshit... and what might be done about it. Learn. Media Technol. **41**, 437–443 (2015)
31. Erdogdu, F., Erdogdu, E.: The impact of access to ICT, student background and school/home environment on academic success of students in Turkey: an international comparative analysis. Comput. Educ. **82**, 26–49 (2015)
32. Clark, W., Luckin, R.: What the research says: iPads in the classroom, pp. 1–31. London Knowledge Lab, Institute of Education, University of London (2013)
33. Kira, E., Mahumbwe, N.: Utilization of the internet facility in Dar es Salaam secondary schools in Tanzania. Global J. Adv. Res. **2**, 1453–1464 (2015)
34. Parry, D., le Roux, D.: Off-task media use in lectures: towards a theory of determinants. Commun. Comput. Inf. Sci. **963**, 49–64 (2019)
35. Saunders, M., Lewis, P., Thornhill, A.: Research Methods for Business Students, 7th edn. Pearson Education Limited, Harlow (2016)
36. Yin, R.K.: Case Study Research: Design and Methods. Sage publications, Thousand Oaks (2017)
37. Marshall, M.N.: Sampling for qualitative research sample size. Fam. Pract. **13**, 522–525 (1996)
38. Braun, V., Clarke, V.: Using thematic analysis in psychology. Qual. Res. Psychol. **3**, 77–101 (2006)

Designing Programming Games for Diversity in Teaching Introductory Programming

Jecton Tocho Anyango(✉) and Hussein Suleman

University of Cape Town, Rondebosch, Cape Town, South Africa
anyjec001@myuct.ac.za

Abstract. Diverse learners from different backgrounds present both significant instructional design challenges and opportunities. Particularly in programming, most serious games that have been created to aid lecturers lack support for diversity. As a result, domain experts who may wish to adopt a Game Based Learning (GBL) approach lack diverse games that are relevant to their local contexts. This paper reports on the design considerations necessary to create diverse programming games for teaching recursion to different novice students. Interviews were conducted with 17 introductory programming (CS1) lecturers from Kenya and South Africa. This was followed by qualitative thematic content analysis. Findings were reviewed by a game expert to validate them from a games design perspective. Results suggest that student background, gender, and culture as well as other factors such as local context, game attributes, pedagogy and practical teaching aspects are core to creating diverse programming games targeting different learners in this generation.

Keywords: Diversity · Games · CS1 · Recursion

1 Introduction

The debate on diversity in Computer Science Education (CSE) is not new. The early work by Camp [11] brought to the attention of the community the declining number of female graduates in Computer Science (CS), pointing out the urgent need for gender inclusivity. Further, the community was called upon to come up with innovative ways to attract and retain students, particularly female, in CS to address the shrinking pipeline. Reiterating low enrollment rates and the lack of diversity in Computer Science education, Bruckman et al. called for changes across CS pipeline [10]. In South Africa, some works have recognized the diverse nature of most classrooms as demonstrated by efforts aimed at preparing

Supported by the Hasso Plattner Institute for Digital Engineering HPI, the National Research Foundation of South Africa (Grant numbers: 85470 and 88209) and University of Cape Town.

G. Wells et al. (Eds.): SACLA 2020, CCIS 1518, pp. 19–36, 2021.
https://doi.org/10.1007/978-3-030-92858-2_2

lecturers to cope with and handle students from culturally and economically divergent backgrounds [5,41].

In this paper, diversity refers to differences among students due to race/gender as well as diversity in (i) motivations, learning needs, educational level and (ii) academic/skills as well as cultural and economic backgrounds [33,47]. Lecturers of introductory programming (CS1) encounter diverse multitudes of students [33,47], bringing with it many challenges [33]. Guzdial and Soloway reiterate that most of these students - the Nintendo or MTV generation - are not easy to motivate using invisible and abstract traditional teaching approaches [19]. Consequently, using old teaching techniques may not be beneficial when engaging with them in the classroom. Probably, new approaches such as digital GBL could be worthwhile. To this extent, some works have suggested the use of games to motivate this generation of students who apparently like engaging with multimedia and animations [19] as well as games [30].

For instance, in South Africa, given the Apartheid system and the diversity of this generation of students entering universities, a pedagogical design for teaching novice programmers CS1 within the local context has been proposed [13]. One of the design principles involves students using construction tools such as puzzles and games while interacting in small groups and speaking in their local languages - mainly Zulu. The results seem promising [13].

Although the motivational and engagement promise of games in learning has been reported in the past [6,28,36,37], designing and developing playable games remains difficult, time consuming and expensive [18,48]. Moreover, most of the material describing games for teaching computing in higher education does not describe how these games have been developed [8]. Additionally, there seems to be no consideration for diversity in design. This was echoed in a report by a special working group on game development for Computer Science Education (CSE) that noted the significant design challenges as well as opportunities presented by different learners given varied gaming backgrounds [24]. If the motivational need of the 21st century learner is to be met through Game Based Learning (GBL) approach then design need to cater to diverse students, especially in CS1 courses.

Given these findings, coupled with the well known difficulty of the recursion topic among students [4,29,31,32,42], the purpose of this study is to identify needs of particular groups of learners with an attempt to understand how programming games that appeal to diverse learners could be designed. It answers the following three questions:

1. What teaching practices are used by programming lecturers in Kenya and South Africa when teaching undergraduate novice students the recursion topic?
2. What teaching examples/analogies are suitable for designing games to teach the recursion topic to diverse novice students?
3. What design considerations are useful when developing a game authoring tool to create games that can teach diverse novice students the recursion topic?

By answering these questions, the novelty and hence contributions of this paper are as follows:

- It is a unique study on teaching CS in developing countries that could be extended to others.
- It highlights factors affecting teaching the recursion topic via games and suggests practical ways to address them as well as suitable teaching analogies.
- It proposes design principles to guide the creation of programming games to teach diverse learners.

The rest of the paper is organised as follows: Sect. 2 is a summary of related work. Section 3 describes the methodology adopted. Results and discussion are presented in Sect. 4, followed by limitations in Sect. 5. Section 6 is conclusions and future work.

2 Related Work

In a recent study, when White, Asian, Hispanic/Latin, and African American/Black students were asked whether they had an interest in video games, their selections for "A lot" were: 52, 48, 27, and 38% respectively [22]. Consequently, a recommendation was made that examples relevant to different groups, including females, and under represented minorities be included in the introductory CS curriculum [22]. In a contrary study, the Lightbot game was used to introduce functions and recursion to students in two post-secondary classrooms from different countries [16]. These were Peru and United States of America (USA) that had different cultural contexts and curricular. In USA, the study was conducted in Berea College (BC), one of the most racially diverse private arts colleges. Results suggested that the game could offer an effective and engaging introduction to the programming concepts regardless of cultural context or course contexts [16]. Given the mixed results from such early works, it would therefore appear that still little is known about how best programming games for diversity could be designed. The work reported in this paper builds on current literature on the topic, but from a different context. It leans on unique experiences drawn from universities from two African countries (Kenya-low-income and South Africa-middle-income).

Advocacy for consideration of diversity, equity, and inclusion in CS curriculum by having content with examples relevant to diverse groups such as female and under represented communities has been on the rise [22,34]. The work by Collain and Trytten [15] investigated diversity by understanding home and precollege computer experiences of students and found out that (i) fathers preferentially taught their sons computing and (ii) there are different pathways students use to join CS or Computer Engineering (CE). Some authors have also argued that well-designed culturally responsive computing can make classrooms more equitable and inclusive to a diverse student population [40]. In the case of South Africa, there is need to design teaching and learning strategies that benefit diverse learners in a regular classroom [50]. Engelbrecht goes on to add that the recognition of diversity in learners is one of the core pillars of inclusivity in education in South Africa [17].

Findings from another work on game design for CSE showed that while gender identity could affect player enjoyment, genre preferences had impact on both experience and outcome performance - independent from gender, hence the need to have a better understanding of target audience [21]. In addition, in an evaluation of a game to enable students to learn the Spanish language, users criticized it for lacking diversity such as diverse game characters, aspects of Spanish culture in game tasks and the inability to customize the game play experience [39].

A claim has also been made that it is best to design games with the background culture of the intended audience in mind [27]. The work by Malliarakis et al. [30] showed that existing educational games designed with a focus on computer programming do not enable lecturers to configure the game environment according to the pedagogical goals related to the respective unit of learning. Moreover, a comparative analysis of five Serious games specifically designed to teach the recursion topic in CS1 (see Table 1) reveals that none are customizable, use simple textbook examples, consider culture or student background in their design. Further, only two are designed with constructivism learning theory [2,20] in mind. In this theory, learning is a constructive process that involves problem solving and understanding as learners build internal models of knowledge [2].

This paper approaches programming game design through the diversity lens. It considers the missing aspects identified in the reviewed literature such as genre preferences, relevant curriculum examples, pedagogy, gender, culture, learning context, game customisation as well as student computer background.

Table 1. Comparative analysis of serious games designed to teach recursion (PYR = Program your Robot)

Feature	Elemental [12]	Cargo Bot [46]	Light Bot [1]	PYR [26]	Saving Cera [7]
Supports writing code	✓	×	×	×	✓
Stack visualisation	✓	✓	×	×	×
Debugging feature	✓	×	×	✓	×
Drag and drop actions	×	✓	✓	✓	×
Learning goals clearly aligned	×	✓	×	✓	×
Supports learning by construction	✓	×	×	×	×
Offers scaffolding	✓	×	×	×	✓
Freedom to explore level	×	✓	×	×	×
Allows customisation	×	×	×	×	×
Considers culture and diversity	×	×	×	×	×
Uses simple textbook examples	×	×	×	×	×

3 Method

3.1 Sampling

The questions raised in this paper are addressed by interviewing 17 CS1 lecturers from higher learning institutions based in Kenya and South Africa. Four participants were female and 13 male. Of the females, 3 came from South Africa and 1 from Kenya. Among the males, 5 were from Kenya and 8 from South Africa. Twelve participants (70%) were very experienced. Overall, the most experienced had 30 years and the least 3 years of experience. One of the participants was a professor and an expert in game development who was recruited on purpose to give insights from game design and development perspective. Six respondents had used games in teaching. Four (66%) had personally designed the games used. Consequently, the data gathered is deemed relevant from both pedagogy and game design perspectives [48]. All the participants from South Africa were recruited from the Southern African Computer Lecturers' Association (SACLA) CS1 lecturers' list created in 2018. On the other hand, it was convenient to recruit respondents from Kenya because of the already established contacts. A summary of the participants is shown in Table 2.

Table 2. Respondents demographic information

Lecturer	Experience (Yrs)	Gender	Country of origin
L1	30	Female	South Africa
L2	13	Male	Kenya
L3	18	Male	South Africa
L4	16	Male	South Africa
L5	8	Female	South Africa
L6	10	Male	Kenya
L7	3	Male	South Africa
L8	6	Male	South Africa
L9	18	Female	Kenya
L10	5	Male	Kenya
L11	10	Male	South Africa
L12	10	Female	South Africa
L13	5	Male	South Africa
L14	25	Male	South Africa
L15	10	Male	Kenya
L16	4	Male	Kenya
L17	14	Male	South Africa

3.2 Procedure

Ethical clearance was first sought from the Faculty of Science Research Ethics Committee of the University of Cape Town and relevant institutions. Invitation emails were then sent out explaining the purpose of the study to the potential participants. Thereafter, interviews were scheduled with the 17 willing participants. This was followed by sending written informed consent forms for signing before the interviews. Each interview lasted an average of 45 min. The interviews were structured to have focused topics aimed at eliciting particular themes relevant to the focus of the research. The interviews contained questions about the teaching experiences, game adoption barriers, teaching practices (e.g. teaching time and class examples) as well as the recommendations on suitable teaching analogies for designing games to teach the recursion topic to diverse students. Additionally, there was a question regarding what the lecturers thought could guide the design and development of a game authoring tool capable of creating different games for diverse students. As the interviews progressed, we began to realise the recurrence of similar data. By the final participant, data collection reached a point where no new insights were generated i.e. data saturation was reached [44]. At this point, no new participants were recruited. Skype interview sessions were recorded using FlashBack Express video screen recorder. For the respondents interviewed face to face, recording was done using a mobile phone audio recorder. A game expert validated the findings.

3.3 Analysis and Presentation

Findings were manually transcribed and summarised into textual data using Microsoft Word. This was followed by Thematic Content Analysis (TCA) [3,43,51]. TCA is a descriptive presentation of qualitative data such as interview transcripts gathered from respondents or other identified texts on the topic of study [3]. Common concepts/ideas in the textual data were identified/highlighted and grouped into common categories/themes. Codes (concepts) were either included or excluded guided by the interview questions. Coding was iteratively done until no new themes emerged. Overall, three iterations were done. The two authors who conducted the analysis met regularly to discuss and agree on the coding/analysis of data. Where appropriate, tables were used to guide discussions.

4 Results and Discussion

Game Adoption Barriers. After asking respondents about their demographic information, an inquiry was done to find out potential reasons why games have not been widely adopted in teaching. Fourteen (82%) reported 'lack of time in the CS1 curriculum and the effort required'; another ten (59%) said 'lecturer background and lack of awareness'; nine (53%) mentioned 'difficulty to get relevant - well designed games and tools'; while three (18%) were not sure if game use would enable lecturers to effectively deliver content. Some remarks follow:

- *"It is hard to infuse games into the curriculum considering time"* [L5].
- *"In my programming education up bringing I have never seen games used in teaching ... so the way I was introduced to programming is what I give back"* [L10].
- *"when I was taught programming in my university I never learned with games"* [L9].
- *"I do not know how effective it would be compared to the traditional learning and so I fear trying to use it because I am not sure of the impact"* [L2].
- *"It takes time and again there are no right game resources"* [L8].

In a broader sense, the findings on adoption barriers could extend the debate on the diversity topic by stressing the need to expose students to introductory programming [15], and use of alternative learning media such as games in an attempt to address the shrinking pipeline problem in Science Technology Engineering and Mathematics (STEM).

4.1 Teaching Practice

Responses to the question on teaching practices and approaches adopted by lecturers when covering the recursion topic can be summarised under two themes: (i) teaching time and number of students and (ii) class examples.

Teaching Time. When asked to comment on time taken to teach the recursion topic and how it was organised, most lecturers said they spent between two and three lectures in total on the topic. However, we noted some variation on the number of hours spent on the topic and how it was spread. As an illustration, a summary of time spent by nine respondents is shown in Table 3.

Table 3. Recursion teaching time

Lecturer	No of lessons	Minutes per lesson
L6, L15	3 to 4	80
L2, L10	2 to 3	60
L3, L4, L13	3	45
L9	3 to 4	60
L5	5 to 7	45

While acknowledging the importance of recursion in computer science, some respondents alluded that the CS1 curriculum did not allocate much time for the topic. Consequently, they were forced to find extra time to reinforce the concept. Comments from such respondents included mentions like:

– *"... in our current curriculum we spend only 2 or 3 lectures if ... may be only 2 but not much"* [L2],
– *"I think it is always sort of the pillar of all these programming courses ... you find it is the one always you repeat after doing ... students will always want you to come back and back and back and back from the first time you introduce it"* [L8],
– *"I usually find time and come back to it after teaching binary search and sorting just to try and reinforce it"* [L5].

Given the pressure of time, two models of practical game adoption were proposed: (i) short games for quick just in time class illustration and (ii) long games for after class practice. Lecturer L1 proposed a design approach that could give students games that allow them to practice for longer hours after class while at the same time give a lecturer the ability to track their progress. She made particular reference to universities that enroll large numbers of students like hers (700 to 1000) for CS1 with different backgrounds. Her proposal was supported by lecturer L9 who added: *"there was a time we wanted to change one of our introductory course ... only to realise that over 70% of the students were government sponsored with no computer background"*. Regarding the number of students, respondents were unanimous that CS1 classes attract large numbers, mostly drawn from different departments.

The finding on time is relevant and in agreement with that of Battistella and Wangenheim [8] on games for teaching computing in higher education that showed that most games are designed to fit into the duration of a class or to be played rapidly (about 10 min). Additionally, other than the practical aspect of time, these results further supplement current literature on programming game design by introducing the practically large number and diverse CS1 students aspect.

Class Examples. Next, participants were asked to comment on the examples they use in class to teach recursion. Findings indicate that a majority (77%) use textbook examples. Popularly mentioned were: Factorial, Fibonacci series, GCD, binary search, power of a number, palindrome, strings, towers of Hanoi, and recursive patterns. Lecturer L11 recalls clearly his experience in teaching recursion by saying: *"I have used patterns of very simple problems to teach recursion"*. It is surprising to note that only two (12%) lecturers said that they use real-life scenarios in class.

4.2 Teaching Analogies and Design Ideas

After questions about recursion teaching practices, participants were asked to recommend suitable teaching analogies (examples) they felt could be designed when creating different programming games to teach novices recursion. Results indicate that fifteen (88%) participants believe that such games should use simple algorithms. In their view, consideration of the fact that the examples used do not introduce extra cognitive load to students is crucial. As exemplars, some of the respondents express their opinion using quotes like:

– *"...give them simple tasks"* [L1]
– *"if not made very complicated can be of value"* [L2]
– *"let them be simple ... not very complicated algorithms ... I tried using a normal fan like pattern in my CS1 class but from experience students found it difficult to understand recursion"* [L3]
– *"simple and basic"* [L4]
– *"simple and neat with simple rewards, levels and scores ... 2D not 3D"* [L5].

A mixed reaction was noticed on whether to use 2D or 3D games. While lecturer L5 who also teaches a design course and lecturer L14 - a professor in game design proposed a 2D game with simple and neat rewards arguing that it will keep it simple and reduce distraction, lecturer L13 with an interest in graphics recommended 3D for their visual learning appeal.

Though textbook examples were suggested by most participants, some respondents brought up a useful debate on the extent to include them. Lecturer L3 felt that it was mandatory to fully include them by arguing: *"classical recursive examples like factorial, binary search, Fibonacci must be fully there in the design knowing very well the importance of maths in computer science"*. On the contrary, L8 saw it differently, commenting: *"I think their use is a disadvantage to students with poor mathematical background"*. This view was supported by lecturer L13 who also cautioned against using only mathematical examples, particularly, those that novice students had not encountered in their high school. He said: *"for me the expectation that every student entering university understands factorial is reasonable ... expectation that they will understand Fibonacci am not sure, I don't think that much thought has gone into the possibility of using real-life scenarios"*.

Findings on class examples and proposed teaching analogies, potentially, could equally add to the literature on the topic in CSE. It seems that inclusivity in programming games design could be advanced by using a variety of teaching examples that would cater for different learning needs. This is in line with the recommendation by Ibe et al. [22]. It could therefore be argued that a design approach that adopts both real life and textbook examples would increase the potential of diverse CS1 educators and learners to embrace a Game Based Learning (GBL) approach.

4.3 Creating Diverse Games

Views of respondents were also sought regarding what could guide the design and development of a game authoring tool capable of creating different instances of games for diverse novice students. To elicit this information, three aspects were investigated: (i) unique design factors; (ii) game attributes and (iii) pedagogy.

Unique Design Factors. From the text analysis of the responses received about unique factors that lecturers thought could guide the design of programming games for diverse students, four key themes emerged, namely: (a) context; (b) gender; (c) religion; and (d) childhood game experience.

One lecturer (relatively young in age) could not hide her frustration with the fact that most digital games played in Africa borrow a lot from the West. She lamented: *"why not contextualize to local games such as ... 'Banu' and local characters"* [L9]. Banu, also known as 'Banta' in Kenya, is a marble game commonly played by two people during childhood. Lecturer L15 who has taught a multimedia course using games also suggested a localised game scenario where a Masai Moran (warrior) fights lions to protect a Manyatta (hut) with the warrior as the main character. Masai is a well known community in Kenya for maintaining their culture.

A similar remark came from another lecturer whose opinion was: *"if you are to design a game for the youth in Kenya then probably, characters based on local games, football stars, celebrities in music industry, and surprisingly politicians will make them interested"* [L2]. A respondent from South Africa echoed the same sentiments by saying: *"background and culture will affect the icons ... shapes ... symbols that you use ... people from different regions are sensitive to different symbols ... a certain icon in Cape Town will mean a totally different thing in Kenya"* [L8].

Three (18%) of the participants considered gender an important factor when designing effective programming games. For instance, after the interview, respondent L12 further sent an email suggesting two different design scenarios for male and female students. Adding his voice to and in support of the gender debate, L14 - a game expert, however suggested gender neutrality in design: "while considering gender ... why not just diverse backgrounds in a neutral way".

Some feedback from four (24%) respondents illustrated that it is equally important to design games that would interest learners based on how they grew up. One of them described his experience when teaching an introductory programming course in the Northern hemisphere. According to him, classes started in October where students did an objects first introductory course from October to December. This was followed by a project in December during the last week of the semester. Students were allowed to pick a project to work on based on scenarios they were interested in. He said: *"one year, two of my students collaborated in a battle Bot game ... a game of Robots ... and they really enjoyed the fact that it was something they could relate with remembering playing it when growing up"* [L11].

Another practical demonstration on how childhood game experience could influence game design for different students in Computer Science Education (CSE) was a revelation by a lecturer from Kenya. She recalls vividly one particular teaching experience saying: *"...I remember there was a time when we had some students actually using AI techniques to develop a local game ... we picked this game so that students could relate with things they have grown up with..."* [L9]. In one unique case, one respondent suggested that lecturers should be careful when using games that include gambling since they may offend some religious beliefs [L11].

As in the National Science Foundation [35] report, this study's findings on gender, context, religion and childhood game experiences contribute to the diversity topic by considering target learners, but from Africa. Particularly, in the context of South Africa, universities predominantly enroll CS1 students from diverse racial, cultural and computer knowledge backgrounds.

Customisable Game Attributes. Participants were further asked to propose game attributes or features a lecturer could change/customise to create different instances of games with unique game play experiences using a game authoring tool. Pointing out the lack of such tools, L15 said: *"I have not seen any educational game development software"*. Supporting this, two other participants observed that game generation would help lecturers save on time spent searching for appropriate teaching games and also present them with variety. One of them said: *"a number of years if you keep using the same examples over and over the students don't see the value ... there is need for refreshing examples used by lecturers"* [L11]. A similar view was held by another who remarked: *"do not constrain them to a particular example"* [L7]. The two views illustrate the need for variety.

Game complexity and challenge levels was also suggested by sixteen (94%) respondents. Some comments included: *"design for challenge at different game levels"* [L1]; *"different levels of difficulty/complexity-for different types of students"* [L6]; *"change game complexity"* [L8]; and *"vary the levels of complexity such that they start by coding simple ones and also to take care of slow learners"* [L10].

Lecturers L2, L6, L9, L10, L12, L15, L16 and L17 felt that characters and their unique parameters like clothes and colour could also provide an opportunity to create unique game experiences. While backing up the idea of game generation, one respondent who considered himself a gamer said: *"I think a nice feature would be a game that has a lot of customization or configuration options for instance I remember in the car race game you could change it to your specification ... change the color, tyres"* [L6]. On the contrary, while stressing the role of visual learning, lecturer L13 observed that the design should steer way from using characters since such could bring in personal issues. Instead, he advised that neutral objects could be used in their place. Other customisable game features suggested were: scene, background, theme, texture, light intensity, interaction style, and character. Lecturer L14 added: "I think that it is very important to consider game genre as another possible attribute". A summary of some customisable game features is presented in Table 4.

Pedagogical Considerations. Finally, participants were asked to reflect on their teaching experiences and share pedagogical/instructional design considerations they deemed suitable for games to teach recursion to different students. It emerged that there is need to balance game play and learning given diversity

Table 4. Customisable game design attributes

Game genre	Game scenario	Back ground	Character	Interaction style	Complexity
Adventure	Maze type game	Color	Cloth	Drag and drop	Simple
Role playing	Factorial of a number	Scene	Hair	Write full code	Intermediate
Simulation	Simple pattern	Texture	Body color	Complete snippet	Complex
Action	Tree traversal game	Sound	Eye color		

in a class. Two lecturers were quick to caution that from their experience one of the reasons why lecturers do not use games in class and some students do not take them seriously is the fact that they do not directly see their educational value. One lecturer advised: *"design to engage them in such a way that they see relevance ... value in it"* [L11]. This was supported by another who added: *"students may feel they are just playing if the feel of playing to learn is not there, particularly the bright ones"* [L8].

Twelve (70%) participants pointed out that individual differences among learners in a class ought to be considered. One remarked: *"I tried using a game in class and the challenge was to get tasks to take care of the weak students ... pitch a game at the very weak students and their capabilities ... design for different types of learners"* [L4]. Another thirteen respondents (76%) made remarks suggesting that game goal and learning goal should be clearly aligned [38], and that a deliberate effort should be made to balance design to cater for all types of learners. In CS, the Cargo Bot serious game by Tessler et al. [46] and Program Your Robot by Kazimoglu et al. [26] are illustrations of design by aligning learning and game goals.

Two lecturers felt that programming game design should allow students to actively construct their programming knowledge through active coding. For instance, lecturer L2 said: *"my experience is that it is important to let students be in control of their learning by writing code"* [L2]. His position was supported by another respondent who commented: *"..design to take them through step by step and to practice by forcing them to write and compile code..."* [L7]. An example of a game in CSE that enables students to learn programming by practicing coding is Elemental the Recurrence [12]. These findings suggest consideration for constructivism learning theory [2,20] in design. Particularly for diverse learners in CS1 [47].

Careful scaffolding and student support, especially for weak students, was yet another admirable instructional design feature according to lecturer L8. He said: *"design for user guided inquiry learning to allow them try different things"*. Scaffolding is demonstrated in Saving Cera serious game by Barnes et al. [7] and Elemental the Recurrence by Chaffin et al. [12] in CSE. Additionally, embedding motivation, reward and challenge in the design was seen as important by respondents L5, L8 and L16. Participant L5 remarked: *"I have taught second year course with games ... students were very interested ... make it fun and amazing ... amusing ...engaging...challenging...motivating"*. Singling out the use of leader boards and points, lecturer L16 who had used games in teaching a third year computer course additionally stressed the need for competition and rewards in design. Agreeing with this position, lecturer L14 (a professor and game expert) added his voice by saying: "reward system is very important from game design point of view ... furthermore the core aim is pedagogy ... to design educational games, particularly to teach the recursion topic".

4.4 Summary

Evidence presented in this paper suggests that game adoption rate still remains low despite the potential motivational and retention impact on learners. Although most lecturers use textbook examples in class, it appears this could also potentially disadvantage some students and further propagate the lack of diversity among CS students. An educational game design approach that strictly adopts only such examples might in itself cause a learning barrier to some learners. Consequently, when designing CS1 educational games for diverse learners, it would be advisable to use textbook examples learners are familiar with, simple algorithms and real life analogies. For example, Depth First Search (DFS) algorithm used in Chaffin et al. [12] would be considered complex for CS1. In addition, ability to have unique characters is equally important. However, caution should be taken and where necessary, neutrality could be considered in design. Some scholars have reiterated the lack of design guidelines/principles to create serious games [14]. This paper attempts to address this gap by proposing the following eight design principles that could guide requirements engineering and creation of diverse games to teach different students programming:

1. Use diverse simple mathematical examples learners are already familiar with to reduce cognitive load, while factoring in practically available time in the curriculum.
2. Use a variety of fun and enjoyable real life analogies that students can easily relate with.
3. Consider particular gender needs.
4. Carefully factor in student's background in terms of computer knowledge, culture and local contexts.
5. Carefully choose game genres bearing in mind religion.
6. Consider students' childhood game experiences.
7. Pitch the design for different learning needs of students in a class, particularly the very weak.
8. Design for game customization to produce a variety of game experiences.

The proposed design principles are supported by findings of other related works. For instance, while calling for a more inclusive design approach, participants in the study by Rankin [39] criticised the game design for lack of (i) diverse game characters that consider gender (principle 3); (ii) cultural aspects (principle 4); and (iii) ability to customise the game play experience (principle 8). Similarly, findings from the works by Khalifa et al. [27] and Rankin [39] confirm that culture (principle 4), customization and game attribute configuration (principle 8) are crucial design considerations when creating diverse games targeting different audiences. Moreover, two key findings from another study that sort the views of 41 game designers and practitioners agree with principles 2 and 7 [23]. In this study, fun is identified as a key design feature for serious games for diversity in line with principle 2. The second key finding suggest that design should gradually release information and allow practice and mastery that can be built upon to give a sense of progression (principle 7). This supports individual progression of students in a CS1 class given different learning abilities [33]. On the other hand, the work by Katz [25] presents seven broad Universal Design (UD) principles for an inclusive education that considers diversity. The first principle, equitable use, maps to design principles 1, 2 and 3 in this paper. The second, flexibility in use, aligns to the proposed principles 4, 5, 6 and 7. Lastly, simple and intuitive use, the third principle aligns to the proposed principle 1.

5 Limitations

In total, 17 CS1 lecturers (domain experts) participated in the qualitative study which was less than expected, but interviews were conducted until data saturation was reached. In addition, most respondents (70%) were very experienced. Another limitation could be the small number of participants who had used games in teaching. Nonetheless, it should be noted that some studies have equally shown that lack of educational game authoring tools [45] and other potential problems/barriers [9,49] have resulted in low GBL adoption in mainstream teaching particularly in developing countries. Therefore, the small number is a clear reflection of this population. Furthermore, a game expert (professor) assisted in reviewing and validating the findings from game design perspective. Subsequently, the small number does not significantly affect the result. Consequently, we argue that findings of the study are valid, broadly representative and could be generalised to teaching CS to other underrepresented minorities and learning situations that represent diversity regardless of country.

6 Conclusions and Future Work

In this paper, we presented findings from a unique study of teaching CS in developing countries. We conducted interviews with 17 CS1 lecturers from higher education and explored how diversity could drive the design and development of programming games. We showed that practical aspects (time, number of students in class and teaching examples) may affect teaching the recursion topic with games. We found that simple mathematical and real life analogies could be designed while considering CS1 syllabus time constraints and diversity among learners. Further, results suggests that domain experts (lecturers and instructors) would be more productive if they were to be given variety of simple games adapted to gender, culture, religion and other environmental factors to aid them in their teaching. Guided by these findings, our next step is to develop a prototype of a game authoring tool (an innovative tool) to be used by programming lecturers to create diverse instances of games to teach the difficult topic of recursion.

Acknowledgements. This research was partially funded by the Hasso Plattner Institute, the National Research Foundation (NRF) of South Africa (Grant numbers: 85470 and 88209) and University of Cape Town. The authors acknowledge that opinions, findings and conclusions or recommendations expressed in this publication are that of the authors, and that the NRF accepts no liability whatsoever in this regard.

References

1. Aedo Lopez, M., Vidal Duarte, E., Castro Gutierrez, E., Paz Valderrama, A.: Teaching abstraction, function and reuse in the first class of CS1: a lightbot experience. In: Proceedings of the 2016 ACM Conference on Innovation and Technology in Computer Science Education, ITiCSE 2016, pp. 256–257. ACM, New York (2016). https://doi.org/10.1145/2899415.2925505
2. Amineh, R.J., Asl, H.D.: Review of constructivism and social constructivism. J. Soc. Sci. Lit. Lang. **1**(1), 9–16 (2015)
3. Anderson, R.: Thematic content analysis (TCA). Descriptive presentation of qualitative data (2007)
4. Anyango, J.T., Suleman, H.: Teaching programming in Kenya and South Africa: what is difficult and is it universal? In: Proceedings of the 18th Koli Calling International Conference on Computing Education Research, Koli Calling 2018, pp. 24:1–24:2. ACM, New York (2018). https://doi.org/10.1145/3279720.3279744
5. Ball, A.F.: Preparing teachers for diversity: lessons learned from the US and South Africa. Teach. Teach. Educ. **16**(4), 491–509 (2000)
6. Barnes, T., Powell, E., Chaffin, A., Lipford, H.: Game2Learn: improving the motivation of CS1 students. In: Proceedings of the 3rd International Conference on Game Development in Computer Science Education, GDCSE 2008, pp. 1–5. ACM, New York (2008). https://doi.org/10.1145/1463673.1463674
7. Barnes, T., Richter, H., Powell, E., Chaffin, A., Godwin, A.: Game2Learn: building CS1 learning games for retention. In: Proceedings of the 12th Annual SIGCSE Conference on Innovation and Technology in Computer Science Education, ITiCSE 2007, pp. 121–125. ACM, New York (2007). https://doi.org/10.1145/1268784.1268821
8. Battistella, P., von Wangenheim, C.G.: Games for teaching computing in higher education-a systematic review. IEEE Technol. Eng. Educ. J. **9**(1), 8–30 (2016)
9. Bayliss, J.D.: Using games in introductory courses: tips from the trenches. In: Proceedings of the 40th ACM Technical Symposium on Computer Science Education, SIGCSE 2009, pp. 337–341. ACM, New York (2009). https://doi.org/10.1145/1508865.1508989
10. Bruckman, A., et al.: “Georgia computes!” improving the computing education pipeline. ACM SIGCSE Bull. **41**(1), 86–90 (2009)
11. Camp, T.: The incredible shrinking pipeline. Commun. ACM **40**(10), 103–110 (1997)
12. Chaffin, A., Doran, K., Hicks, D., Barnes, T.: Experimental evaluation of teaching recursion in a video game. In: Proceedings of the 2009 ACM SIGGRAPH Symposium on Video Games, Sandbox 2009, pp. 79–86. ACM, New York (2009). https://doi.org/10.1145/1581073.1581086

13. Chetty, J., van der Westhuizen, D.: Towards a pedagogical design for teaching novice programmers: design-based research as an empirical determinant for success. In: Proceedings of the 15th Koli Calling Conference on Computing Education Research, Koli Calling 2015, pp. 5–12. ACM, New York (2015). https://doi.org/10.1145/2828959.2828976
14. Chorianopoulos, K., Giannakos, M.N., Chrisochoides, N.: Design principles for serious games in mathematics. In: Proceedings of the 18th Panhellenic Conference on Informatics, PCI 2014, pp. 1–5. Association for Computing Machinery, New York (2014). https://doi.org/10.1145/2645791.2645843
15. Collain, M., Trytten, D.: You don't have to be a white male that was learning how to program since he was five. In: Proceedings of the 50th ACM Technical Symposium on Computer Science Education, SIGCSE 2019, pp. 968–974. ACM, New York (2019). https://doi.org/10.1145/3287324.3287383
16. Duarte, E.V., Pearce, J.L.: A cross-cultural review of lightbot for introducing functions and code reuse. J. Comput. Sci. Coll. **33**(2), 100–105 (2017). http://dl.acm.org/citation.cfm?id=3144645.3144660
17. Engelbrecht, P.: The implementation of inclusive education in South Africa after ten years of democracy. Eur. J. Psychol. Educ. **21**(3), 253 (2006)
18. Green, M.C., Khalifa, A., Barros, G.A., Nealen, A., Togelius, J.: Generating levels that teach mechanics. In: Proceedings of the 13th International Conference on the Foundations of Digital Games, p. 55. ACM (2018)
19. Guzdial, M., Soloway, E.: Teaching the Nintendo generation to program. Commun. ACM **45**(4), 17–21 (2002)
20. Hadjerrouit, S.: A constructivist framework for integrating the java paradigm into the undergraduate curriculum. In: Proceedings of the 6th Annual Conference on the Teaching of Computing and the 3rd Annual Conference on Integrating Technology into Computer Science Education: Changing the Delivery of Computer Science Education, ITiCSE 1998, pp. 105–107. ACM, New York (1998). https://doi.org/10.1145/282991.283079
21. Horn, B., et al.: Design insights into the creation and evaluation of a computer science educational game. In: Proceedings of the 47th ACM Technical Symposium on Computing Science Education, SIGCSE 2016, pp. 576–581. ACM, New York (2016). https://doi.org/10.1145/2839509.2844656
22. Ibe, N.A., Howsmon, R., Penney, L., Granor, N., DeLyser, L.A., Wang, K.: Reflections of a diversity, equity, and inclusion working group based on data from a national CS education program. In: Proceedings of the 49th ACM Technical Symposium on Computer Science Education, SIGCSE 2018, pp. 711–716. ACM, New York (2018). https://doi.org/10.1145/3159450.3159594
23. Isbister, K., Flanagan, M., Hash, C.: Designing games for learning: insights from conversations with designers. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI 2010, pp. 2041–2044. Association for Computing Machinery, New York (2010). https://doi.org/10.1145/1753326.1753637
24. Johnson, C., et al.: Game development for computer science education. In: Proceedings of the 2016 ITiCSE Working Group Reports, pp. 23–44. ACM (2016)
25. Katz, J.: Teaching to Diversity: The Three-block Model of Universal Design for Learning. Portage & Main Press, Winnipeg (2012)
26. Kazimoglu, C., Kiernan, M., Bacon, L., Mackinnon, L.: A serious game for developing computational thinking and learning introductory computer programming. Procedia-Soc. Behav. Sci. **47**, 1991–1999 (2012)

27. Khalifa, A., Perez-Liebana, D., Lucas, S.M., Togelius, J.: General video game level generation. In: Proceedings of the Genetic and Evolutionary Computation Conference 2016, pp. 253–259. ACM (2016)
28. Kirriemuir, J., McFarlane, A.: Literature review in games and learning (2004)
29. Lahtinen, E., Ala-Mutka, K., Järvinen, H.M.: A study of the difficulties of novice programmers. SIGCSE Bull. **37**(3), 14–18 (2005)
30. Malliarakis, C., Satratzemi, M., Xinogalos, S.: Educational games for teaching computer programming. In: Karagiannidis, C., Politis, P., Karasavvidis, I. (eds.) Research on e-Learning and ICT in Education, pp. 87–98. Springer, New York (2014). https://doi.org/10.1007/978-1-4614-6501-0_7
31. McCauley, R., Grissom, S., Fitzgerald, S., Murphy, L.: Teaching and learning recursive programming: a review of the research literature. Comput. Sci. Educ. **25**(1), 37–66 (2015)
32. Miller, C.S., Settle, A., Lalor, J.: Learning object-oriented programming in python: towards an inventory of difficulties and testing pitfalls. In: Proceedings of the 16th Annual Conference on Information Technology Education, SIGITE 2015, pp. 59–64. ACM, New York, NY (2015). https://doi.org/10.1145/2808006.2808017
33. Mohamed, A.: Designing a CS1 programming course for a mixed-ability class. In: Proceedings of the Western Canadian Conference on Computing Education, WCCCE 2019, pp. 8:1–8:6. ACM, New York (2019). https://doi.org/10.1145/3314994.3325084
34. Monge, A.E., Fadjo, C.L., Quinn, B.A., Barker, L.J.: EngageCSEdu: engaging and retaining CS1 and CS2 students. ACM Inroads **6**(1), 6–11 (2015)
35. National Science Foundation: Women, minorities, and persons with disabilities in science and engineering: 2013. Technical report, National Center for Science and Engineering Statistics, Directorate for Social, Behavioral and Economics Science (2013)
36. Papastergiou, M.: Digital game-based learning in high school computer science education: impact on educational effectiveness and student motivation. Comput. Educ. **52**(1), 1–12 (2009)
37. Parberry, I., Kazemzadeh, M.B., Roden, T.: The art and science of game programming. In: Proceedings of the 37th SIGCSE Technical Symposium on Computer Science Education, SIGCSE 2006, pp. 510–514. ACM, New York (2006). https://doi.org/10.1145/1121341.1121500
38. Prensky, M.: Digital game-based learning. Comput. Entertain. **1**(1), 21 (2003)
39. Rankin, Y.A.: Diversity by design: female students' perception of a Spanish language learning game. In: Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems, CHI EA 2016, pp. 670–679. ACM, New York (2016). https://doi.org/10.1145/2851581.2851610
40. Richard, G.T.: Video games, gender, diversity, and learning as cultural practice: implications for equitable learning and computing participation through games. Educ. Technol., 36–43 (2017)
41. Robinson, M., Zinn, D.: Teacher preparation for diversity at three South African universities. J. Educ. **42**(1), 61–81 (2007)
42. Schulte, C., Bennedsen, J.: What do teachers teach in introductory programming? In: Proceedings of the Second International Workshop on Computing Education Research, ICER 2006, pp. 17–28. ACM, New York (2006). https://doi.org/10.1145/1151588.1151593
43. Spiel, K., et al.: "It started as a joke": on the design of idle games. In: Proceedings of the Annual Symposium on Computer-Human Interaction in Play, CHI PLAY 2019, pp. 495–508. ACM, New York (2019). https://doi.org/10.1145/3311350.3347180

44. Corbin, J., Strauss, A.: Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory. Sage, Thousand Oaks (1998)
45. Tang, S., Hanneghan, M.: A model-driven framework to support development of serious games for game-based learning. In: Developments in E-systems Engineering (DESE 2010), pp. 95–100. IEEE (2010)
46. Tessler, J., Beth, B., Lin, C.: Using cargo-bot to provide contextualized learning of recursion. In: Proceedings of the Ninth Annual International ACM Conference on International Computing Education Research, ICER 2013, pp. 161–168. ACM, New York (2013). https://doi.org/10.1145/2493394.2493411
47. Thevathayan, C., Hamilton, M.: Supporting diverse novice programming cohorts through flexible and incremental visual constructivist pathways. In: Proceedings of the 2015 ACM Conference on Innovation and Technology in Computer Science Education, ITiCSE 2015, pp. 296–301. ACM, New York (2015). https://doi.org/10.1145/2729094.2742609
48. Torrente, J., del Blanco, A., Cañizal, G., Moreno-Ger, P., Fernández-Manjón, B.: <e-Adventure3D>: an open source authoring environment for 3D adventure games in education. In: Proceedings of the 2008 International Conference on Advances in Computer Entertainment Technology, ACE 2008, pp. 191–194. ACM, New York (2008). https://doi.org/10.1145/1501750.1501795
49. Torrente, J., Blanco, A., Marchiori, E., Moreno Ger, P., Fernández-Manjón, B.: <e-Adventure>: introducing educational games in the learning process, pp. 1121 – 1126, May 2010. https://doi.org/10.1109/EDUCON.2010.5493056
50. Walton, E., Nel, N., Hugo, H., Muller, H.: The extent and practice of inclusion in independent schools in South Africa. S. Afr. J. Educ. **29**(1), 105–126 (2009)
51. Willson, R.: Analysing qualitative data: you asked them, now what to do with what they said. In: Proceedings of the 2019 Conference on Human Information Interaction and Retrieval, CHIIR 2019, pp. 385–387. ACM, New York (2019). https://doi.org/10.1145/3295750.3298964

A Proposed Structure for Managing IT Diploma Qualifications in South Africa

Andre P. Calitz[1(✉)], Lester Cowley[1], and Sue Petratos[2]

[1] Department of Computing Sciences, Nelson Mandela University, Port Elizabeth, South Africa
{Andre.Calitz,Lester.Cowley}@Mandela.ac.za

[2] School of IT, Nelson Mandela University, Port Elizabeth, South Africa
Sue.Petratos@Mandela.ac.za

Abstract. IT diploma and advanced diploma qualifications are presently offered by public and private universities in South Africa. The public universities include research universities, comprehensive universities and Universities of Technology. Historically in South Africa, IT diploma programme content was collaboratively managed by the Technikon Computer Lecturers Association (TECLA). TECLA comprised of Technikon academics, who met annually to discuss the IT diploma programmes and curriculum content. TECLA ceased to exist when the South African Government introduced new university structures in 2005. Internationally and in South Africa, university IT programmes are accredited by professional accreditation bodies, such as ABET and the British Computer Society. Presently, there is no academic body that coordinates the curriculum and ensures standardised curriculum content and quality assurance standards for the different types of IT diploma qualifications offered by the various academic institutions in South Africa. In this paper, a South African IT Diploma Advisory Board (SAITDAB) is proposed that will be associated with the South African Computing Accreditation Board (SACAB). The SACAB is established under the control of the Institution of IT Professionals of South Africa (IITPSA). The purpose of this research study is to investigate how other international IT professional bodies manage diploma programme content and standards and to obtain the opinion of academics at institutions offering IT diploma programmes on the proposed establishment of the SAITDAB. The findings indicate strong support for establishing a rationale and process for the creation of a Body of Knowledge for IT diploma and IT advanced diploma programmes offered by South African Universities by the SAITDAB.

Keywords: IT diploma programmes · Accreditation authorities · IITPSA · SAITDAB

G. Wells et al. (Eds.): SACLA 2020, CCIS 1518, pp. 37–56, 2021.
https://doi.org/10.1007/978-3-030-92858-2_3

1 Introduction

Internationally, non-governmental organisations, such as the Accreditation Board for Engineering and Technology, Inc. (ABET), accredit post-secondary education programmes in applied and natural sciences, computing, engineering and engineering technology [1]. In several countries, national IT professional bodies, such as the British Computer Society (BCS) [5], Australian Computer Society (ACS) [3] and more recently the IT Professionals New Zealand (ITPNZ) [16] accredit university Information Technology (IT) related degree and diploma programmes. In South Africa, Computer Science and Information Systems degree programmes have been accredited by ABET and the BCS.

The Seoul Accord [23] is an international accreditation agreement for professional computing and information technology academic degree programmes. The BCS [5], ACS [3] and the ITPNZ [16], for example, are signatories to the Accord and are responsible for the accreditation of IT related degree and diploma programmes in their specific countries. The South African Computing Accreditation Board (SACAB) [21] was established by IITPSA [14] in 2014 and will be responsible for the accreditation of IT related degree programmes offered at universities in South Africa. The IITPSA intends to become a signatory of the Seoul Accord.

The current South African Higher Education (HE) landscape was implemented in 2005 and caters for three types of public universities, namely Traditional/Research Universities, Comprehensive Universities and Universities of Technology. Comprehensive Universities were established by the amalgamation of selected traditional/research universities with Technikons. The remaining Technikons became Universities of Technology. Before 2005, the Technikons were responsible for offering IT diploma programmes. Presently, South Africa has 26 public universities and over 22 private universities [20].

Prior to the restructuring of the South African HE landscape, the IT academics from the various Technikons in South Africa formed an academic body called the Technikon Computer Lecturers Association (TECLA). The members of TECLA met annually at a conference to discuss IT diploma curriculum programmes, industry liaison and related topics [24]. TECLA ceased to exist in 2005 when the South African Government introduced the current HE landscape. A new body, HEICTA (Higher Education Information and Communication Technology Association), was established in 2006 and was also involved with industry liaison and curriculum topics [4]. For a number of years, it provided collaboration amongst IT departments offering IT diplomas and liaised with industry. HEICTA gradually ceased to exist. The last HEICTA activity found on social media was in 2013.

National IT diploma programmes are considered equivalent to a foundation degree or the first two years of a bachelor's degree [12]. A diploma equips students with focused knowledge and skills in a particular field, such as IT. The aim of the IT diploma programmes is to enable qualifying students to analyse, design, develop and maintain, for example, web systems, database systems, programming products and communication networks for a business. Diploma programmes in South

Africa are at National Qualification Framework (NQF) level 6, advanced diplomas at NQF level 7 and post-graduate diplomas at NQF level 8.

Currently there is no academic body or organisation that is responsible for the accreditation of IT diploma and IT higher diploma programmes in South Africa. The South African Computing Accreditation Board (SACAB) [21] proposed the formation of a South African IT Diploma Advisory Board (SAITDAB) at the Southern African Computer Lecturer's Association (SACLA) conference in 2018 [22]. The aim of this paper is to report on a national survey that was conducted among Heads of Departments at Comprehensive Universities and Universities of Technology offering IT diploma and advanced diploma programmes concerning the proposed implementation of a SAITDAB.

The paper is structured as follows: In Sect. 2, the research problem and the research design are presented. The literature review in Sect. 3 reports on the various IT diploma programmes offered at South African universities and proposes the establishment of a SAITDAB. Section 4 reports on the different IT diploma programmes offered by public and private universities in South Africa and Sect. 5 presents and discusses the results of the national survey. In Sect. 6, a discussion of the proposed structure and process for the creation of an IT diploma programme Body of Knowledge (BOK) is presented. Section 7 presents the conclusions of the paper.

2 The Research Problem and Research Design

Comprehensive Universities, Universities of Technology and private universities in South Africa offer various IT diploma and IT advanced diploma programmes. The universities offering the various IT diploma programmes formerly collaborated through bodies such as TECLA and HEICTA [2]. The problem statement investigated in this study is that South African universities offering IT Diploma programmes do not have a professional body accrediting and governing IT diploma programme content, standards and quality assurance.

The research objectives of this study are to:

- Propose the establishment of a South African IT Diploma Advisory Board (SAITDAB);
- Identify and record the IT diploma and IT advanced diploma programmes offered at Comprehensive Universities, Universities of Technology and private universities in South Africa;
- Conduct a survey amongst these universities to determine the support for the establishment of the SAITDAB; and
- Propose a rationale and process for drafting guidelines for an IT diploma programme and advanced diploma programme BOK for South African universities.

The research design used comprises a literature review, an investigation into diploma offerings by tertiary institutions in South Africa and a survey of HODs and academics at Comprehensive Universities and Universities of Technology on the establishment of a SAITDAB (Appendix A). The research approach followed was the compilation of a questionnaire (Appendix A), that was distributed to the HODs of departments offering IT diploma programmes at 14 Comprehensive Universities and Universities of Technology. A total of 29 HODs were identified using the SACLA HOD mailing list at the 14 public universities. Seven universities completed and returned the survey request over a three-week period, following three requests for participation. The qualitative survey data were thematically analysed.

3 Literature Review

3.1 Academic Programme Accreditation

The Seoul Accord [23], established in 2008, is an international accord of bodies accrediting tertiary-level computing and IT-related programmes. It is a multilateral agreement amongst agencies responsible for accreditation or recognition of tertiary-level computing and IT-related qualifications, in their respective countries. Signatories include ABET, Inc. [1], Australia Computer Society (ACS) [3], the Japan Accreditation Board for Engineering Education (JABEE) [18], Canada's Association for IT Professionals (CIPS) [7], the British Computer Society - Chartered Institute for IT (BCS) [5] and more recently, the Institute of IT Professionals New Zealand (IITPNZ) [16] and the Philippine Computer Society.

Internationally, organisations such as the IEEE [13] and the ACM [2] have provided curricula requirements or content for degree programmes in Computer Science [9], Information Systems [15], Information Technology [17] and other IT related fields. All curricula recommendations by the ACM [2] provide guidelines for undergraduate degree programmes in the fields listed above.

In related disciplines such as Engineering, the Engineering Council of South Africa is empowered by the Engineering Profession Act, 2000 (Act 46 of 2000), to conduct accreditation visits to educational institutions that offer Engineering programmes to determine whether the Engineering qualifications offered can be recognised by the Council for purposes of professional registration. The educational programmes considered for accreditation are degree and diploma programmes [10].

3.2 Accreditation by Professional IT Organisations

Internationally, departments offering IT related programmes seek accreditation from professional bodies. The commissions that accredit programmes are the Engineering Accreditation Commission (EAC), who accredits civil engineering programmes; the Engineering Technology Accreditation Commission (ETAC) of ABET who is responsible for the accreditation of civil engineering technology programmes. The Applied and Natural Science Accreditation Commission (ANSAC) of ABET accredits applied and natural science programmes at the following levels: associate (two-year degree), bachelor (four-year degree) and master (post-graduate programmes). The Computing Accreditation Commission (CAC) of ABET accredits computing programmes, specifically four-year bachelor degrees.

The seven main signatories of the multilateral Seoul Accord for computing programmes accredit degree and diploma programmes [23]. The *Accreditation Board for Engineering Education in Korea (ABEEK)* accredits mainly engineering degree programmes, BSc in Engineering and Computer Science related fields. The *Accreditation Board for Engineering and Technology (ABET) - ABET, Inc.* is a nonprofit, non-governmental agency that accredits academic programmes in applied and natural science, computing, engineering and engineering technology [1]. ABET accreditation provides assurance that a college or university programme meets the quality standards of the profession for which that programme prepares graduates. ABET's Computing Accreditation Commission (CAC) accredits computing programmes for four-year bachelor degrees.

The *Australian Computer Society (ACS)* accredits bachelor and master degree programmes [3]. *The British Computer Society (BCS) Chartered Institute for IT* accredits degree and diploma programmes in the UK [5]. The accreditation includes both undergraduate and post-graduate programmes. The *Hong Kong Institution of Engineers (HKIE)* accredits Engineering Degree Programmes, Computer Science Programmes and Higher Diploma Programmes. The *Institution of Engineering Education Taiwan (IEET)* accredits mainly engineering programmes.

The *Japan Accreditation Board for Engineering Education (JABEE)* was established in 1999 and started accreditations in 2001 [18]. The JABEE has been a member of the Washington Accord, an agreement which provides a mechanism for mutual recognition between signatory bodies of engineering education accreditation processes. The JABEE became a member of the Seoul Accord in 2008. The Accreditation Commission reports to the Board of Directors of the JABEE, as indicated in Fig. 1.

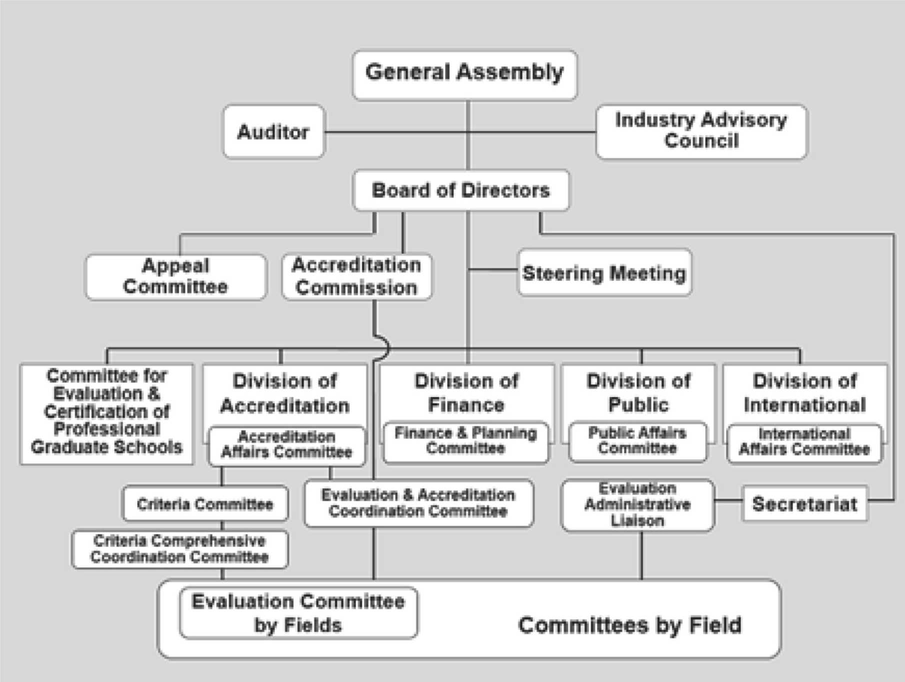


Fig. 1. JABEE organisational structure and Accreditation Commission [18]

Six organisations hold provisional status and include *Canada's Association for Information Technology Professionals (CIPS)* [7] and the *Institute of IT Professionals New Zealand (IITPNZ)* [16]. The CIPS (2021) accredits degree programmes, such as Computer Science and Software Engineering degrees, interdisciplinary degrees, College/Applied degrees and Computer Technology diplomas [8]. The IITPNZ accredits bachelor degrees in engineering and IT [16].

3.3 South Africa: Institute of IT Professionals South Africa (IITPSA)

In South Africa, the accreditation of IT related degree and diploma programmes will be managed by the South African Computing Accreditation Board (SACAB) [21]. The SACAB was commissioned by the Institute of IT Professionals South Africa (IITPSA) [14] and the Southern African Computer Lecturers' Association (SACLA) [22]. IITPSA is affiliated to the International Federation for Information Processing (IFIP) and is a South African Qualifications Authority (SAQA) recognised professional body. The IITPSA is a member of the International Professional Practice Partnership (IP3) and associated with the Seoul Accord.

A generic national IT diploma programme is no longer followed by Higher Education Institutions (HEIs) in South Africa, as was done prior to 2005. Most Comprehensive Universities, Universities of Technology and private universities have tried to create more distinct and divergent IT diploma programmes to allow students access to the various fields in IT and to distinguish their programmes from those offered by other universities, although most do still have a focus on IT, Software Development and Communication Networks (Table 4). All institutions offering IT diplomas are now responsible for registering their own qualifications with SAQA. This diverse and independent process, creates concerns regarding the quality assurance of these qualifications over the long term as there is no authorised body to hold the HEIs accountable for the renewal/maintenance of the relevance and quality of the programme after registration by SAQA.

3.4 The SAITDAB Proposal

SACAB [21] proposes the establishment of SAITDAB, as indicated in Fig. 2. The body will consist of members from universities in South Africa offering IT diploma programmes. SAITDAB will be responsible for the recommendation of IT diploma and advanced diploma standards, core competencies and curriculum content. SAITDAB will further assist SACAB in the accreditation of IT diploma and advanced diploma programmes at South African universities.

The focus of SAITDAB will be to:

- Understand the industry requirements for skills and qualifications of IT diploma students;
- Specify the BOK for IT Diploma specialisation programmes;
- Define IT diploma programme core competencies;
- Define specialisation fields core competencies;
- Define IT diploma programme deviation percentages content per IT diploma specialisation field; and
- Ensure that HEIs offer diploma programmes registered with SAQA.

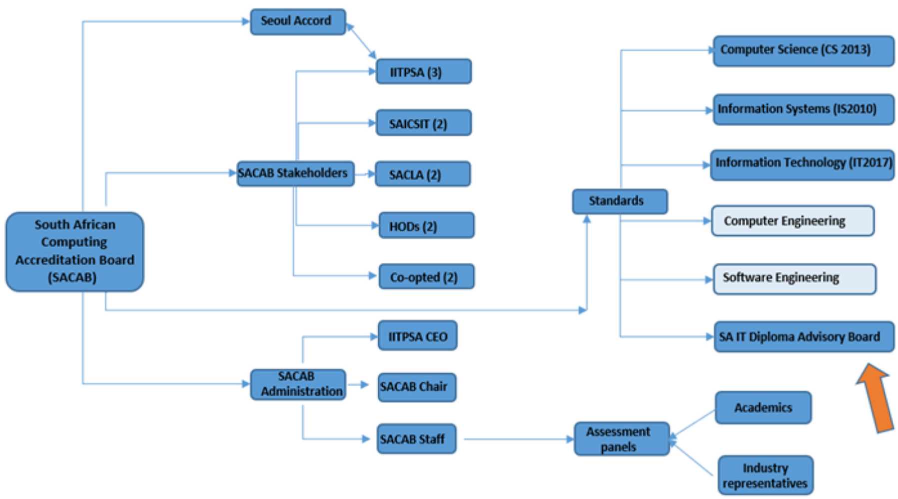


Fig. 2. Proposed SAITDAB structure

3.5 Literature on IT Diploma Programmes

Internationally, an academic degree is a qualification awarded to students upon successful completion of a course of study in HE, usually at a college or university. HEIs commonly offer degrees at various levels, including bachelors, masters and doctorates, often alongside other academic certificates and professional degrees [5]. The most common undergraduate degree is the bachelor's degree (NQF 7 in South Africa), although in some countries there are lower level higher education qualifications that are also titled degrees (e.g. associate degrees and foundation degrees). These are similar to Higher Certificate (NQF level 5) and diploma programmes at academic level (NQF level 6) in South Africa [12].

Universities in South Africa offer IT diploma programmes specialising in various fields, such as Information Technology, Software Development, Communication Networks and Support Services (Tables 4 and 5). Ten HEIs in South Africa offer advanced diplomas in IT, specialising mainly in Information Technology, Application Development and Communication Networks. All institutions offered one or more of the registered specialisation fields.

According to the Nelson Mandela University [19], the academic emphasis on learning in a diploma qualification, is on a practical approach based on theoretical foundations. These qualifications are career-focused. Diploma studies are offered on a full-time and/or part-time basis but are generally offered over a three-year period. Industry training varies from programme to programme and can be anything from six months to a year. With a degree qualification, the emphasis is placed on an academic approach to a specified field of study or profession, such as law, pharmacy and architecture. Degree programmes are offered on a full-time or part-time basis over three to four years, depending on the type of degree programme. Certain programmes require industry training of about six to eight weeks.

4 South African Universities Offering IT Diploma Programmes

A desktop study was done, using the websites of the universities listed in Tables 1 to 3, to determine the IT diplomas and advanced diplomas offered by public and private HEIs. Table 1 indicates the IT diplomas and advanced diplomas offered by Comprehensive Universities, Table 2 IT diploma qualifications those offered by Universities of Technology and Table 3, IT diploma qualifications offered by private universities. One Comprehensive University's departmental website only listed the names of staff members, with no details of qualifications offered.

Table 1. Comprehensive Universities in South Africa

Institution	School/Department	IT Diploma programmes offered
Nelson Mandela University	School of IT	– Diploma - IT: Communications Networks – Diploma - IT: Software Development – Diploma - IT: Support Services – Advanced Diploma - IT
University of Johannesburg	School of Consumer Intelligence and Information Systems	– Diploma - Business IT – Advanced Diploma - Business IT
University of South Africa	School of Computing	– Diploma - IT – Advanced Diploma - Information Resource Management
University of Venda	Computer Science and IS	– Only degree programmes offered
University of Zululand	Computer Science	– Only degree programmes offered
Walter Sisulu University	Science, Engineering and Technology	– National Diploma in IT - Software Development – National Diploma in IT - Support Services – National Diploma in IT - Communication Networks

A summary of the IT diplomas offered by universities in South Africa (Table 4) indicates that the most popular IT diploma specialisation programmes that universities offer are Information Technology (n = 7), Software Development (n = 6), Communication Networks (n = 4) and Support Services (n = 3). The most popular IT advanced diplomas offered by HEIs in S.A. (Table 5) are an Advanced Diploma in Information Technology (n = 5), an Advanced Diploma in Application Development (n = 3) and an Advanced Diploma in ICT: Communication Networks (n = 2). A total of 29 IT diplomas and 14 IT advanced diplomas are offered by 12 public universities and 5 private universities, as indicated in Tables 1 to 3.

South Africa has over 35 private universities, offering a range of qualifications, including higher certificate, diploma and degree programmes relating to IT, IS and CS [6]. Table 3 highlights the diploma programmes offered by some private institutions in South Africa. Varsity College is a brand of the Independent Institute of Education (IIE), a registered private higher education provider.

Table 2. Universities of Technology in South Africa

Institution	School/Department	IT Diploma programmes offered
Cape Peninsula University of Technology	Informatics and Design	– Diploma: ICT - Multimedia Applications – Diploma: ICT - Communications Networks – Diploma: ICT - Applications Development – Advanced Diploma in ICT - Multimedia Applications – Advanced Diploma in ICT - Communication Networks – Advanced Diploma in ICT - Application Development
Central University of Technology	Department of Information Technology	– Diploma - Computer Networks – Diploma - IT – Advanced Diploma - Computer Networks – Advanced Diploma - IT
Durban University of Technology	Department of Information and Communication Information	– Diploma in ICT - Applications Development – Diploma in ICT - Business Analysis
Mangosuthu University of Technology	Department of Accounting and Informatics	– Diploma in IT
University of Mpumalanga	School of Computing and Mathematical Sciences	– Diploma in ICT - Applications Development – Advanced Diploma in ICT - Applications Development
Sol Plaatje University	School of Natural and Applied Sciences	– Diploma in ICT - Applications Development – Advanced Diploma in ICT - Applications Development
Tshwane University of Technology	Department of Information Technology and Informatics	– Diploma in IT – National Diploma in IT: Communication Networks – National Diploma in IT: Support Services – Diploma in Informatics – National Diploma in IT: Business Applications – Advanced Diploma in Informatics – Advanced Diploma in IT
Vaal University of Technology	Department of Information and Communication Technology	– Diploma in IT – Advanced Diploma in IT
	Department: Process Control and Computer Systems	– Diploma in Computer Systems

Table 3. Private universities in South Africa

Institution	School/Department	IT Diploma programmes offered
Boston City Campus and Business College	IT	– Diploma in Systems Development – Diploma in Network Systems
Damelin	The School of Information Technology	– Diploma in IT
Monash South Africa	IT	– Diploma in IT
MANCOSA	Faculty of Business, Engineering and Technology	– Higher Certificate in Information Technology – Postgraduate Diploma in Information and Technology Management
Varsity College	Faculty of Information & Communications Technology	– IIE Diploma in Information Technology in Software Development

Table 4. Summary of the number of IT diplomas offered by universities in South Africa

IT Diplomas offered	Count
Information Technology	7
Software/Application Development	6
Communication Networks	4
Support Services (Help Desk and Infrastructure Management)	3
Computer Systems/Network Systems/Networking	3
Business Applications/Business IT	2
Multimedia Applications	1
Business Analysis	1
Informatics	1
Systems Development	1

Table 5. Summary of the number of IT advanced diplomas offered by universities in South Africa

IT Advanced Diplomas offered	Count
Advanced Diploma: Information Technology (IT)	5
Advanced Diploma: Application Development	3
Advanced Diploma: ICT: Communication Networks	2
Advanced Diploma: ICT: Multimedia Applications	1
Advanced Diploma: Business IT	1
Advanced Diploma: Informatics	1
Advanced Diploma: Information Resource Management	1

5 Results of the National Survey

The following section provides a summary of the responses received from seven of the twelve public universities in South Africa offering IT diploma programmes. The survey (Appendix A) was sent to the 29 HODs offering IT diploma programmes at the 14 public universities listed in Table 1 and 2. Seven universities, three Comprehensive Universities and four Universities of Technology, responded to the call for participation in the survey. The following paragraphs address each question in the survey.

The respondents (Q1) indicated that they no longer offer National Diplomas and B-Tech programmes, but now offer Diplomas and Advanced Diplomas. The diploma programmes offered by some institutions, as indicated on the institution's website (Tables 1 and 2), did not always correspond to the responses received from academics who completed the survey. Academics at one University of Technology, for example, indicated that they offer a Diploma in Information Technology, focusing on Business Analysis and Software Development. However, the website of the university did not indicate this differentiation. One respondent indicated that they will only offer a National Diploma in IT - Business Applications and a Diploma in Informatics in 2020, including only one Advanced Diploma in Informatics. The website in 2021, however indicated that the university offered additional diploma programmes.

The respondents (Q2) indicated that they decided on diploma programme offerings based on the "old" national diploma, resources available, input from industry and in some cases, on recommendations provided by an Industry Advisory Board. Four institutions indicated that they have a programme review committee, managed by the institution. One respondent indicated that their decisions were "based on the Higher Education Qualification Sub Framework (HEQSF) aligned framework" and the requirements of industry. One institution indicated that they initially (prior to 2005) used the National Diploma in IT as a baseline, however during the past decade, institutions decided individually on diploma programme offerings. The former Department of Higher Education (DHE) detailed National Diploma programmes which Technikons could present, if they chose to do so. They could deviate up to 50% from these programmes. The DHE has been replaced by the Department of Higher Education and Training (DHET). The DHE regulated National Diploma programmes via convenor bodies (as for the old Technikon system) but the DHET use the current HEQSF to assess diploma programmes.

The respondents (Q3) indicated that their IT diploma programmes are still based on the structure of the old national diplomas. The respondents further indicated that they have workshops with academics, stakeholders and curriculum specialists to advise on programme content. When new programmes are introduced, benchmarking on international curricula is used to provide guidelines. The use of the programme review committee at an institution further assists with decisions on programme offerings. Institutions indicated that a programme review committee, managed by the institution assists with programme structure and content. The Industry Advisory Board at universities further played a leading role in guiding decisions regarding programme content.

The respondents (Q4) all indicated that other institutions' websites and in some cases, academics, were consulted on deciding on the structure and content of diploma programmes. One institution indicated they also investigated overseas university offerings. Another institution indicated that their diploma programmes are still based on discussions held at the TECLA and HEITCA conferences. International companies, such as CISCO, further provide "major input into the diploma programmes".

The majority (75%) of the respondents (Q5) indicated that they were not intending to introduce new IT diploma programmes in the next two years. One University of Technology is introducing a Post-Graduate Diploma (Information Technology) and another a Post-Graduate Diploma in Informatics in 2021. One Comprehensive University is introducing a Post-graduate Diploma in Cybersecurity. The major recurriculation of diploma programmes is presently been undertaken at some institutions.

One respondent (Q6) indicated that they have recurriculation exercises with industry advisors, academics and education specialists within the university, every few years. They make use of external examiners and moderators. Advisory Boards and alumni were mentioned as methods used to ensure that IT diploma programmes remain current. Four institutions indicated that they are currently planning to review their programmes. This exercise is undertaken every three years.

The Quality Promotion Unit (QPU) is used at some universities to ensure the quality (Q7) of the different programmes annually. The QPU conducts annual audits of the departments and qualifications offered. Consultation with the Industry Advisory Board, the use of external examiners and moderators for most subjects further assists with quality assurance. Employing academics with the relevant qualifications further assisted with quality assurance. The important role of the Industry Advisory Board was emphasised by most departments.

Generally (Q8), the respondents did not collaborate with other institutions to ensure the quality of the diploma programmes that they offered. The use of external examiners and moderators assisted in quality assurance of specific subjects. Most respondents indicated that they had attended the TECLA conferences and that it was an excellent organisation (Q9). The respondents indicated that if the proposed SAITDAB could fulfill a similar role, it would be an excellent initiative. The respondent from one Comprehensive university was unaware of the existence of TECLA.

All respondents (Q10) indicated that they would support the proposed SAITDAB. They further indicated that the recognition of qualifications offered by other institutions would be made easier and students would be able to transfer between institutions more easily. Students would also be able to complete outstanding modules through institutions offering distance education, such as UNISA. The respondents indicated that the proposed SAITDAB would assist in accreditation and review processes and guidelines for standardisation should further be provided. The responses included the following remark: “This would provide set standards and help with articulation”.

All respondents (Q11), except one, indicated that they would support a national IT diploma and advanced diploma programme BOK as proposed by the SACAB. A national IT diploma and advanced diploma programme BOK will ensure a set of standards all institutions could adhere to, without losing the diversity of specific diploma programmes offered by institutions. The general consensus was the acceptance of a core IT diploma and advanced diploma programme BOK, with the “ability to be unique and different”.

The benefits (Q12) of an IT diploma and advanced diploma programme BOK would ensure that institutions offer a recognised and accepted core programme, endorsed by all institutions and the SACAB. This will further assist with the employability of students and the provision of relevant standardised qualification offerings. The respondents generally wanted the ability to supplement the core IT diploma BOK with local industry requirements.

All institutions that responded (Q13) provided a list of names of academics that could serve on the proposed SAITDAB. The general comments (Q14) provided by all respondents included comments thanking the authors for the initiative. One respondent indicated that this is a “long overdue initiative”. The respondents expressed a desire that their IT diploma programme offering “be different, offer local specialised content and be able to differentiate their IT diploma offerings from other institutions, based on local industry requirements”.

6 Proposed Rationale and Process for Drafting an IT Diplomas’ and Advanced Diplomas’ Programme BOK

In South Africa, all qualifications offered by Higher Education Institutions must be approved by the Department of Education (DOE), the South African Qualifications Authority (SAQA) and the Higher Education Quality Committee (HEQC) before the institution can offer them. All programmes must be Higher Education Qualifications Framework (HEQF) compliant. The purpose of the Higher Education Qualification Sub Framework (HEQSF) [12] is to define the relationships between qualification types.

Internationally, organisations such as the ACM [2] have made efforts to specify requirements or content for degree programmes in Computer Science [9], Information Systems [15] and Information Technology [17]. There are a number of additional Information Technology Body of Knowledge guides. These include the Canadian Association of IT Professionals’ Guide to the Common Body of Knowledge for Computing and IT (CBOK) and the European Foundational ICT Body of Knowledge [8,11].

All curricula recommendations (e.g. ACM [1]) prescribe core degree programme content, both specialised and generic. Institutions can then select additional modules from the elective modules specified to compile an institution-specific degree programme. The responses received from the survey strongly support the creation of a core IT diploma and advanced diploma programme Body of

Knowledge (BOK). Elective modules for the specific diploma programmes, such as Software Development, Communication Networks, Support Services and Web Development can be incorporated by an institution to provide a differentiated diploma offering. As indicated in Fig. 3, all IT diploma programmes will have a common core first year. In the second-year, specialisation core modules are proposed for specialisation IT diplomas such as Software development, Information Technology and Communication Networks. In the third year, an institution can select elective modules for the IT advanced diploma in, for example, Information Technology or Communication Networks.

The rationale for such a core IT diploma and advanced diploma programme BOK is that it prepares students with the right combination of knowledge and practice, to enable them to design, implement, manage and support solutions for an organisation's information technology infrastructure and systems and give quality service to the people who use it. Information Technology is the emphasis on the technology aspects driving a modern business.

Graduates of an Information Technology diploma qualification will address the gap between understanding the information needs of business and implementing and developing new algorithms by having an organisational infrastructure and integration focus.

The IT diploma BOK must be drafted by the members nominated by each institution to serve on the proposed SAITDAB. The SAITDAB, in collaboration with the SACAB standards committee, must draft proposed core and elective modules for the different IT diploma and advance diploma programmes offered by HEIs in South Africa. The SAITDAB must function within the requirements as specified by the Department of Higher Education and Training (DHET).

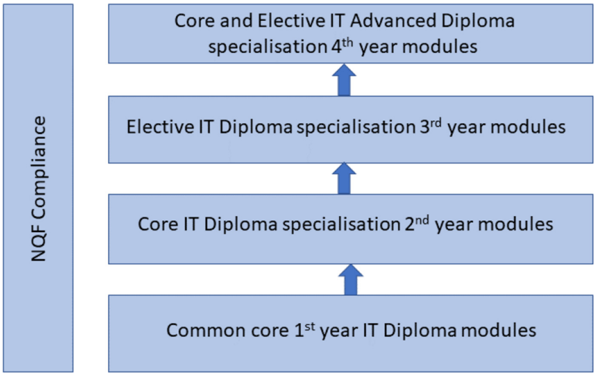


Fig. 3. Proposed IT diploma and advanced diploma programme structure

7 Conclusions

Non-governmental organisations, such as ABET [1] and national IT professional bodies, such as the BCS [5], ACS [3] and ITPNZ [16] accredit university IT related degree and diploma programmes. The accreditation of academic programmes is of critical importance to maintain the required academic standards and international recognition. The establishment of the SACAB [21] by the IITPSA [14] will provide internationally accepted degree and diploma programme accreditation in South Africa.

The research objectives of this study included determining the support for the proposed South African IT Diploma Advisory Board (SAITDAB) and proposing an initial mechanism for drafting and implementing an IT diploma and advanced diploma programme BOK for South African universities. In South Africa, four Comprehensive Universities, eight Universities of Technology and five private universities offer IT diploma and IT advanced diploma programmes. Ten IT diploma specialisation programmes and seven IT advanced diploma specialisation programmes were identified from the websites of the public and private HEIs.

The summary (Tables 4 and 5) of IT diploma programmes offered by HEIs indicates that there are 29 programmes offered by twelve institutions and 14 IT advanced diplomas. The respondents indicated that generally no collaboration between institutions is taking place to ensure standards and quality assurance. Most institutions indicated that quality assurance is obtained through external examination and moderation of exit level modules.

The results of the survey indicated full support by the respondents for the creation of the SAITDAB. The membership of the SAITDAB must include representatives from Comprehensive Universities (Table 1), Universities of Technology (Table 2) and private universities (Table 3), as such representatives would be the best equipped persons for this task. The SAITDAB must further establish administrative guidelines for IT diploma programmes and draft a core BOK for the different IT diploma programmes. Evaluation criteria (a standards document) for accreditation purposes must be compiled and published by the SACAB, based on the work done by the SAITDAB.

The limitations of this study were that not all universities offering IT diploma programmes in South Africa responded to the three calls for participation in the survey. Additionally, the private universities were not included in the survey. Further research will be required for establishing the SAITDAB, diploma and advanced diploma accreditation requirements and drafting of a document for the SACAB IT Diploma Programme BOK.

Appendix A: HOD IT Diploma Survey

Proposed IT Diploma Advisory Board for South African Universities

A South African Computing Accreditation Board (SACAB) workshop for universities offering IT Diploma programmes was held at the SACLA 2018 Conference, chaired by Dr Sue Petratos and Prof Andre Calitz on 20 July 2018. Representatives of six universities offering IT Diploma programmes that attended the meeting agreed on the establishment of a **South African IT Diploma Advisory Board (SAITDAB)**. Universities offering IT Diploma programmes were requested to inform Dr Petratos about programmes being offered. This survey is in response to the request made in 2018.

Institution Name: ______ Department name/School name: ______________
Respondent (title, name, surname): ______________________
Email address: __________________________

Q1. What IT diploma and IT higher diploma programmes does your institution offer?

____

Q2. How did your department/school decide on which IT diploma and IT advanced diploma programmes to offer?

____

Q3. How did your department/school decide on the structure and content of the IT diploma and IT higher diploma programmes that it offers?

____

Q4. Did your department/school collaborate or consult with any other institution when deciding on the structure and content of the IT diploma and IT higher diploma programmes that it offers? If so, what was the nature of the collaboration or consultation?

____

Q5. Do you intend to introduce new IT diploma and/or IT higher diploma programmes in the next two years? (Yes/No). If Yes, what programmes?

____

Q6. How does your department/school ensure that your IT diploma programmes remain up-to-date over time?

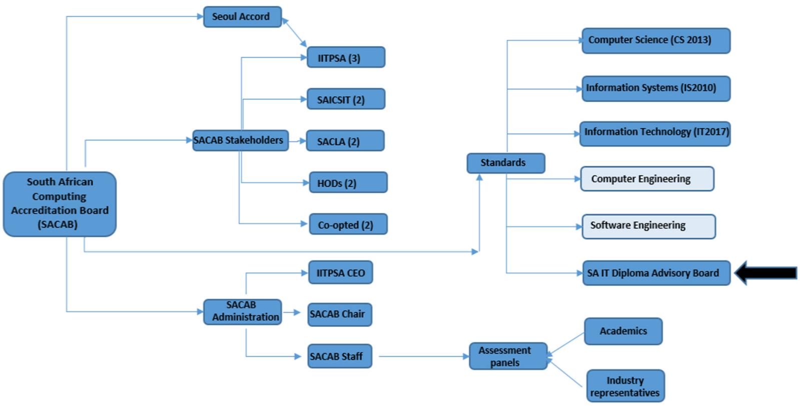
____

Q7. How does your department/school ensure the quality of the IT diploma programmes offered?

Q8. Does your department/school collaborate or consult with any other institution when ensuring that your IT diploma programmes remain up-to-date over time, and ensuring the quality of the IT diploma programmes offered. If so, what is the nature of the collaboration?

TECLA (Technikon Computer Lecturers Association) was an organisation that organised an annual conference for members of Technikons that offered IT diploma programmes.

Q9. Did your department/school support the old TECLA? Yes/No (Motivate your answer please)



SACAB is proposing the creation of a national *South African IT Diploma Advisory Board (SAITDAB)* for Comprehensive Universities and Universities of Technology offering IT diploma and IT higher diploma programmes. This body, as indicated in the figure above, would be re-sponsible for specifying the Body of Knowledge (BOK) for specific diploma programmes and quality assurance. The standards set by this body would be used for accreditation purposes by the SACAB.

Q10. Would your department/school support the proposed South African IT Diploma Advisory Board (SAITDAB)? Yes/No (Motivate your answer please)

Q11. Would your department/school support a national IT Diploma BOK as proposed by the SACAB? Yes/No (Motivate your answer please)

Q12. What benefits would an IT diploma programme BOK have for your department/school, in your opinion?

Q13. Provide the contact details of person(s) who could serve on the proposed South African IT Diploma Advisory Board (SAITDAB).

Q14. Do you have any other comments?

Thank you for your participation.

References

1. ABET: accreditation board for engineering and technology, Inc. https://www.abet.org/. Accessed 2021
2. ACM: Curricula recommendations. https://www.acm.org/education/curricula-recommendations. Accessed 2021
3. ACS: Australian computer society. https://www.acs.org.au/cpd-education/accredited-courses.html. Accessed 2021
4. Alexander, B.: A collaborative university of technology and further education and training curriculum development project. In: INEER ICEER Conference - Marrakesh: Association the Development of Education in Africa. Marrakesh, Morocco (2021)
5. BCS: British computer society - degree accreditation. https://www.bcs.org/deliver-and-teach-qualifications/university-accreditation/. Accessed 2021
6. Chiemezie, C.: 10 best private universities in South Africa 2021. https://worldscholarshipforum.com/best-private-universities-in-south-africa/. Accessed 2021
7. CIPS: Canada's association of information technology professionals. https://cips.ca/accreditation/. Accessed 2021
8. CIPS: Canada's association of I.T. professionals. Guide to the common body of knowledge for computing and IT (CBOK). https://cips.ca/CBOK/. Accessed 2021
9. CS2013: Curriculum guidelines for undergraduate programs in computer science. https://www.acm.org/binaries/content/assets/education/cs2013_web_final.pdf. Accessed 2021
10. ECSA: Scope of ECSA's accreditation function. https://www.ecsa.co.za/education/SitePages/Accreditation.aspx. Accessed 2020

11. EFICTBOK: The European foundation ICT body of knowledge. https://ec.europa.eu/. Accessed 2020
12. HEQSF: Higher education qualification sub framework. https://www.dhet.gov.za/ (2020). Accessed 2020
13. IEEE: The institute of electrical and electronics engineers (2020). https://www.ieee.org/. Accessed 2020
14. IITPSA: Institute of information technology professionals South Africa. https://www.iitpsa.org.za/. Accessed 2020
15. IS2010: Curriculum guidelines for undergraduate degree programs in information systems. https://www.acm.org/binaries/content/assets/education/curricula-recommendations/is-2010-acm-final.pdf. Accessed 2020
16. IT Professionals New Zealand: Degree accreditation. https://itp.nz/Activities/Degree-Accreditation. Accessed 2021
17. IT2017: Curriculum guidelines for baccalaureate degree programs in information technology. https://www.acm.org/binaries/content/assets/education/curricula-recommendations/it2017.pdf. Accessed 2020
18. JABEE: Japan accreditation board for engineering education. https://jabee.org/en/. Accessed 2021
19. NMU: Diploma programmes. https://www.mandela.ac.za/Apply/Frequently-asked-questions/Study-@-NMMU/What-is-the-difference-between-a-diploma-and-a-deg#FAQLink1789. Accessed 2020
20. SA HEI: South African higher education institutions. https://www.southafricaeducation.info/higher-education/list-of-higher-education-institutions-in-south-africa.html. Accessed 2021
21. SACAB: South African computing accreditation board. https://www.sacab.org.za/. Accessed 2020
22. SACLA: Southern African computer lecturer's association. http://www.sacla.org.za/. Accessed 2020
23. The Seoul Accord: The Seoul accord signatories. https://www.seoulaccord.org/. Accessed 2021
24. Von Solms, R.: Role of TECLA and HEICTA. Personal communication (2020)

Goofs in the Class: Students' Errors and Misconceptions When Learning Regular Expressions

Olaperi Yeside Okuboyejo[1(✉)], Sigrid Ewert[1], and Ian Sanders[2]

[1] University of the Witwatersrand, Johannesburg, South Africa
{olaperi.okuboyejo,sigrid.ewert}@wits.ac.za
[2] University of South Africa, Science Campus, Florida Park, Roodepoort, South Africa
sandeid@unisa.ac.za

Abstract. The knowledge of students' errors and misconceptions is important because it helps instructors to understand the difficulties students experience when learning. This knowledge is also beneficial during the automation of the teaching process. Although studies on errors and misconceptions have been reported for several computer science courses, there is a gap in respect of Regular Expressions (REs), one of the topics taught in Formal Languages and Automata Theory. Regular Expressions are a vital part of the computer science curriculum and very useful in the software industry. Students, however, find REs difficult to learn and there is a need to understand the types of errors they make in order to build an intelligent tutoring system. Therefore, this research investigated the errors students make and misconceptions they can have when learning REs. A total of 393 students' solutions to six RE questions were qualitatively analysed. Errors in students' submissions can be syntax errors, slight errors or logical errors, while misconceptions include misunderstanding of the empty string and confusion of the Kleene star operator with the Kleene plus. The identification of these errors and the associated misconceptions will guide in automatic error detection and feedback generation on e-learning platforms and mobile devices on which students can practise on-the-go and get immediate feedback. The findings can also be used to adjust the teaching process in traditional classrooms to improve learning.

Keywords: Regular expressions · Errors · Misconceptions · Computer science education · Theory of computation

1 Introduction

Terms such as difficulties, misconceptions, errors, bugs, and mistakes are often used interchangeably to describe students' inaccurate or incomplete understanding. For this research, the intended meanings of the terms are given thus:

G. Wells et al. (Eds.): SACLA 2020, CCIS 1518, pp. 57–71, 2021.
https://doi.org/10.1007/978-3-030-92858-2_4

Definition 1. ***Slips:*** *They are incorrect answers given carelessly and are not usually repeated. Slips can be made by both novices and experts, however, they are easily detected, and can be quickly corrected [10].*

Definition 2. ***Misconceptions:*** *They are faulty underlying conceptual structures in the mind of a student which generate observed errors [10]. They are erroneous understandings in students' knowledge and are sometimes referred to as alternate conceptions [7].*

Definition 3. ***Errors:*** *They are incorrect answers given as a result of faulty underlying conceptual structures (misconceptions), which are regularly applied or repeated under similar circumstances [10].*

Misconceptions impede learning because they are ingrained in students' thinking but the students are unaware that the knowledge they have is faulty [7]. Misconceptions lead to errors and errors lead to low grades. Identification and understanding of students' misconceptions enhance the effectiveness of instructors [25], lead to effective teaching [22] and 'larger classroom gains' [16].

In the field of computer science, students' errors, difficulties and misconceptions have been studied across several subject areas including: Formal Languages and Automata Theory (FLAT) [14], programming [6,15], data structures [25], operating systems [12], digital logic [3], and dynamic programming [24].

However, studies specifically on topics of FLAT are few, and to the best of the researchers' knowledge, there has been no study specifically on students' errors and misconceptions in Regular Expressions (REs).

This work, therefore, aims to investigate the errors students make and the underlying misconceptions they have when learning REs. This is for the long-term goal of assisting learners outside the traditional classroom and in the absence of direct access to human tutors to guide them when faced with difficulties. The understanding of these errors and misconceptions will be used to build an automatic error detection and feedback generation system, which will be very useful to the new generation of learners today.

The remainder of this paper is structured as follows: Sect. 2 explains REs and related concepts while Sect. 3 discusses related works that have investigated students' misconceptions in FLAT. Details of the methodology, research questions, data and ethics approval are presented in Sect. 4. Section 5 discusses the errors and misconceptions that were identified, Sect. 6 presents the discussion and gives some recommendations for teaching REs, and the paper rounds off with the conclusion in Sect. 7.

2 Regular Expressions

Regular Expressions date back to the 1950s when Stephen Cole Kleene described regular languages [5] and till today, they are taught globally and have a wide range of application. REs are used in lexical analyzer generators and tools/utilities of the Unix operating system. They are also used for pattern matching, data validation, text replacement and reformatting in different software applications [4,18,20].

The formal definition of an RE and definitions of other related terms used in this research are presented next. Also, concepts taught to students when learning REs are discussed.

Definition 4. *An **alphabet** refers to a finite set of symbols and is denoted by the Greek symbol Σ [8].*

Definition 5. *A **string** over an alphabet Σ is a finite sequence of symbols in Σ, including zero symbols. For a string x, $|x|$ stands for the length (the number of symbols) of x [8].*

Definition 6. *The **exponential notation** is used to represent the concatenation of k copies of a single symbol a. If $k > 0$, then $a^k = aa \dots a$, where there are k occurrences of a [8].*

Definition 7. *A **regular expression** R over an alphabet Σ is [18]:*

1. *a, for some a in the alphabet Σ*
2. *λ, the empty string*
3. *$\emptyset$, the empty set*
4. *$(R_1 + R_2)$, where R_1 and R_2 are REs*
5. *$(R_1 R_2)$, where R_1 and R_2 are REs, or*
6. *(R_1^*), where R_1 is an RE.*

The RE a represents the language $\{a\}$ while the RE λ represents the language $\{\lambda\}$, containing a single string i.e., the empty string. The RE $\emptyset$ represents the empty language i.e., the language that contains no strings. The language obtained from the union of R_1 and R_2 is represented by $(R_1 + R_2)$, $(R_1 R_2)$ represents the concatenation of R_1 and R_2, and (R_1^*) the star of language R_1.

Concepts in Regular Expressions. Several textbooks are used to support classroom teaching of REs. From a close look at these textbooks [1,4,8,18], major concepts taught in the construction of REs include:

1. The operators:
   - the concatenation - '.' (or no symbol),
   - the Kleene star - '*',
   - the Kleene plus - '$^+$', and
   - the union - '+'.
2. Precedence of the operators (proper use of brackets).
3. The notions of the empty string and the empty language.
4. The alphabet over which an RE is defined.
5. Writing REs to handle constraints/restrictions specified, such as:
   - the length of the string,
   - the position of symbols of the given alphabet,
   - symbol(s) that should not be present.
6. The exponential notation.

3 Related Works

In FLAT topics, research has been carried out on studying students' difficulties, though not as much as in the programming domain.

Smith and McCartney [19], identified errors made by students when applying the pumping lemma for regular languages. Final examination documents which had a question on the pumping lemma and students' submissions to other class activities during the teaching of the course were reviewed. The incorrect solutions of the students were the primary focus of the researchers. Errors made were identified and categorized. However, the focus here was only on the pumping lemma for regular languages.

Errors made by students when designing finite automata to recognize given regular languages have also been studied by Sanders et al. [17]. Students were given design exercises to solve and submit via an on-line platform. Solutions to the given exercises were assessed and several mistakes were identified.

A research work that partly studied REs was carried out by Pillay in 2010 [14]. The researcher studied the learning difficulties experienced by students in a course on FLAT. Specifically, four topics in the course were considered, namely: regular languages, transducers, context-free languages and Turing machines. Students' solutions to assessment tests and weekly tutorials were analyzed to identify these difficulties. No difficulties were discovered regarding converting REs to Non-deterministic Finite Automata (NFAs), however, students made errors in applying the algorithm to convert NFAs to REs. Some students omitted components of REs, while others used the concatenation instead of the union operator and vice-versa. On the construction of REs, the author only observed that students' solutions contained logical errors, which was attributed to a lack of problem-solving skills. Specific errors made or difficulties identified by the students in formulating the REs were not identified, and underlying misconceptions were not investigated.

As seen from the literature, research that studies specifically the errors that students make and misconceptions they have when learning REs has not been reported to the best of the researchers' knowledge. As such, this is what this study investigated.

4 Methodology

The methodology for this research followed the nomenclature and approach proposed by Pilkington and Pretorius [13], where any research has an underlying philosophical worldview (whether explicitly stated or not), which determines and influences the research design and subsequently, the research methods.

We adopted the interpretivist philosophical worldview which guided the overall approach employed. The term interpretivism refers to 'epistemologies or theories about how we can gain knowledge of the world, which loosely rely on interpreting or understanding the meanings that humans attach to their actions' [11, p. 119]. The interpretivist inquiry is subjective [9], and its purpose is the understanding of a particular situation [23].

In this research, we have reviewed answers given to RE questions from the perspective of the students in order to interpret their prevalent misunderstandings. This is in line with interpretivism, which allows researchers to view the world from the perspective of the participants and use those experiences to construct and interpret their (the researchers') understanding from the gathered data [18]. It is important to note that the analysis and deductions have been made based on the researchers' assessments and views and might differ in some way from other researchers'. However, the researchers were careful to explain the rationale for their interpretations, by clearly giving examples.

Furthermore, the design adopted for the research is empirical since it required systematic collection and analysis of data [2]. Specifically, the case-study design was used. According to Merriam (1988), as cited by Willis in [23], a case study is 'an examination of a specific phenomenon such as a program, an event, a person, a process, an institution, or a social group'. In this case, data was gathered from past questions of students at the University of the Witwatersrand, Johannesburg (Wits University). Case studies are usually about real people and situations [23], as such the data collected were from real students in examination settings.

The research method, i.e., instrument/tool used to gather and analyze the data was document review and analysis, and the data source was past examination scripts of students who were taught REs in a traditional classroom setting.

The remainder of this section details the research questions, ethics clearance and data collection process. It explains the data, describing the examination questions and model solutions.

4.1 Research Questions

The questions this research sought to answer are:

1. How can we classify errors students make when learning to construct REs?
2. What misconceptions do the students acquire during learning that leads to the errors?

Qualitative analysis of the research data was engaged to answer these questions. The data is explained next, while the answers to the research questions are discussed in Sect. 5.

4.2 Ethical Consideration

Ethical clearance was obtained to review students' past scripts which are confidential. The clearance was approved on the 20th July 2018, on the agreement that anonymity of records is ensured. Subsequently, approval was also acquired from the University's Registrar and the Head of School. It was only after all the necessary approvals were obtained, that the data collection commenced.

4.3 The Data

The data used for this research was obtained from previously answered exam questions on REs by third-year computer science students at Wits University. REs are taught to computer science majors at Wits University in their third year. In the final examinations for the course, there is always a question that tests students' knowledge of REs. These solutions were obtained from students' solutions to six RE questions from 2015 to 2018. Years 2016 and 2017 had two RE questions while 2015 had one. The 2018 examination had two variants of a single question (due to deferred examinations), however, only one of the variants is presented as they are very similar.

A total of 393 students' solutions were initially reviewed. After the first assessment, there were 150 correct solutions, 228 incorrect solutions and 15 blank answers. The incorrect solutions were then further analyzed to answer the research questions.

The Questions. The RE questions the students solved and their respective model solutions are presented next.

Question 1. (2015) Give a regular expression for the following language:
$L = \{w \in \{a, b\}^* \mid |w| \geq 4\}$ and the second symbols from the front and the back are equal.

Solution 1. $((a + b)a(a + b)^*a(a + b)) + ((a + b)b(a + b)^*b(a + b))$

Question 2. (2016a) Give a regular expression for the following language:
$L = \{a^n \mid n \geq 0, n \neq 3\}$

Solution 2. $\lambda + a + aa + aaaaa^*$

Question 3. (2016b) Give a regular expression for the following language:
$L = \{w \in \{a, b\}^* \mid |w| \geq 4$ and the fourth symbol from the back is a $b\}$.

Solution 3. $(a + b)^*b(a + b)(a + b)(a + b)$.

Question 4. (2017a) Give an RE for the following language:
$L = \{a^n \mid n \geq 2, n \neq 5, n \neq 7\}$

Solution 4. $a^2 + a^3 + a^4 + a^6 + a^8a^*$

Question 5. (2017b) Give an RE for the following language:
$L = \{w \in \{a, b\}^* \mid |w| \geq 2$ and every second symbol is a $b\}$

Solution 5. $((a + b)b)^+(a + b + \lambda)$

Question 6. (2018) At Wits University, the name of a research account consists of the following components in the given order:

1. surname: four letters, each letter one of $\{A, B, \ldots, Z\}$
2. hyphen, i.e., '-'
3. funder code: one of NRF, CSIR, DST
4. semi-colon, i.e., ';'
5. calendar year: four digits, each digit one of $\{0, 1, \ldots, 9\}$.

Give an RE for all possible research account names for the years 1998 to 2013.

Solution 6. Let $\ell = A + B + C + \ldots + Z$.
$\ell^4 - (\text{NRF} + \text{CSIR} + \text{DST}); (199(8+9) + 200(0+1\ldots9) + 201(0+1+2+3))$

5 Results: Errors and Misconceptions

This section answers the research questions, highlighting the errors and misconceptions identified.

5.1 Errors Identified

All the students' solutions were classified into the appropriate error category and Table 1 shows the details of the classification. The following are the categories of errors identified:

1. Syntax Errors
2. Slight Errors
3. Logic Errors
   (a) Additional Restriction Error (does not represent some valid strings)
   (b) Omitted Restriction Error (represents invalid strings)
   (c) Incorrect Restriction Error (does not represent some valid strings and incorrectly represents invalid strings)

Table 1. Error classification summary

Type of error	Q1	Q2	Q3	Q4	Q5	Q6	Total
Correct solution (no error)	22	27	32	34	3	32	150
Syntax error	5	6	9	5	6	9	40
Slight error	7	4	0	5	4	0	20
Logical error - additional restriction	4	4	8	6	27	9	58
Logical error - omitted restriction	4	10	5	13	17	24	73
Logical error - incorrect restriction	9	6	6	4	12	0	37
No answer given	2	3	0	3	1	6	15
	53	**60**	**60**	**70**	**70**	**80**	**393**

The classes of errors are explained with examples. To ensure clarity and simplicity, examples are presented from Question 4 and Question 5.

Syntax Errors. These errors are made when students give solutions that do not conform to the standard rules of writing REs, such as starting with the incorrect symbol or operator; ending with the incorrect symbol or operator; using an invalid symbol; having mismatched parentheses; using the incorrect type of bracket delimiters. In REs where a syntax error is present, it is often not the only error present, i.e., apart from the syntax error, other errors might also be present. Examples of syntax errors in students' solutions to Question 4 include:

1. A solution ending with an incorrect symbol.
   Example: $(aa)^{+}+$
   Here, a union operator has been used to end the RE which is invalid, as every union operator must have an operand on its right-hand side.
2. Using invalid symbols.
   Example: $(a^n)^*\backslash((a^5) \cup (a^7))$
   Several invalid symbols such as the letter 'n' and symbol '\' have been used. Also, the exponent operator has been used with an invalid operand.

Slight Errors. It was observed that some students make errors that seem accidental. Such errors are due to slips. Instances of these errors include using a Kleene star instead of a Kleene plus and vice-versa; writing a letter of the alphabet in place of another; using a concatenation operator in the place of a union operator and vice-versa. For this work, we assume that a slip is an error that can be corrected by a single edit. This implies that if there are multiple errors in an RE, such as several operators being misused, it cannot be classified as a slight error. Examples of slight errors in students' solutions to Question 4 include:

1. Using the union (+), instead of concatenation (.). This is in line with a similar finding reported in [14].
   Example: $a^2 + a^3 + a^4 + a^6 + a^8 + (a)^*$
2. Using the Kleene star (*) instead of Kleene plus ($^+$).
   Example: $(a^2 + a^3 + a^4 + a^6 + a^7.a^*)$
3. Using an incorrect bracket delimiter.
   Example: $[(a^2)(a^2)^* + (a^3) + (a^9)(a^2)^*]$
   In this solution, square brackets have been used instead of parentheses. However, the solution in itself is correct, and since a single edit of changing the brackets used will make the solution correct, it is classified as a slight error.

Logical Errors. Often, students write REs that are valid syntactically, but do not fully cater for the requirements or restrictions in the given question. They could be including additional restrictions (either implicitly or explicitly), thereby not representing some valid strings, this is called the **additional restriction error**. They also could be leaving out some restrictions, thereby representing invalid strings, this is called the **omitted restriction error**. In some situations, they include extra restrictions not required and leave out necessary restrictions, this is called the **incorrect restriction error**, implying that the requirements have not been properly handled at all. Instances of these errors in students' solutions to Question 4 and Question 5 are presented.

1. **Additional restriction error** (does not represent some valid strings):
   (a) The length of an RE can be restricted, such that the RE does not represent all possible word lengths.
   Example from Question 4: $a^2(a + a^2 + a^4 + a^6a^*)$
   This solution does not represent a string length of two. The string aa is a valid string based on the question, but the RE does not represent it.
   (b) The words that can be generated from the RE can only start with one letter (or a subset of letters) from the given alphabet, i.e., words cannot be formed, starting with all possible symbols.
   Example from Question 5: ab^+
   The strings generated from this RE cannot start with the letter b.
   (c) The words that can be generated from the RE cannot end with all possible letters or combinations of letters from the given alphabet.
   Example from Question 5: $((a + b)b)^+(\lambda + a)$
   In some situations, strings generated from the RE cannot end with double letter b, e.g. the RE does not represent abb.
2. **Omitted restriction error** (the RE represents invalid strings): Here, the RE does not specify all required restrictions as stated (or implied) in the question, thus representing invalid strings.
   Example from Question 5: $(a + b)b(a + b)^*$
   The restriction needed to ensure that every second symbol is a letter b is missing. The RE incorrectly represents $abaa$.
   Example from Question 4: $(aa + aaa + aaaa)^*$
   A restriction to ensure that the length of the string is longer than or equal to two is missing. The RE incorrectly represents the empty string.
3. **Incorrect restriction error** (does not represent some valid strings and represents invalid strings): Here the restrictions specified in the questions have been violated from both ends. Unspecified restrictions have been included and required restrictions have been omitted, therefore the RE both generates invalid strings and does not represent some valid strings. Examples from Question 4 are:
   (a) $(a^2(a^2)^* + a^3 + a^4 + a^6 + a^8) + (a^9 + \lambda)^*$ does not represent a^{11}; incorrectly represents λ
   (b) $a^2 + a^4 + a^+a^3$ does not represent a^3; incorrectly represents a^5
   (c) $(a^2 + a^3 + a^4) + (a^6a^+)$ does not represent a^6; incorrectly represents a^7
   (d) $(aaa^* + aa)(aaa^* + aa)$ does not represent a^2; incorrectly represents a^5

5.2 Misconceptions Identified

This section answers our second research question: What misconceptions did the students acquire during learning that led to the errors?

Having reviewed the errors, some underlying misconceptions were identified. It is assumed that misconceptions on some of the concepts of REs are the root cause of the errors observed. Since the source of the data was past examination scripts which were reviewed in the absence of the students, these misconceptions are the findings of the researchers based on the critical analysis of the incorrect solutions.

The Need to Handle only the Explicitly Stated Requirements. The students seem to have the wrong notion that they only need to handle explicitly stated requirements. They do not try to think of other requirements that have not been stated which must be catered for.

For example, in Question 5, some students gave REs that allow for every second symbol to be the letter b (which is the requirement stated), however, the RE they gave can only accommodate strings with even length (which should not be the case). A situation in which a single letter a or b can end the RE is not catered for. Examples of such solutions are $((a+b)b)^+$ given by eleven students; $((a+b)b)$ $((a+b)b)^*$ given by five students, $(ab+bb)^+$ given by three students, and $(a+b)b+((a+b)b)^*$ given by a student.

Use of the Empty String. It was observed that students probably do not understand when to use the empty string. Sometimes, the empty string is necessary to cater for strings that might not contain an optional symbol, but they left it out. Question 5 required proper use of the empty string (based on the approach the students decided to take in answering the question), however, out of the sixty-nine solutions given, only four used the empty string correctly.

For example, Question 5 was answered thus by a student: $((a+b)b)^+(a+b)$; this means the strings represented by the RE must always end with an additional a or b, which will not always be the case. The addition of the empty string, to give $((a+b)b)^+(a+b+\lambda)$, would make the solution correct.

Misconception on the Kleene Star. Some students seem not to understand that when using the Kleene star (*) it is possible that no symbol is selected. For example, they have used the Kleene star where the minimum length of string to be allowed is two.

A sample solution to Question 5 reveals this: $((a+b)b)^*$. Though the solution will remain incomplete regardless of the Kleene star, the usage of the Kleene star here shows that the student does not realise that it is possible that no symbol is selected because the question requires a string with the minimum length of two. Another example seen in Question 4 is $(a^2+a^3+a^4+a^6+a^7.a^*)$. This solution allows the string a^7 which it should not; this could have been avoided if a Kleene plus was used instead.

6 Discussion and Recommendations

6.1 Discussion

As seen from Table 1, only 38% i.e. 150 of the 393 solutions were correct. This validates that students find REs difficult.

The least type of errors present were slight errors, just 9% of the total errors. This is very much expected since they are 'accidental' errors or 'slips' on the part of the student.

The next category is the syntax errors, accounting for 17%. This includes scenarios where the student enters an invalid letter of the alphabet or an invalid symbol or violates rules of writing REs.

The highest category of errors with 74% is the logical error. The total number of logical errors was 168 comprising 58 additional restriction errors, 73 omitted restriction errors and 37 incorrect restriction errors.

Across the years examined, 2015 (Question 1), 2016 (Questions 2 and 3), 2017 (Questions 4 and 5), and 2018 (Question 6), the percentage of correct solutions ranges from 40% to 53.3%, except for Question 5 in 2017 which was 4.2%.

In a bid to understand the low success rate for Question 5, further investigation was done and we discuss the findings.

To start with, the low success rate cannot be attributed to a peculiarity in the teaching process that year for two reasons. First, the lecturer was the same across the years. Second, that year (with the lowest success rate, 2017) had two questions of which the second question, Question 4 recorded the second highest success rate of 48.5% amongst the six questions examined in this research.

Furthermore, a review of the six questions shows that Question 5 had some implicit (or hidden) requirements (such as, ensuring strings can end with the letter a and can be of odd length) which require a thoughtful use of the empty string as in $((a+b)b)^+(a+b+\lambda)$ or the Kleene star, as in $((a+b)b)^+(a+b^*)$. On the contrary, a number of the REs given by the students could only represent strings of even length and strings that cannot end with the letter a which further shows a lack of understanding of the empty string. Besides, of the six questions, only Question 5 required a possible use of the empty string and this was present in the three correct students' solutions. The correct solutions were: $(b+a)b((ab+(bb)^+)^*)(a+b^*+\lambda)$, $((a+b)b)^+(a+b+\lambda)$ and $(ab+bb)^+(a+b+\lambda)$.

On the other hand, verbal discussions with some of the students by the course lecturer after the examination indicated a possible misunderstanding of the question. They thought the phrase: 'every second symbol is a b' meant *the second symbol of every string must be a 'b' and the rest of the string did not matter*. This misunderstanding might also have been responsible for the low success rate on the question since thirteen students solved the question this way by giving the solution $(a+b)b(a+b)^*$ or an equivalent variant.

6.2 Recommendations

The number of logical errors present shows that instructors need to assist students to reason deeply when solving RE questions. As such, the following recommendations are made.

Focused Practice Exercises: Having identified concepts and areas that students struggle with such as empty strings and handling implicit requirements, it will be beneficial if instructors construct exercises that help to emphasize and harness these concepts.

Emphasize a Broad Perspective: In the course of lectures, instructors and tutors can emphasize how to think about REs from a broad point of view, where instead of focusing only on what is explicitly stated in the question, students are groomed to also think of other requirements that are depended upon indirectly. Exercises that will enable this thought process can be carefully designed to assist the students to 'think outside the box'.

Real-Life Examples: Students seem to be used to writing REs to represent random strings of symbols that do not necessarily have meaning in the real world, i.e., they are not valid English words, numbers or strings that represent valid information such as an e-mail address. Instructors might want to consider giving more examples of REs that create valid words or symbols. This will help represent the use of REs in the real world. In Question 6, most students' solutions were incorrect because they tried to represent a year as a random string of symbols and did not consider the year range given. This led to REs that allow years such as 9999, 2999, or even 0000, whereas the range given was 1998–2013.

Special Attention to Some Operators: The concept of the empty string may need to be taught with a lot of thought on its use and application areas, while emphasis should be placed on the difference between the use of the curly braces '{ }' and the parentheses '( )'. Students should also be informed that the comma can be used only when representing a set of symbols, and not when writing an RE. The difference between the Kleene star and Kleene plus should be stressed.

Learning from the Errors: Students can learn from the error categories and actual errors made by other students in different ways. The details of the error categories and their respective examples can be presented to students learning REs. This knowledge can guide them in assessing their solutions and prevent them from making the same errors.

Moreover, lecturers can make a compilation of incorrect solutions available to students either during lectures or tutorial sessions. By trying to identify the errors and misconceptions present in incorrect solutions, the students will sharpen their skills and be wary of such. They will also be aware of the misconceptions to watch out for while they are learning. In addition, as a variant of assessment and tutorial questions, partially incorrect solutions can be presented to students to fix.

7 Conclusion

This research sought to discover errors and underlying misconceptions that students have when learning REs in an attempt to help students and improve teaching. A total of 393 students' solutions to RE exercises were qualitatively reviewed. Amongst other findings, it was observed that students make three types of errors: syntax errors, slight errors and logic errors. Underlying misconceptions responsible for these errors were also discussed.

The errors and misconceptions elicited throw light into the challenges students face when learning the topic which is taught globally and used heavily in the software industry. The information obtained is useful to lecturers, instructors and tutors because if carefully considered, it would help to improve future delivery of the course. It would assist in areas where emphasis needs to be placed, as such enhancing students' understanding and performance.

Furthermore, considering the new generation classrooms and increase in the adoption of e-learning, massive open online courses (MOOCs) and distance learning, the ultimate goal of this research is to utilize these findings in the development of an intelligent tutoring system that will automatically detect errors and give feedback to learners directly on their mobile devices, anywhere and anytime.

Acknowledgments. This research was supported by the L'Oréal-UNESCO *For Women in Science* Sub-Saharan Africa Programme and the Postgraduate Merit Award of the University of the Witwatersrand, Johannesburg. The authors would like to appreciate the effort and support of the administrative officer of the School of Computer Science and Applied Mathematics, University of the Witwatersrand, Johannesburg for her assistance during the data collection process. The authors also acknowledge the helpful comments given by the reviewers.

References

1. Cohen, D.I.A.: Introduction to Computer Theory, 2nd edn. Wiley, New York (1996)
2. Dan, V.: Empirical and Nonempirical Methods, pp. 1–3. American Cancer Society (2017). https://doi.org/10.1002/9781118901731.iecrm0083
3. Herman, G.L., Loui, M.C., Kaczmarczyk, L., Zilles, C.: Describing the what and why of students' difficulties in Boolean logic. ACM Trans. Comput. Educ. **12**(1), 3:1–3:28 (2012). https://doi.org/10.1145/2133797.2133800
4. Hopcroft, J.E., Motwani, R., Ullman, J.D.: Introduction to Automata Theory, Languages, and Computation, 3rd edn. Pearson Education Inc., Boston (2007)
5. Kleene, S.C.: Representation of events in nerve nets and finite automata. Technical report, US Air Force, Project RAND (1951)
6. Kohn, T.: Variable evaluation: an exploration of novice programmers' understanding and common misconceptions. In: Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education, SIGCSE 2017, pp. 345–350. ACM, New York (2017). https://doi.org/10.1145/3017680.3017724

7. Lucariello, J., Naff, D.: How do I get my students over their alternative conceptions (misconceptions) for learning? (2013). https://www.apa.org/education/k12/misconceptions.aspx. Accessed 17 Jan 2019
8. Martin, J.C.: Introduction to Languages and the Theory of Computation, 4th edn. McGraw-Hill, New York (2011)
9. Mathison, S.: Encyclopedia of Evaluation. SAGE Publications Inc., Thousand Oaks (2005). https://doi.org/10.4135/9781412950558
10. Olivier, A.: Handling pupils' misconceptions. Pythagoras **21**, 9–19 (1989)
11. O'Reilly, K.: Key Concepts in Ethnography. SAGE Publications Ltd., London (2009). http://methods.sagepub.com/book/key-concepts-in-ethnography. https://doi.org/10.4135/9781446268308
12. Pamplona, S., Medinilla, N., Flores, P.: Exploring misconceptions of operating systems in an online course. In: Proceedings of the 13th Koli Calling International Conference on Computing Education Research, Koli Calling 2013, pp. 77–86. ACM, New York (2013). https://doi.org/10.1145/2526968.2526977
13. Pilkington, C., Pretorius, L.: A conceptual model of the research methodology domain - with a focus on computing fields of study. In: 7th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management, IC3K 2015, vol. 2, pp. 96–107. SciTePress (2015)
14. Pillay, N.: Learning difficulties experienced by students in a course on formal languages and automata theory. ACM SIGCSE Bull. **41**(4), 48–52 (2010). https://doi.org/10.1145/1709424.1709444
15. Qian, Y., Lehman, J.: Students' misconceptions and other difficulties in introductory programming: a literature review. ACM Trans. Comput. Educ. **18**(1), 1:1–1:24 (2017). https://doi.org/10.1145/3077618
16. Sadler, P.M., Sonnert, G., Coyle, H.P., Cook-Smith, N., Miller, J.L.: The influence of teachers' knowledge on student learning in middle school physical science classrooms. Am. Educ. Res. J. **50**(5), 1020–1049 (2013). https://doi.org/10.3102/0002831213477680
17. Sanders, I., Pilkington, C., van Staden, W.: Errors made by students when designing finite automata. In: Coleman, E. (ed.) 44th Annual Southern African Computer Lecturers' Association 2015, vol. 27, pp. 110–116. University of the Witwatersrand, Johannesburg (2015)
18. Sipser, M.: Introduction to the Theory of Computation, 3rd edn. Cengage Learning, Boston (2013)
19. Smith, T., McCartney, R.: Mathematization in teaching pumping lemmas. In: 2013 IEEE Frontiers in Education Conference (FIE), pp. 1671–1677, October 2013. https://doi.org/10.1109/FIE.2013.6685122
20. Stubblebine, T.: Regular Expression Pocket Reference, 2nd edn. O'Reilly Media Inc. (2007)
21. Thanh, N.C., Thanh, T.T.: The interconnection between interpretivist paradigm and qualitative methods in education. Am. J. Educ. Sci. **1**(2), 24–27 (2015)
22. Vallejos, S., Berdun, L.S., Armentano, M.G., Soria, Á., Teyseyre, A.R.: Soploon: a virtual assistant to help teachers to detect object-oriented errors in students' source codes. Comput. Appl. Eng. Educ. **26**(5), 1279–1292 (2018). https://doi.org/10.1002/cae.22021
23. Willis, J.: Foundations of Qualitative Research: Interpretive and Critical Approaches. SAGE Publications Inc., Thousand Oaks (2007). http://methods.sagepub.com/book/foundations-of-qualitative-research. https://doi.org/10.4135/9781452230108

24. Zehra, S., Ramanathan, A., Zhang, L.Y., Zingaro, D.: Student misconceptions of dynamic programming. In: Proceedings of the 49th ACM Technical Symposium on Computer Science Education, SIGCSE 2018, pp. 556–561. ACM, New York (2018). https://doi.org/10.1145/3159450.3159528
25. Zingaro, D., et al.: Identifying student difficulties with basic data structures. In: 2018 International Computing Education Research Conference, ICER 2018. ACM, New York (2018). https://doi.org/10.1145/3230977.3231005

Teaching Problem Solving to Undergraduate STEM Students: A Systematic Literature Review

Nandipha Dilla(✉) and Marita Turpin(✉)

Department of Informatics, University of Pretoria, Pretoria, South Africa
marita.turpin@up.ac.za

Abstract. This systematic literature review aims to explore the field of undergraduate teaching using the Problem Based Learning (PBL) methodology and to discover the ways that researchers have used the PBL methodology as a means to teach problem solving. This is a review of 20 primary studies on teaching problem solving by means of PBL to undergraduate Science, Technology, Engineering and Mathematics (STEM) students. Through a thematic analysis of the 20 papers, four central themes emerged as being significant to PBL regardless of the subject discipline. These themes are: the role of prior knowledge; teamwork and team composition; the role of the instructor; and contextual learning methods. In terms of prior knowledge, it was found that students need to have sufficiently mastered the subject's domain knowledge at an individual level before they can benefit from PBL. The teamwork theme emphasized the importance of teams in problem solving. Small heterogeneous teams appear to be conducive to effective teaching by means of PBL. In terms of the role of the instructor, it was found that teams significantly benefit in their learning experience, from an instructor or tutor who facilitates and motivates the team as opposed to telling them what to do. Lastly, most of the PBL studies employ learning methods that are specific to their domain but will help the students better solve the problem. Based on the insights gained from the literature, the review suggests good practices that can be used by other researchers when trying to implement the PBL methodology in their undergraduate STEM classes.

Keywords: Problem solving · Problem based learning · STEM · Undergraduate students · Systematic literature review

1 Introduction

The skill of practical problem solving is integral to the work environment and is considered to be indispensable [1]. The 1993 National Adult Literacy Survey (NALS) found that basic problem solving skills are in short supply in the workplace, with more than half of employed adults having difficulty solving even simple assigned tasks [2]. Employers have noted that they actively seek graduates

G. Wells et al. (Eds.): SACLA 2020, CCIS 1518, pp. 72–87, 2021.
https://doi.org/10.1007/978-3-030-92858-2_5

who have shown to engage in problem solving [3]. The ability to solve complex real-world problems is particularly important in the STEM disciplines [4]. Unfortunately, problem solving is not taught explicitly at most universities, with subjects being mainly taught at a theoretical level with little focus spent on teaching the subject's practical value [1]. Students need to be provided with opportunities to build the skills and abilities that are necessary to solve the various types of problems that can occur in the field that they work in.

Problem-based learning (PBL) is a teaching method that requires students to solve problems relevant to their discipline in an active learning environment [5]. It was originally developed in the medical domain but subsequently proved to be a valuable means to teach problem solving in the STEM disciplines [4,5].

This systematic literature review seeks to collect research evidence on the use of PBL in undergraduate STEM fields in order to identify how PBL is used to teach problem solving. Using the systematic approach to evaluate existing literature helps us to more dependably find patterns that exist in the studies, or even review any conflicts in the existing body of work.

This paper is organised as follows. The next section provides literature background, in particular about problem solving and PBL in undergraduate teaching. Following this, the research method is presented. An analysis of the structured literature review's key findings is subsequently presented, starting with descriptive statistics before proceeding with a thematic analysis. The paper ends with a reflection of findings in the Conclusion.

2 Literature Background

This section first considers the notion of problem solving before proceeding to PBL as a learning strategy that is centered on problem solving.

2.1 Problem Solving

Problem solving is the process of finding a solution when the path to the solution is uncertain, according to Martinez [6]. Martinez further states that "problem solving involves an interaction of a person's experience and demands of the task" [6], p. 606. Universities can teach problem solving through the use of problem solving methods and strategies in order to help develop students' cognitive abilities. The generic problem solving process as defined by Kimbell, Green and Stables [7] is shown in Fig. 1:

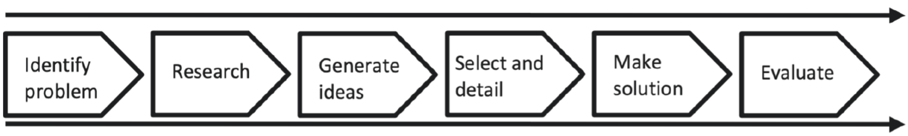


Fig. 1. The problem solving process, adapted by [7].

Davidson and Sternberg [8] states that the types of problems that can be classified for solving can either be well defined or ill-defined. Well defined problems are those with clear goals, obstacles and path to the solution, while ill-defined problems have an unclear path to the solution. This then makes it more difficult to solve ill-defined problems as they do not have an unambiguous problem statement which makes problem representation more difficult. Davidson and Sternberg [8] gives a meta-level problem solving process consisting of eight steps that largely correlates with and partially extends the process presented in Fig. 1. The steps are: recognizing that a problem exists; defining the nature of the problem; allocating appropriate resources to solve the problem; deciding how to represent information pertaining to the problem; deciding on appropriate problem solving steps; combining the steps into a problem solving strategy; monitoring the execution of the strategy; and finally, evaluating the problem's solution.

2.2 Problem Based Learning

PBL is "an active pedagogical strategy commonly used in some disciplines to develop general and specific competences" [9], p. 105. It is a teaching model that encourages learning by applying knowledge of content and the use of problem solving skills to solve problems relevant to a specific discipline [10]. The introduction of PBL in universities has been used to develop professionals who are better at actively seeking solutions.

PBL was first used in the medical studies at McMaster University by Barrows and Tamblyn [11] who hypothesized that learning through solving problems would be more valuable than memorizing a large amount of content knowledge and that problem solving skills are more important for medical practitioners than having competent memory skills. PBL places the responsibility for determining the requirements of the problem in order to best define and then solve it on the student instead of the teacher. This may also include finding needed learning material in order to solve the problem [12].

The six core features of PBL, as defined by Barrows [13], are what differentiates the classroom pedagogy from other collaborative learning pedagogies. The features are: student-centered learning; learning in small groups; teachers who only act as facilitators in the learning process; problems being the main focus when learning; the use of problems to spur the development of the students' critical thinking and problem solving skills; and the discovery of information through self-driven learning.

Janpla and Piriyasurawong [14] developed a PBL model, with the following steps: define the problem; understand the problem; continue the research; synthesize the knowledge; conclude and evaluate the answer; and present and evaluate the students' work. It can be seen that these steps resemble the problem solving processes described in the previous section.

PBL enables students to be better prepared for the real world [5]. Skills are developed that can be learned through solving problems and to help store, retrieve and use information. PBL assists with achieving the following [4]:

1. The ability to connect content knowledge to real world application;
2. Creating deeper understanding by requiring the use of content knowledge in new ways through collaboration with others;
3. Encouraging students to reflect on their use of thinking structures by helping them evaluate their thinking process;
4. Employing the use of technology to help students independently find the most apt tools to solve their problems with;
5. Using students to create, verify and evaluate their own contributions and the contributions of others;
6. Providing opportunities for students to solve complex problems that may result in new perspectives and solutions to problems;
7. Encouraging collaborative efforts in the gathering of information, the solving of complex problems, the brainstorming and sharing new of ideas;
8. Providing opportunities for students to take responsibility for their learning and working well with others; and
9. Helping students connect between subjects, ideas and other people.

3 Research Method

The research methodology carried out in this systematic literature review follows the guidelines by Okoli and Schabram [15], as shown in Fig. 2:

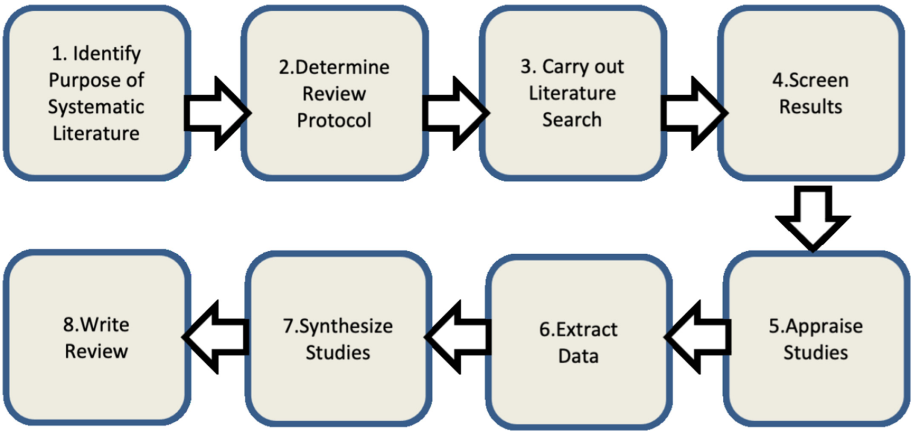


Fig. 2. The SLR process (Okoli and Schabram, 2010)

The research question guiding this study is: *How is PBL used to teach problem solving to undergraduate STEM students?*

The research question is limited in scope by focusing on PBL and how STEM undergraduate students - students who are in the Science, Technology, Engineering and Mathematical fields are taught. The following search term was used:

((“problem based learn*” OR “teaching problem solving”) AND “undergraduate” AND (“engineer*” OR “math*” OR “comput* (NEAR/2)”))

A Google Scholar search returned 2500 results, which were too many even before applying selection filters. The same search term was subsequently used in a different database search, namely EBSCOhost while restricting the database to only return results from MasterFILE Premier and Academic Search Complete. These databases included full text search results. After applying the source selection and inclusion and exclusion criteria presented below, this search returned 127 potentially relevant sources.

3.1 Source Selection

- The databases chosen included journals and articles that gave detailed information on problem solving and problem based learning for undergraduate students in STEM.
- The databases chosen included comprehensive information on the topics presented and each database was not subject specific.
- The databases allowed for advanced Boolean searches with keywords.
- Through the University of Pretoria library, full text documents were accessible from the database search.
- Two other relevant sources, namely the Journal of Information Systems Education (JISE) and the African Journal of Research in Mathematics Science and Technology Education (AJRMSTE) were included. These were not available through the data-bases and were searched separately.

3.2 Inclusion Criteria

- Texts that directly answer the research question.
- Focus on teaching using PBL to undergraduate computing/computer science, information system, engineering and mathematics students.
- Studies that were published in the years 1980–2018.
- Empirical studies on problem solving in computing/computer science, information system, engineering and mathematics.

3.3 Exclusion Criteria

- Texts that focus on teaching problem solving for STEM subjects to non-STEM students.
- Studies that rely purely on expert opinion or are purely anecdotal.
- Studies that are not related to any of the research questions.
- Studies that are not of undergraduate STEM university students - including high school students and post-graduate students.
- Studies whose findings are unclear.
- Studies that are not in English.

The PRISMA flowchart summarizing the search process is presented in Fig. 3.

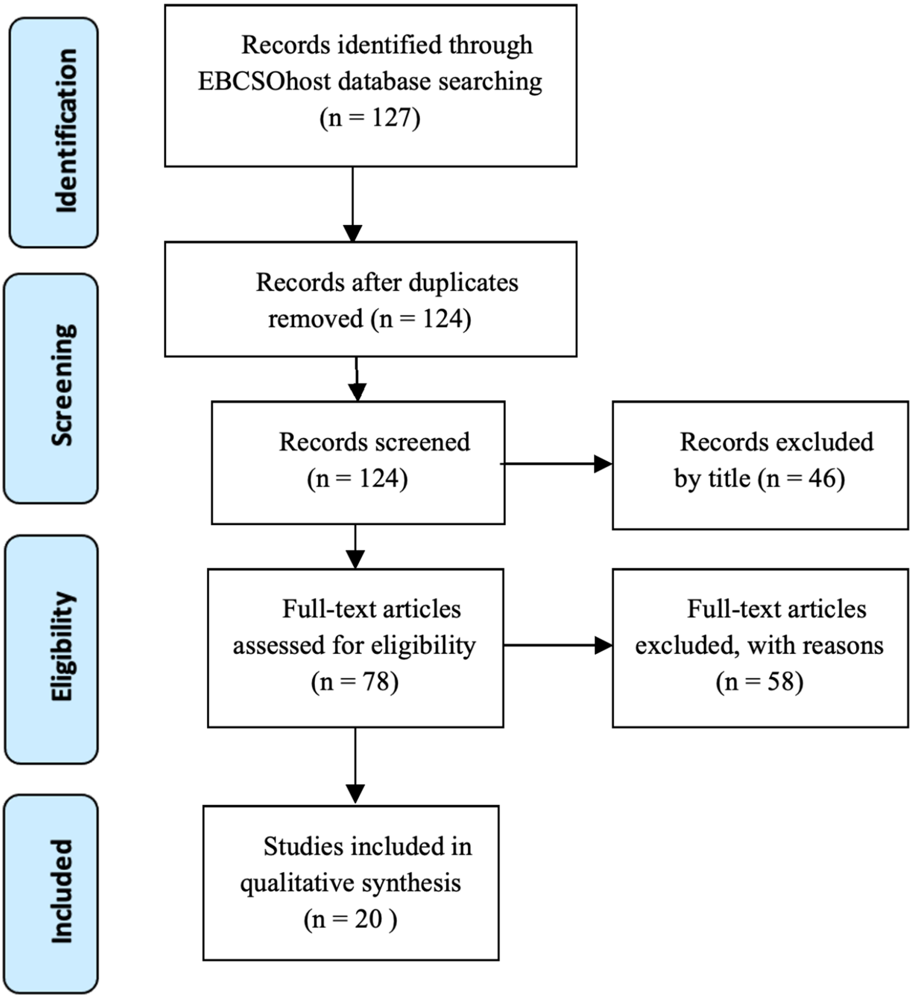


Fig. 3. PRISMA flowchart

4 Analysis of Findings

This section reports on the findings arising from the 20 articles analysed. The analysis will commence with a summary of the data in the form of descriptive statistics. Thereafter a thematic analysis will be presented.

4.1 Descriptive Statistics

Countries in Which Studies were Performed. Figure 4 indicates that the United States dominates as the country with the highest number of studies of

PBL in undergraduate STEM students. Seven of the 20 studies were performed in the USA, more than double the number performed in any of the other countries. Overall, most of the studies occurred in the US, UK and Europe.

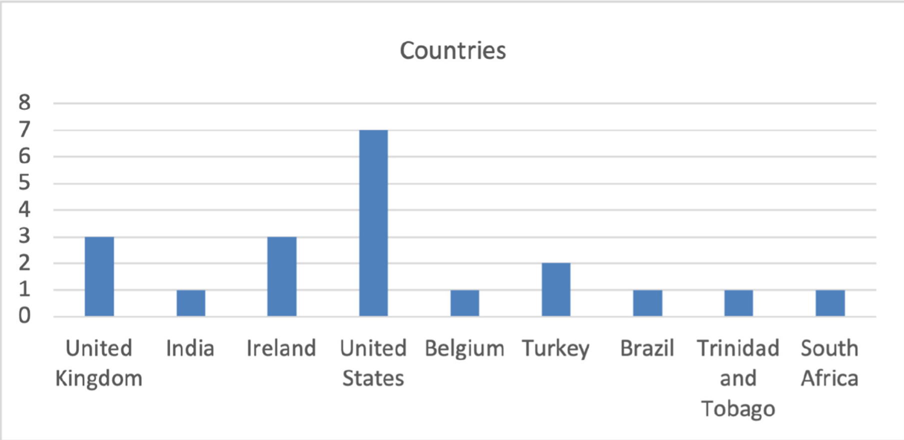


Fig. 4. Countries in which the studies were performed

Journals in Which Studies were Published. The journals that included the largest number of studies were engineering education journals, with the European Journal of Engineering Education containing the most studies, namely five of the 20 (see Fig. 5).

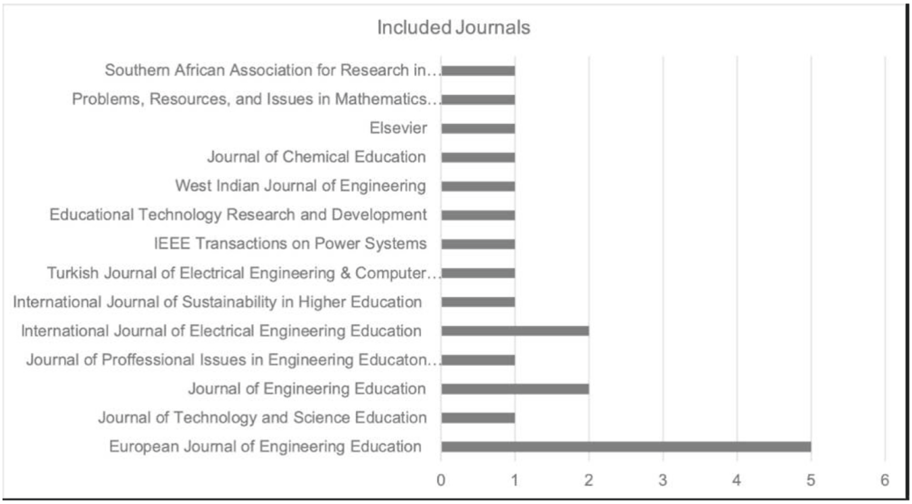


Fig. 5. Journals in which studies were published

Domains. The overwhelming majority (15 of the 20) studies were performed in engineering education, as can be seen in Fig. 6. However, there were also 3 studies in computing education, which will be further discussed in the thematic analysis that follows.

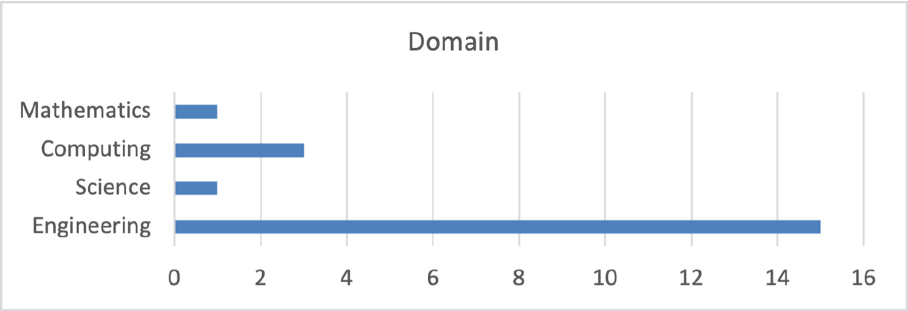


Fig. 6. Domains in which studies were performed

Study Year During Which PBL was Introduced. From Fig. 7 it is evident that there is a preference to introduce PBL at the earlier years of study. This makes sense, as problem solving is a basic capability required by STEM students, and in particular engineers (since most of the studies were in the engineering domain).

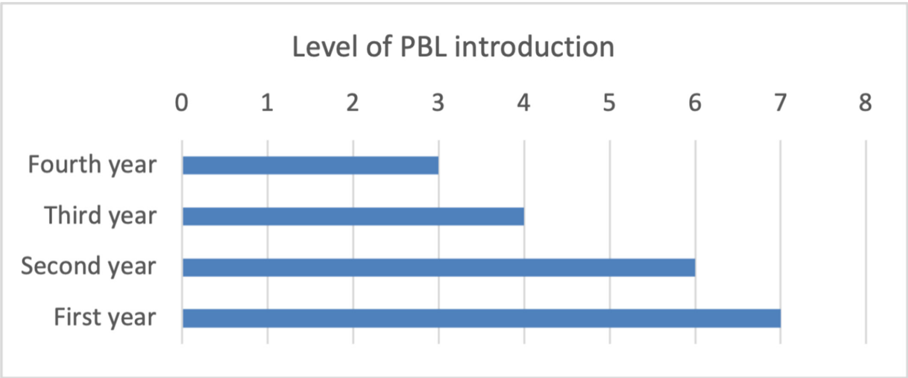


Fig. 7. Study year or level of introducing PBL to students

From the descriptive statistics, we learn that in the 20 articles included in the SLR, engineering education dominated as the domain where most studies were performed. There was a preference to introduce PBL in the earlier years of undergraduate study. Further, the USA, UK and Europe dominate as the locations where most studies were performed.

4.2 Thematic Analysis

A thematic analysis of the 20 papers resulted in the identification of four emergent themes. The analysis was done with the assistance of a data table, where the articles appeared as rows and the issues they addressed were noted in columns. The four common themes that emerged are: The role of prior knowledge, teamwork and team composition, the role of the instructor, and contextual learning methods.

The Role of Prior Knowledge. Prior knowledge is the amount of content knowledge imparted to students prior to the introduction of the PBL methodology. The amount of prior knowledge that has been imparted to students seems to have a favourable effect on the students and their perception of the PBL methodology, as will be seen in the discussion that follows.

In McCrum's [5] interdisciplinary study involving structural engineering students, he observed that students need to have a strong base on their content knowledge that they should be able to draw from in order to better engage with the PBL tasks. This will allow them to use, examine, and integrate problems more effectively [5]. The students will then be encouraged to connect prior learned ideas, experiences and knowledge and use those ideas in identifying patterns and principles that may exist, examining evidence against the initial assumptions made, and critically checking the logic and arguments [5]. "Based on the principle of using problems as a starting point for the acquisition and integration of new knowledge, the method is designed to create learning through prior learning experience and to reinforce existing knowledge" [16]. Ensuring that the students have reinforced their content knowledge would provide them with the skills and confidence to engage more fully in the PBL interaction which will address Gavin's [17] concern namely that missing concepts may cause a failure to learn in the PBL approach.

Research has shown that students rely heavily on their prior knowledge when solving problems in the PBL approach, which has created the assumption that a stronger knowledge base results in better problem solving [18]. This has been illustrated by Bledsoe and Flick [18] in Fig. 8.

Prior knowledge can be reinforced in multiple ways, with the most popular approach being to follow the didactic (traditional learning) approach for one or two years of undergraduate study [19]. In Mitchell and Smith's [20] study involving electrical engineering students, the students followed the didactic learning approach in order to be taught foundational knowledge as preparation for introducing PBL in their curriculum. However, other learning approaches such as didactic, Socratic and inquiry-based should be viewed as complementary to the learning process to ensure that the students receive the best possible outcome from PBL [19].

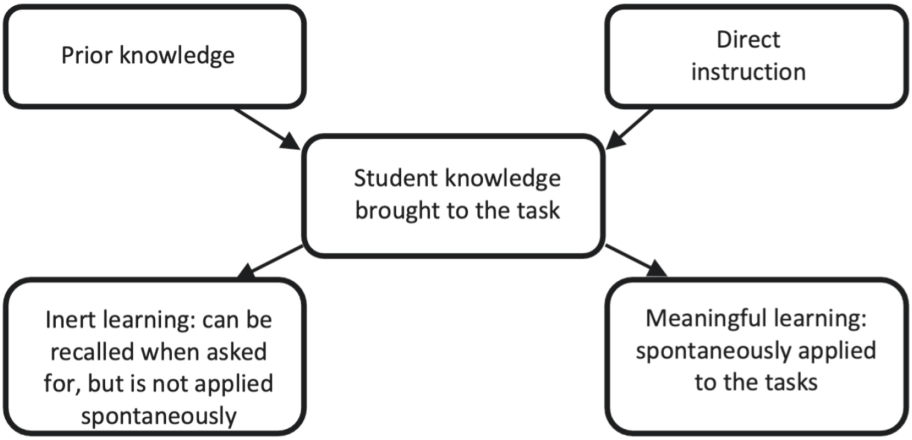


Fig. 8. A model of learning as adapted from Bledsoe and Flick [18].

The use of PBL in computer science education is also viewed as a way to engage with prior knowledge while moving away from the didactic learning approach to a more problem centred pedagogy. Dunlap [21] used the software development life cycle (SDLC) approach to solving problems that students would be more familiar with and mirrored it to the PBL approach as it is more closely aligned when restructured to accommodate prior knowledge. This helps both the instructors and students as they don't have to introduce new concepts while partaking in the problem centred pedagogy [21].

If prior learning is not carefully reinforced, it may have adverse effects on the progress of introducing PBL to students. In a study by Dyer, Grey and Kinnane [22], the majority of civil engineering students were unable to create simple sketches at the beginning of the design-led introduction of problem based learning, even though they were expected to have skills such as being able to sketch and draw when practicing professionally. This needed to be addressed by the instructors where formal lectures and design studios had to be held in order to first show the students how to do those tasks properly and then engage in the PBL approach [22].

Adverse effects may occur where the solution to the ill-defined problem that is given in a PBL environment cannot be found through the following of a set of classroom notes or the use of an example given in a textbook [23]. When traditional tools used to understand and solve problems are no longer sufficient in the PBL approach, students have found that they have to use external sources of information such as the internet or library in order to gain more insight on the important concepts and how to approach and solve the problem presented [23].

Teamwork and Team Composition. Studies across engineering as well as computer science have noted the importance of teamwork during use of the PBL approach [5]. This section discusses the number and types of students in the teams that the researchers have decided to form as part of their studies related to PBL.

A study by Yadav and colleagues [24] promotes heterogenous teams with numbers ranging from 3–5. Students work with an instructor who facilitates the students' learning journey. At least one electrical and one computer engineering student was placed in the same team [24]. In a study by Vemury, Heidrich, Thorpe and Crosbie [25] 8–12 civil engineering students of heterogenous academic abilities were placed in what they referred to as design groups to produce deliverables that reflected the objectives given in the design brief.

In Johnson and Hayes' study [26], students worked in groups to develop a number of competences in the collection, evaluation and duplication of data in order to solve complex problems that reflect those found in the real world.

Mitchell and Smith [20] struggled to find guidance from literature with regard to the selection of teams in a PBL approach, and opted for a solution that combined having tutors select groups based on their ability and having students select their own teams. Students were given the flexibility to choose two of their friends to be their partners while tutors randomly selected another group of students to form part of the group to form a larger group of six [20].

In a study with computer science students [21], the team was composed of what would most likely look like a software development team structure with three to four students working together to define the problem given by the client and the clients' needs, the written proposals, conduct analyses, plan solutions, implement and test those solutions [21].

In Şendağ and Odabaşi's [27] study, smaller groups termed sub-groups were randomly assigned. In this study individual reports on the problem solution are combined with group work to determine individual merit [27]. Martí, Gil and Juliá [28] opted to create groups of five to twelve students that could suggest either a problem or project that works towards the solution. Quinn and Albano [23] found having two to four students in a group with a faculty advisor playing a supportive role was an ideal arrangement for PBL. The students selected their own groups and used their interests to help them choose a faculty advisor based on their needs [23]. In her study of BSc Computer Science and Information Technology students, Havenga [29] followed a similar model with groups that comprised of only two members. However, because of the course demographics being mostly male, the teams were homogenous in skill set and gender [29]. Persad and Athre's [30] study of various engineering disciplines finds that students should not be given the choice to form their own teams and should be pre-screened by the instructor before being assigned. This is to ensure that students who are stronger academically are distributed evenly throughout all the groups while also having a student that is proficient with creating prototypes and models [30]. This creates heterogeneity and balance in the class. The number of students that made up a group was four, with each class having an even number of groups [30]. Denayer, Thaels, Sloten and Gobin [31] opted to have a randomized group of five to seven students without pre-screening students' academic abilities or skill sets. This strategy was mirrored in Cowden and

Santiago's [32] study where three to five students were selected based only on their pre-existing lab partnerships.

To conclude: while there is no consensus on the sizes of teams that perform PBL, the heterogeneity of teams in terms of skills, ability as well as demographics is emphasized.

The Role of the Instructor. The role that the instructor plays in the PBL approach is one that facilitates the students' learning. The amount of support offered to the students varies by study but the most common role of the instructor is to act as a guide and motivator to the students.

In a PBL study with structural engineering students [5], in the architectural design module, the tutor played a role that presented students with guidance with regards to design and not instructions. Students were assisted to improve their learning of foreign concepts through interactive engagement, to reflect on their content knowledge and to motivate them with their tasks [5].

In Quinn and Albano's [23] study involving structural engineering students, the instructors role was defined as one that would stimulate learning and research thinking by avoiding the traditional didactic role of being the expert source of knowledge. The instructor had to be approachable by listening and asking questions from the students, cultivating an environment where the students would be comfortable to share their ideas and discuss their understanding of the theories for solving the problem [23].

The PBL approach encourages instructors to facilitate the students' learning process [26]. With the instructor no longer taking the role of being the sole source of content or information, the responsibility is placed on the student to find external sources of information [26]. This was found to urge the groups to more extensively explore their options and only receive input from the instructors when asked or when the direction of the problem was obscure [20]. However, with traditional teaching methods engrained in the instructors, guiding the group towards the instructor's way of thinking was sometimes inevitable [20].

Motivating the students and creating the desire to learn has been shown as integral to the teacher's role in a study involving civil engineering students [33]. In this study, students were found to feel frustrated as facilitators only provide limited guidance on how to tackle the problems and could not provide them a concrete solution to the problem [33].

In a study with computer science students, the instructor did not provide direct instruction to the students on solving the problem but gave guidance to help the students with the process of brainstorming and prioritizing ideas [21].

In Şendağ and Odabaşi's [27] study of online computer science students participating in PBL, the instructor played a more active role in participating in discussion forums, responding to questions, encouraging students to think more deeply and conduct external research and use that information to make decisions on how to approach the problem. He challenged students through questions and encouraged questions from students but avoided responding directly to enquiries that would lead to him giving explicit information [27].

In the PBL approach, the role of the tutor changes to become that of an advisor in the quest to solve the problem and not only an evaluator [28]. This fosters a better working relationship between tutors and students that can make tutors more approachable [28]. The role of the tutor is given as [28]:

– "Clarifying the student's ideas. Tutors should not impose their points of view, but rather identify contradictions in the group's reasoning and request opinions from all members;
– Encourage critical evaluation of ideas and knowledge;
– Facilitate discussion, through student interaction and exchange of ideas and experiences; and
– Discuss strategies with the students".

In Denayer et al.'s [31] study, the instructor was more heavily involved in the team's progress and monitored it using a pre-designed manual throughout the project lifespan. The tutor monitored the teams and had various tasks which include but were not limited to: answering questions about the design problem; acting as the customer to assist the students during the early phase of the design process; steering the various groups through the process; managing the students and their time schedule; evaluating the efforts from the teams and give feedback; and attending the team meetings to ensure they apply the design methodology [31].

Contextual Learning Methods. With PBL, learning can be enhanced through the use of contextual learning methods. In McCrum's [5] study of engineering students in the architectural design module, students were encouraged to use sketches of complex engineering ideas to help them achieve higher order thinking and equip them with the tools to solve more complex problems. It was believed that the ability to solve complex problems that were en-countered in the study would enable the students to better engage in the creative engineering and architectural design problems and be less frightened of the unknown [5].

In a study involving electrical engineering students, [24] problems were presented in a realistic case study format. The problems encouraged students to make more complex engineering decisions in the context of tasks which helped the students better develop their final project with the use of their conceptual understanding by means of in-class problems [24]. This is in line with Johnson and Hayes' [26] use of the PBL approach where real-world scenarios are used to help students conceptualise solutions.

Quinn and Albano's [23] study was based on an undergraduate project known as the major qualifying project (MQP) that comprises part of the degree requirements for engineering students. MQP concentrates on the application of the learned material in the students' discipline by creating a solution for real-world problems while working with an external organisation or company [23].

In Alayont's [34] study of Mathematics students, she created pre-class activities that the students could complete beforehand in order to give them relevant knowledge on the lecture subject and also help students work more efficiently during in-class activities. Mota, Mota and Morelato [35] chose to use computer-mediated learning in order to enforce PBL, since the study was conducted in a web environment. In a study with computer science students, the systems design for a real-life client was used to further enforce learning by means of a PBL approach [21].

Overall, it can be seen that authentic, real-life problem scenarios were typically presented to students as part of their PBL.

5 Conclusion

This study presents a systematic literature review of 20 papers with empirical studies on using PBL as a means to teach problem solving to STEM students. It was found that the United States appears to be most active in conducting PBL research in STEM education, followed by the United Kingdom and Europe. The overwhelming majority of the studies were done in engineering education (15 of the 20) and also published in engineering education journals. This makes sense as the ability to solve real-world practical problems is at the essence of engineering. It was encouraging to see that the second most number of studies were in the computing field.

A thematic analysis of the 20 studies resulted in the emergence of four overarching themes. These themes were evident regardless of the subject disciplines. The first theme emphasized the importance of prior subject knowledge when attempting PBL. PBL requires the creative application of domain knowledge, and hence can only be successful if participating students' domain knowledge is at the appropriate level. The second theme was around the importance of teamwork during PBL activities. Team members learn from each other, hence heterogeneity in team composition was emphasized as important. Thirdly, it emerged that PBL initiatives benefited from instructors who acted as facilitators in the students' learning process without helping them too much. Lastly, the role of contextual learning methods were emphasized. The real-world decision-making environment of the STEM graduate needed to be emulated in order for them to benefit most from the experience.

Of the 20 studies, the one where all of the mentioned themes were explicitly acknowledged was McCrum's [5] study on developing creative problem solving abilities in engineering students. Interestingly, the four identified themes also align closely with the framework presented in [36] for developing the creative ability of under-graduate Information Systems (IS) students.

This study was limited to the studying of selected empirical studies on PBL for undergraduate STEM students. It contributes by synthesizing from the literature, a number of suggested practices for teaching problem solving to STEM students by means of PBL. Further research is suggested to expand these practices for IS and computing students, since there are very few studies focusing on the use of PBL in the IS and computing domains. In the light of the importance of problem solving skills in the workplace it is also suggested that the 1993 National Adult Literacy Survey be revisited to see whether and how workplace problem solving skills have improved in the decades since.

References

1. Chaudhry, N., Rasool, G.: A case study on improving problem solving skills of undergraduate computer science students. World Appl. Sci. J. **20**(1), 34–39 (2020)
2. Foshay, R., Kirkley, J.: Principles for teaching problem solving. Technical paper (1998)
3. Jollands, M., Jolly, L., Molyneaux, T.: Project-based learning as a contributing factor to graduates' work readiness. Eur. J. Eng. Educ. **37**(2), 143–154 (2012)
4. Beers, S.: 21st century skills: preparing students for their future. Diakses dari (2011). http://www.yinghuaacademy.org/wpcontent/uploads/2014/10/21st_century_skills.pdf
5. McCrum, D.P.: Evaluation of creative problem-solving abilities in undergraduate structural engineers through interdisciplinary problem-based learning. Eur. J. Eng. Educ. **42**(6), 684–700 (2017)
6. Martinez, M.E.: What is problem solving? The Phi Delta Kappan **79**(8), 605–609 (1998)
7. Kimbell, R., Green, R., Stables, K.: Understanding Practice in Design and Technology. Open University Press (1996)
8. Davidson, J.E., Sternberg, R.J.: The Psychology of Problem Solving. Cambridge University Press, Cambridge (2003)
9. Hernández, C.H., et al.: Problem based learning and the development of professional competences: an experience in the field of biomedical engineering. Turk. Online J. Educ. Technol. (TOJET) **17**(3), 104–112 (2018)
10. Milman, N.B., Kilbane, C.R.: What is problem-based learning? In: Distance Learning, vol. 14, no. 2, p. 49 (2017)
11. Barrows, H.S., Tamblyn, R.M.: Problem-Based Learning: An Approach to Medical Education. Springer, Heidelberg (1980)
12. Goltz, S.M., et al.: Teaching teamwork and problem solving concurrently. J. Manag. Educ. **32**(5), 541–562 (2008)
13. Barrows, H.S.: Problem-based learning in medicine and beyond: a brief overview. In: New Directions for Teaching and Learning, vol. 1996, no. 68, pp. 3–12 (1996)
14. Janpla, S., Piriyasurawong, P.: The development of problem-based learning and concept mapping using a block-based programming model to enhance the programming competency of undergraduate students in computer science. TEM J. **7**(4), 708 (2018)
15. Okoli, C., Schabram, K.: A guide to conducting a systematic literature review of information systems research. Sprouts Working Papers in Information Systems, vol. 10, no. 26, pp. 1–51 (2010)
16. Mantri, A.: Working towards a scalable model of problem-based learning instruction in undergraduate engineering education. Eur. J. Eng. Educ. **39**(3), 282–299 (2014)
17. Gavin, K.: Case study of a project-based learning course in civil engineering design. Eur. J. Eng. Educ. **36**(6), 547–558 (2011)
18. Bledsoe, K.E., Flick, L.: Concept development and meaningful learning among electrical engineering students engaged in a problem-based laboratory experience. J. Sci. Educ. Technol. **21**(2), 226–245 (2012). https://doi.org/10.1007/s10956-011-9303-6
19. Güzeliş, C.: An experience on problem based learning in an engineering faculty. Turk. J. Electr. Eng. Comput. Sci. **14**(1), 67–76 (2006)

20. Mitchell, J.E., Smith, J.: Case study of the introduction of problem-based learning in electronic engineering. Int. J. Electr. Eng. Educ. **45**(2), 131–143 (2008)
21. Dunlap, J.C.: Problem-based learning and self-efficacy: how a capstone course prepares students for a profession. Educ. Tech. Res. Dev. **53**(1), 65–83 (2005). https://doi.org/10.1007/BF02504858
22. Dyer, M., Grey, T., Kinnane, O.: GiveMe Shelter: a people-centred design process for promoting independent inquiry-led learning in engineering. Eur. J. Eng. Educ. **42**(6), 729–744 (2017)
23. Quinn, K.A., Albano, L.D.: Problem-based learning in structural engineering education. J. Prof. Issues Eng. Educ. Pract. **134**(4), 329–334 (2008)
24. Yadav, A., et al.: Problem-based learning: influence on students' learning in an electrical engineering course. J. Eng. Educ. **100**(2), 253–280 (2011)
25. Vemury, C.M., et al.: A holistic approach to delivering sustainable design education in civil engineering. Int. J. Sustain. High. Educ. **19**(1), 197–216 (2018)
26. Johnson, M., Hayes, M.J.: A comparison of problem-based and didactic learning pedagogies on an electronics engineering course. Int. J. Electr. Eng. Educ. **53**(1), 3–22 (2016)
27. Şendağ, S., Odabaşi, H.F.: Effects of an online problem based learning course on content knowledge acquisition and critical thinking skills. Comput. Educ. **53**(1), 132–141 (2009)
28. Martí, E., Gil, D., Juliá, C.: A PBL experience in the teaching of computer graphics. In: Computer Graphics Forum. Wiley Online Library (2006)
29. Havenga, M.: Problem-based projects in computer programming: students' cooperation, responsibilities and dependencies. Afr. J. Res. Math. Sci. Technol. Educ. **22**(2), 254–264 (2018)
30. Persad, U., Athre, K.: Experiences with teaching introductory product design to engineering undergraduates. West Indian J. Eng. **36**(1), 66–78 (2013)
31. Denayer, I., et al.: Teaching a structured approach to the design process for undergraduate engineering students by problem-based education. Eur. J. Eng. Educ. **28**(2), 203–214 (2003)
32. Cowden, C.D., Santiago, M.F.: Interdisciplinary explorations: promoting critical thinking via problem-based learning in an advanced biochemistry class. J. Chem. Educ. **93**(3), 464–469 (2016)
33. Ahern, A.A.: A case study: problem-based learning for civil engineering students in transportation courses. Eur. J. Eng. Educ. **35**(1), 109–116 (2010)
34. Alayont, F.: Using problem-based pre-class activities to prepare students for in-class learning. Primus **24**(2), 138–148 (2014)
35. Mota, A.A., Mota, L.T.M., Morelato, A.: Teaching power engineering basics using advanced web technologies and problem-based learning environment. IEEE Trans. Power Syst. **19**(1), 96–103 (2004)
36. Turpin, M., Matthee, M., Kruger, A.: The teaching of creativity in information systems programmes at South African higher education institutions. Afr. J. Res. Math. Sci. Technol. Educ. **19**(3), 278–288 (2015)

A Robust Portable Environment for First-Year Computer Science Students

Dane Brown(✉) and James Connan

Rhodes University, Grahamstown 6140, South Africa
{d.brown,j.connan}@ru.ac.za
http://www.ru.ac.za/computerscience

Abstract. Computer science education in both South African universities and worldwide often aim at making students confident at problem solving by introducing various programming exercises. Standardising a computer environment where students can apply their computational thinking knowledge on a more even playing field – without worrying about software issues – can be beneficial for problem solving in classroom of diverse students. Research shows that having consistent access to this exposes students to core concepts of Computer Science. However, with the diverse student base in South Africa, not everyone has access to a personal computer or expensive software. This paper describes a new approach at first-year level that uses the power of a modified Linux distro on a flash drive to enable access to the same, fully-fledged, free and open-source environment, including the convenience of portability. This is used as a means to even the playing field in a diverse country like South Africa and address the lack of consistent access to a problem solving environment. Feedback from students and staff at the Institution are effectively heeded and attempted to be measured.

Keywords: Computational thinking · Linux · Problem solving

1 Introduction

Computation skills in some form are an essential component of many undergraduate programmes and the wider process of lifelong learning [5]. To better understand the issues of teaching and learning in Computing Science, it is useful to have insight into the major concerns of the subject. While Computer Science has had some of the greatest impacts on everyday life (social media, automated services, etc.), it is also a fast moving discipline at tertiary level. Unfortunately, there are high dropout and failure rates at the Institution[1]. Increased risk of ill-prepared first-year students is another cause for concern in South Africa, especially with the combination of the 2019 "free education" and the amendment that lowered university entry level requirements [7]. Computer Science is thus a

[1] First year computer science had a combined 50% dropout and failure rate in 2018 at an unnamed university referred to as the Institution.

G. Wells et al. (Eds.): SACLA 2020, CCIS 1518, pp. 88–103, 2021.
https://doi.org/10.1007/978-3-030-92858-2_6

discipline that has evolved at considerable pace, and teaching the subject matter in South Africa has challenges ranging from keeping an up-to-date curriculum to dealing with the lowered education standards in primary and secondary schools.

In a bid to mitigate the impact of these changes as well as enhance computer science education in general, a portable environment can be constructed wherein students have consistent access to problem solving tools; the exact same tools (and environment) used everyday during lectures. Over the past two years at the Institution, this portable environment has proven to be robust across platforms and convenient to students and lecturers alike. This has especially been utilised in a novel way since 2019 by running it directly off the flash drive, without virtualisation. The environment combined with reassessed teaching styles and content that are centred around students rather than lecturers, is part of a contribution towards a new curriculum with promising preliminary results.

This paper is thus relevant to "TL; DR – teaching the new generation" as it directly addresses the theme with the following outcomes: Providing students with a portable, practical environment removes the requirement for students to spend time setting up via "how-to guides". Another advantage is that everyone can use the same environment wherever they desire – convenience with a chance of significant benefits during the COVID-19 crisis. The dreaded initial learning curve is reduced since students can learn how to use the environment directly from the lecturer during class time, and simply apply it during practicals. Lastly, students are free to use their own environment or the proposed one as a base to build their own.

The remainder of the paper is organised as follows: Sect. 2 adds context to the paper while Sect. 3 relates the literature. Section 4 explains the implementation of the proposed portable environment. Section 5 analyzes the new curriculum and discusses its contribution. Section 6 relies on student feedback and staff debates to determine the direction of the new curriculum. Section 7 concludes the paper and outlines future work.

2 Teaching Context at the Institution

While this paper introduces many technical topics, the goal remains teaching the new generation. Henceforth, the teaching context that accompanies technical contributions is first introduced. Margaret Mead said: "Children must be taught how to think, not what to think". How about students too? This is core to computational thinking – a set of cognitive and metacognitive strategies related to the use of procedural knowledge in problem solving [13]. Problem solving using Python is the core part of the first module during first year. Computational thinking using notations on paper are also lectured but once a week between problem solving lectures. Therefore, students spend most of their time solving problems on computers while having a pen and paper handy as suggested by the literature [4,12]. These students come from diverse backgrounds, many of which have not used a computer before. This is why normalising the environment used for programming for all students has been one of the major goals during the past three years – diverse student-centric goals[2].

Deep or surface learning is not inherent to a student but depends on the context and style of teaching [2]. However, certain students may not have an effective learning strategy. Lecturers can help them "catch up" with students that find learning to be a simple, natural phenomenon. Computational thinking develops new thinking strategies to analyze, identify, and organize relatively complex abstract tasks towards problem solving that often makes use of algorithms [10,13]. Showing students why understanding a concept instead of just memorising without reason, is key to enabling an effective learning strategy in Computer Science across the undergraduate curriculum.

Constructive alignment means coherence throughout the undergraduate curriculum. All components in the system address the same agenda and ideally support each other across all years. The students are "entrapped in this web of consistency", optimising the likelihood that they will engage the appropriate learning activities, which relies on the effective use of procedural knowledge in Computer Science [10]. However, routines differ once in the real (working) world. Students are encouraged to mold the information that they receive in such a way that they can adapt their own style as a long term outcome of this pedagogy – not only learning *what to think* but *how to think*. Should (especially computer illiterate) students have to worry about the acquirement, set up and possible maintenance of technical resources, such as software licences and differences in software versions – nagging updates? Not to mention the time tutors and lectures spend on students that panic when they clicked update as suggested by the nagging screen, only to be greeted with, please register your copy.

As Benjamin Franklin said: "By failing to prepare, you are preparing to fail". Sufficiently preparing technical resources, including equipment and tools can benefit students (especially in their first year) when learning the fundamentals of problem solving [5]. Teaching practices used include a simple iterative process of preparation, reflection and finally revision. The success of the curriculum has relied on this process since 2012, and more notably since the implementation of the proposed curriculum since 2017. But how does one measure success? A typical way most courses at universities measure it is by the pass rate, annual or degree completion rate, retention of students, and lastly, by default, class average as suggested in the literature [8,11]. Individually, students have duly performed (DP) requirements at South African universities, which typically rely on a coursework mark, attendance, and completing a certain number of assessed tasks. However, success remains a subjective concept, and for now running some basic statistics, such as standard deviation of year marks, is better than no measure at all. Measuring success is important because in tandem with reflection, it can improve the review process. In this case, can a technical system at least partially bridge the gap of student diversity?

[2] Teaching focused goals are in the co-authored SACLA publication: "Using Technology to Teach a New Generation".

3 Related Portable Systems

Three related studies that use USB flash drives to offer a preset practical environment are discussed. Thereafter, a brief summary is presented and directions for an approach are provided.

3.1 Customisable Portable Virtual Machine (CPVM)

Butler and Pettey [3] presented a customisable portable virtual machine (CPVM) environment such that both students and professors can develop a customised, fully loaded, functional virtual machine (VM) on a flash drive. This VM can run on any computer that has a USB port or an ftp client without significantly compromising the host. A copy of the portable VM, provides compilers, window managers, and integrated development environments (IDEs) on a selective operating system (OS). They use the same copy across labs. The CPVM contains free, open-source software, available via anonymous ftp. This allows students to use their personal Windows computers as if they were the Linux machines in the lab without having to install Linux. Furthermore, instructors use it so that they can 'take' their office environments with them to remote locations such as conferences. Their environment is, however, restricted in size and builds off a processor-emulation layer via the VM. The main issue is that it leads to lowered response time even after the OS loads. Another disadvantage is that users still need access to host machine software such as Hypervisor, VMware or VirtualBox.

Wagner and Johnson [15] created a portable software called MOBILE, specifically geared for course instructors to hold workshops on any available computers, including different lab settings or a privately loaned computer. The software runs on a Virtualbox copy of a customised Ubuntu Linux. The software includes the ability to build and configure a workshop session on any desired information technology topic, creation and management of that session on top of any existing network OS. MOBILE is different to the previous study, since it does not rely on host machine software as far as hard disk space is concerned. This means that the physical drive is directly accessed, providing faster reading and writing operations. MOBILE successfully piloted the portable system to conduct four highly-rated workshops at technology education conferences and more after the time of writing. This demonstrates the practical use of such portable systems. Another advantage of MOBILE compared with CPVM is that the full environment runs directly on the hardware without requiring user accounts.

3.2 Lab on a Stick

Shipman and Bull [14] produced a portable environment named Lab on a Stick. They base their system on MOBILE to be a portable, modular, and scalable way to specifically teach networking, systems administration, and computer security related material to students of all ages. This is very similar to MOBILE as it functions inside a virtualised laboratory environment, on commonly available USB flash drives, which allows for ease of distribution to students. In other words, students can complete various networking or computer security related exercises in a classroom environment, and take it home to continue to work at their own pace. A promising result in their study, over four different use cases, show it had a positive productive effect on varsity students for full semester varsity classes, cybersecurity summer camps for school students, and for extended studies school students.

3.3 Summary of Studies and Proposed Approach

Summary of Studies: All three of the related studies enable a portable customised virtual environment with benefits over default approaches as it is decentralised with the use of a USB flash drive. This is an extremely useful for first year students, especially those that want to work in a location other than the 24/7 labs at the Institution. But what about providing a native fully customised environment on a USB flash drive? The advantages of this includes better performance and less disk space usage, which means it can work on smaller, cheaper (8 GB) flash drives.

Diversity: The Institution accepts a diverse number of students. Those that have only had limited exposure to computers can feel very intimidated by those who have had a lot of exposure. One way to limit this effect is to put everybody into a new but productive environment for problem solving. Students are hence put into a GNU/Linux environment. Those already familiar with Microsoft Windows are exposed to a new environment, while the rest do not feel intimidated because their peers seem to "know everything beforehand".

Teaching in the Environment: A major advantage of this portable environment is that an exact duplicate can be given to each and every student without any concerns about proprietary licences. The student can also install the exact environment used in the classroom and laboratory on their personal machine by double clicking on a script.

The language selected for this course is Python, which can be defined as a relatively simple interpreted language that is widely used (nationally and internationally) for scripting and rapid prototyping. Python is slow, but highly portable, allowing for flexibility for students that wish to deviate from the portable environment in the second semester of first year. From the perspective of a student, it supports a lot of modern programming concepts and importantly it has very good support for the five basic principles of programming: input, output, branching, repetition and calculations – fundamental for the first semester of first year.

With the student base in mind, the portable environment needs to be based on a user-friendly OS such as Linux Mint[3]. This can run off either a virtual machine or a native environment. Both of these approaches were explored since 2017 and implemented while paying close attention to both formal and verbal

[3] Consensus in the Linux community.

student feedback to determine its success. However, what constitutes success and how is it measured?

Measuring Progress in Context: Being part of the solution that equips students with the fundamentals has been a goal of the ComSci team[4] since 2017 at the Institution. First year is arguably the most important course that an aspirant computer scientist will ever do[5], as it is the foundation on which all other computer science courses are built. How does one know whether they are successfully teaching the fundamentals of Computer Science such that it feeds into the next course/module, the one after that, and so forth? In other words, when is constructive alignment obtained?

The proposed pedagogy does not measure success using hard outcomes alone, and occasionally follows the student body language, questions and attention – subjective measures. On the other hand, in a study titled "How do you measure success?" [6], the authors coin the phrase Academic Wholism as a proposed measure of success. Concisely, by using this pedagogical paradigm students can expand their learning profiles beyond the cognitive level to the personal domain so that success becomes a multifaceted reality; self-empowerment and self-confidence are included in that domain. While this appears to work well in their case study, it is not as easy to generalize this for Computer Science. The ComSci team at the Institution have bounced ideas on this and alternatives to no avail, and have thus deferred it to future work.

Other, arguably subjective measures include polling students or colleagues during and after classes or with course feedback forms or a simple informal conversation [9]. This is performed extensively, and one of the measures that the approach in this paper relies on.

Monitoring the grades of undergraduate students was deemed good enough by the department in 2017[6]. The idea is to use this success measure in order to determine whether there is alignment across all undergraduate years. But what do grades alone over a three year period represent – graduation with the undergraduate degree? In other words, the first year programming course should be pitched at such a level that the average student that is graded in the course, has a realistic chance of completing the entire undergraduate degree. This, together with adequate coverage of the relevant concepts is the proposed measure of success at the Institution. This will be further evaluated in Sect. 6. Before this evaluation, the implementation of the portable environment will fully detailed.

4 Creating the Portable Environment

The proposed portable environment aims to cover all the tools required with regards to learning during first year. In Fig. 1, the Desktop of the environment

[4] The name of a team comprised of four lecturers (including the authors of this paper) in first year at the Institution.

[5] Academic staff at the Institution agreed in a meeting.

[6] Used in several meetings and presented in this paper generically for ethical reasons.

Fig. 1. Desktop of the environment is shown for visualisation purposes.

is shown with the quick_help directory containing additional Python help from reputable free books, a 'quicksheet' and the PDF version of the textbook.

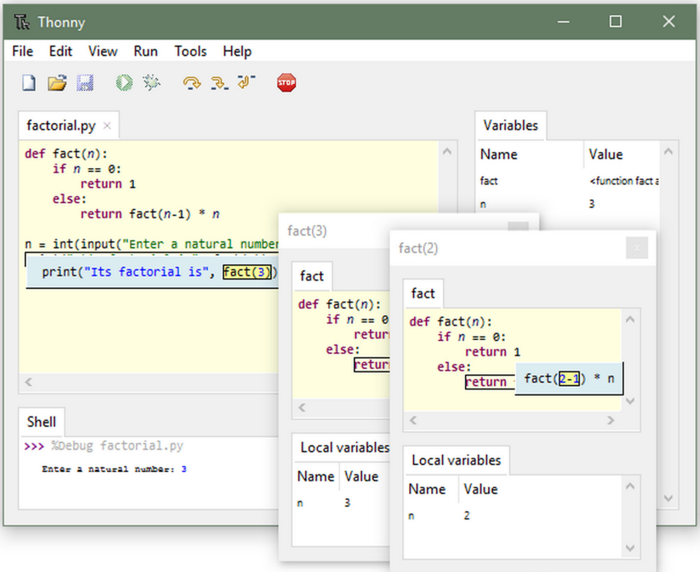


Fig. 2. Thonny IDE tracking local variable use.

4.1 Integrated Development Environment (IDE)

It is not always possible to come up with an objective measure of how good a particular IDE is. The 'best' IDE is thus well suited to novice programmers and that meets the objectives of this course.

Thonny calls itself "The IDE for beginners" and is developed specifically for teaching Python Estonia [1]. It is cross-platform and synergises well with the portable environment as it is 'lightweight'. Figure 2 shows how variables can be tracked with verbosity enabled that is often used by the majority of students. Also note that Thonny and a terminal is readily available in the task panel/bar next to Firefox in Fig. 1.

4.2 Runestone Interactive Textbook

The content for the course has evolved over several years, which led to the adoption of an open interactive textbook, "How to think like a Computer Scientist"[7]. Runestone Academy provides a framework for presenting interactive courses for both the student and instructor and includes this book. While students work through the content, this activity gets logged for the instructor to monitor. The anonymised stats, such as those in Fig. 3, are shown during some lectures when key concepts have been covered or it is noticed that many students were especially negligent during a particular week. Students are also presented with a PDF version of the book on the Desktop of the portable environment, and have the option to print and bind it.

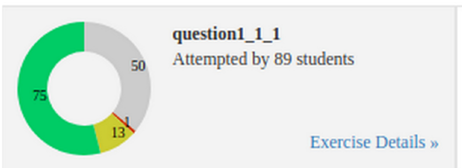


Fig. 3. Runestone showing engagement by students on the first exercise.

4.3 Imaging the Persistent Flash Drive

The final step involves creating a persistent bootable USB flash drive – a live USB bootable OS but with the additional ability to save changes in the same way as one would on a normal PC. This setup provides two advantages to students: they can save their work in the exact same manner as running Linux on an internal hard disk and they can take do this on almost any PC, including

[7] How to Think Like a Computer Scientist: Interactive Edition (2019), https://runestone.academy/runestone/books/published/thinkcspy/index.html.

laptops, chromebooks and tablets. A program called guidus was used to create a persistent USB flash drive.

A GUID Partition Table (GPT) was adopted, which allowed for the creation of more than three primary partitions. Besides the four partitions required for the persistent OS, an additional convenient EXFAT partition was created that can be accessed on Windows and MAC and used for typical flash drive purposes as seen in Fig. 4. The flash drive was successfully tested on a variety of machines – legacy and new on various platforms including, laptops, macs, tablets and your average PC. Since graphics drivers have had substantial on these machines, the GRUB menu was modified to include a typical option and one for failsafe graphics. This has catered for all machines to date but one – a tablet with a display orientation issue. The dd command produces an image of the resulting flash drive, which allows for mass cloning purposes. However, initial testing on different machines revealed GRUB does not load correctly on legacy BIOS-based personal computers. This was corrected by adding a BIOS boot partition.

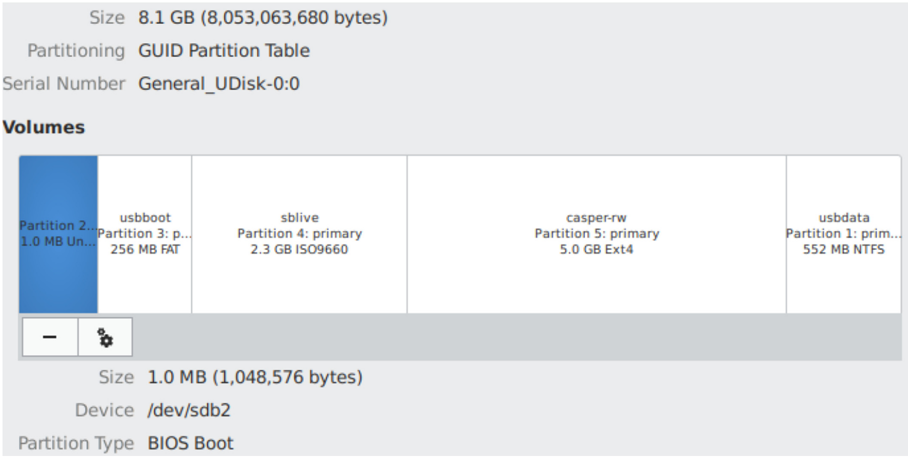


Fig. 4. GPT partition stucture of the portable environment flash drive.

The BIOS Boot Partition: is a partition on a data storage device that GRUB uses on legacy BIOS-based personal computers in order to boot an OS, when the actual boot device contains a GPT. In other words, the BIOS boot partition is only needed by GRUB on a GPT setup as seen in Fig. 4. On the other hand, on a MBR setup, GRUB uses the post-MBR gap for embedding the core image. On GPT, however, there is no guaranteed unused space before the first partition. For modern BIOS-based personal computers – that is, UEFI systems, the extra partition is not required, since no embedding of boot sectors takes place in that case. Installing GRUB on UEFI systems still requires an EFI system partition whether MBR or GPT setup. The image is successfully created using a combination of the mksquashfs command to capture the customised Linux Mint and guidus[8] to place the image on USB. Finally, the dd command is used to mass image the flash drives now that it works across all BIOSs.

Secure Boot and BIOS Security Caveat: The proposed distribution supports Secure Boot, as it comes with signed bootloaders. Furthermore, Grub has been setup to allow the distribution to work on both legacy BIOS and newer UEFI systems. A significant caveat, however, is that other than labs, many computer labs on campus have a locked BIOS. Students are made aware of this and those owning personal laptops and tablets are encouraged to ask the lecturer for help for enabling permanent automatic USB boot.

5 Assessment

The first year curriculum is geared according to the students' needs that – "TL; DR – teaching the new generation". First, it needs to prepare mainstream students to have sufficient problem solving skills to complete their computer science degree. Secondly, it needs to be comprehensive enough to students that will only ever do one course in programming to leave with sufficient knowledge to be able to write basic programs. Furthermore, it needs to provide students, that are from other disciplines, with enough tools to enable them to better perform their respective tasks.

The portable environment started off as a virtual machine on a portable flash drive in 2018 similar to the related studies, but it has evolved into a fully-fledged native portable environment, used for 2019 and 2020. The major problem encountered were from unaware students that would pull out the flash drive while it is still in operation. This was mitigated and is detailed in Sect. 6.2. It is important to note that students are free to use any environment they want as long as they use the assessed programming language. However, overall many students tended to stick to the portable environment or use it as a basis for their own, as they preferred to become accustomed to the environment that they would be assessed on.

5.1 Catering for Diverse Students Cohorts

Students who typically subscribe to this course have a diverse set of motivations for doing so. Moreover, this subsection highlights that all students are assessed in the same way but have substantially different initial skill levels.

The first group of students are mainstream and will have done either a programming course or computer literacy at school, as opposed to their counterparts that have had very little to no exposure to computers. The next group of students are those who do not intend to continue with Computer Science, but would like some exposure to programming before continuing into other fields. This group

[8] Ubuntu:mkusb – tool to create boot drives (2020), https://help.ubuntu.com/community/mkusb/.

may also contain a group of students who are misinformed about what exactly an introductory programming course is and had mistaken it for a computer literacy course. The last group are often senior, sometimes postgraduate, students, who have realised that they need some exposure to programming in order to do their studies in fields outside of Computer Science. It is desirable for this group to not just get some experience and learn what Computer Science is about, but also to acquire a usable and useful skill. For example, data modelling using Python for use in subjects such as AI-assisted climate change analysis in Environmental Sciences or rock pattern analysis in Geology etc. Moreover, a trending profession like data science benefits greatly from basic Python skills.

No matter the type of student, the curriculum needs to be comprehensive enough that those who will only ever do this one course in programming leave with enough knowledge to be able to write basic programs, and it needs to give those students who have come in from other areas of study with enough tools to enable them to better perform their tasks.

While all of these students need to reach the the same finishing line, they start from very different places like the analogy in Fig. 5. Of course there are exceptions to this, but regardless, bridging this gap is a non-trivial problem. While technologies like cellphones are mostly accessible to everyone now, the cultural and economic context of the first group of students cause many to still have difficulties accessing certain technologies like the personal computer and the Internet. The ComSci team are in the process of transformation, in accordance with South Africa's demographics, and this was given particular thought especially in terms of which lecturer should present extra classes.

A prevalent trait of teaching and learning in Computer Science is that no matter a student's background, they lack "how-to" knowledge in this discipline. It is like "riding a bike", you are not born with it. While some students perhaps "pick it up" a lot faster, this still means that as lecturers, we start this curriculum from the ground up and focus on providing a solid foundation for second year and later year students. This also becomes convincing when you are the extra classes lecturer for two consecutive years[9] and feedback is very positive as reported later.

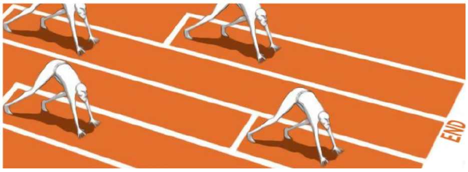


Fig. 5. The challenge of teaching first years sufficiently for constructive alignment.

[9] I had continually improving verbal feedback during those lectures after certain changes were made and much improved results from certain students.

The department has 24 h labs with PCs accessible to all computer science students. However, travelling at night can be a problem for those students that want to especially work hard[10], while spending too much time on computer science and neglecting other subjects is another potential problem. It is hoped that the portable environment allows students to access computers that are in closer proximity to them to avoid this problem.

5.2 Hard Outcomes: Towards Constructive Alignment

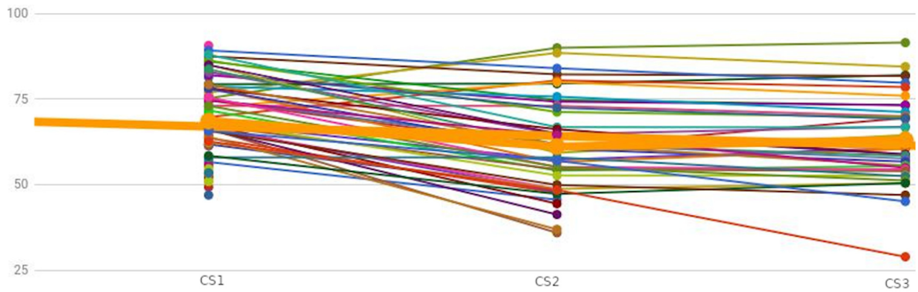


Fig. 6. Mark deviation (in %) as the same group of students (each colour lines) progress through first, second and final year.

In Fig. 6, it is apparent that the first-year course as a whole has been excellent in preparing students for following years – constructive alignment has been achieved from a first-year perspective. The class average is denoted by the thick orange line, with a trend that is almost flat but slightly downwards. This suggests that a student leaves first year with the right foundation to perform well in following years and with a realistic assessment of their ability. Individual semesters have not been analysed as of yet. While the trend line suggests that achieving a mark close to the class average sufficiently prepares one for second year, more data is required for conclusive analysis.

The extra classes track has also finally produced results that actually lead to additional passes. Positive results in practicals, tests and exams were recorded across a diverse set of student cohorts[11]. This positive result for the new curriculum was a huge victory for the student, ComSci team and hopefully, in the long term, the Institution as a whole. However, it is important to not become complacent but to continuously strive toward championing and achieving transformation. South Africa has a long way to go in achieving this goal. Another important note is that not just the student but *leaders can learn too*. Transformative teaching and learning should have an effect of mutual improvement between students and lecturers.

[10] A common problem in South Africa, nationwide.
[11] More detail: Ethics approval pending.

5.3 Problems with the Portable Environment

The total number of students for 2018, 2019 and 2020 are 99, 120 and 160, respectively. The 2018 group used a VM version of the environment stored on the flash drive that had multitudes of problems. VMs were thus used for future years.

Typically, 30–50 spare flash drives were imaged in case of general problems that may be encountered by students. Figure 7 shows how the number of problems were mitigated in 2020 compared to 2019 over a three week period. "OS/other" problems are defined as all issues that are *unrelated* to flash drives physically ("bricked"). In 2019, these issues were typically due to students pulling out flash drives prematurely or during write operations, as well as due to Linux running out of space. In 2020, these problems were virtually eliminated by allocating the majority of space to the Linux partition and by making all boot-related partitions read only. The positive effect this had is very evident in the graphs in the figure and has especially lead to far less issues in general.

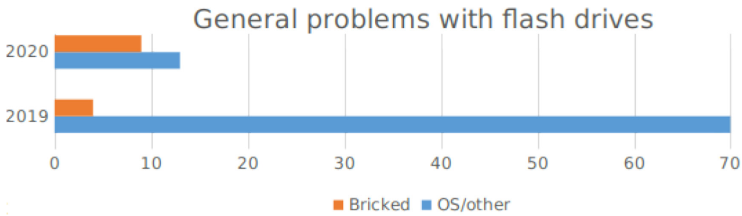


Fig. 7. Number of occurrences of problems that students have experienced with flash drives

Remaining problems were not recorded and were limited to some students that did not have access to any PCs and due to the many labs that forbid booting from flash drives.

6 Feedback

6.1 Language

The next step involved considerations to good first-year programming languages and development environments[12]. Why Python in first semester and Java in second semester? First year at the Institution is a programming course, at its core, and thus it is the norm to build a programming course around a specific language. Since it was geared towards first years, the language and environment

[12] Much of the teaching side of the proposed curriculum are based on the outcomes of the meetings. Whereas, the portable environment was discussed and created "in-house" by the ComSci team.

needed to adequately implement the fundamental concepts to be taught in the course, with an easy learning curve, while keeping up with current trends in the real world. The programming language was decided via a long debate amongst all staff members and the result coincided very well with a survey[13] shown in Fig. 8.

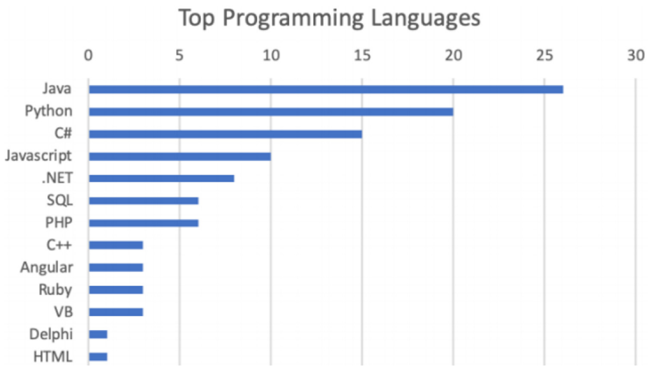


Fig. 8. Programming languages in demand in SA

6.2 Portable Environment

The ComSci team is actively involved during practicals and move between students in tandem with tutors and teaching assistants. This led to acquiring much verbal feedback that could be used to improve the portable environment since 2018. The move from virtual machines to a native portable environment has been positive with many students sticking to the native portable environment throughout first semester as opposed to the virtual one, which is now deprecated. Students also use the portable environment for second semester but only up to a certain point. Both methods have at least partially bridged the gap between the different types of students. Emphasis must, however be place on the latter method involving the use of a native portable environment – it is faster, uses less disk space and is less prone to corruption by unaware students.

Feedback from students has also been very positive on Python. Students that start off as computer illiterate are not intimidated by Thonny. Those who are computer literate were often less enthusiastic about it, but it is interesting that very few of them switch to other IDEs, even though there is no explicit requirement to stay with Thonny.

One of the best features of Thonny is its ability to step through statements and evaluate expressions with incredible verbosity. It is very pleasing to walk through a practical and observe students stepping through code in a line by line

[13] BusinessTech: The top programming languages and tech skills neededin South Africa right now (2019), https://businesstech.co.za/news/it-services/345582/the-top-programming-languages-and-tech-skills-needed-in-south-africa-right-now/.

fashion and seeing exactly how each line is interpreted, evaluated and executed, followed by a eureka moment!

The second semester uses the Java language with the benefit that the majority of code on Android phones are in some form of Java and that it is both in huge demand in South Africa (Fig. 8) and popular. A detailed review of second semester is for future work.

7 Conclusion and Future Work

Continuous improvement through all the forms of feedback is something that appears to have had a positive effect on the first-year students. This has been central to improving the portable environment since 2018. This and other general highlights are summarised in the following subsection.

7.1 Conclusion

The move from virtual machines to a native portable environment has been positive with many students sticking to it throughout first semester. Both methods have at least partially bridged the gap between different types of students.

The teaching curriculum as a whole shows great promise for preparing a first-year student to complete the entire computer science degree. The type of students that only plan to do first year Computer Science may find this course particularly useful for data science, as it provides fundamentals for other problem solving environments too.

7.2 Future Work

While there is some discussion of transformation, especially with respect to the diverse student body, more engagement with global and national higher education contexts can be explored and data captured. The impact of the portable environment on teaching context is also important. Additional data is required for comprehensive statistical analysis.

Secondly, more feedback data from students and peers will help with critical reflection which may enable "closing the loop", that is, acting on the feedback obtained in an explicit way. An example of this is with the COVID-19 crises: feedback on whether the portable environment was helpful during distance learning is important to investigate.

Lastly, investigating the feasibility of cross-disciplinary services that span university wide can be worthwhile.

Acknowledgements. Thank you to National Research Foundation (120654) for funding. This work was undertaken in the Distributed Multimedia CoE at Rhodes University.

References

1. Annamaa, A.: Introducing thonny, a python ide for learning programming. In: Proceedings of the 15th Koli Calling Conference, pp. 117–121 (2015)
2. Biggs, J.: What the student does: teaching for enhanced learning. High. Educ. Res. Dev. **31**(1), 39–55 (2012)
3. Butler, R., Pettey, C.C.: CPVM: customizable portable virtual machines. In: Proceedings of the 44th Annual SRC, pp. 616–619 (2006)
4. Fronza, I., El Ioini, N., Corral, L.: Students want to create apps: leveraging computational thinking... In: Proceedings of the 16th Annual Conference on Information Technology Education, pp. 21–26 (2015)
5. Fry, H., Ketteridge, S., Marshall, S.: A Handbook for Teaching and Learning in Higher Education: Enhancing Academic Practice. Routledge, Milton Park (2008)
6. Giuliano, B.A., Sullivan, J.L.: How do you measure success? J. Coll. Sci. Teach. **34**(3), 41 (2004)
7. Griffiths, D.: #feesmustfall and the decolonised university in South Africa: Tensions and opportunities in a globalising world. Int. J. Educ. Res. **94**, 143–149 (2019)
8. Horton, D., Craig, M.: Drop, fail, pass, continue: persistence in cs1 and beyond in traditional and inverted delivery. In: Proceedings of the 46th ACM Technical Symposium on Computer Science Education. SIGCSE 2015, pp. 235–240. Association for Computing Machinery, New York, NY, USA (2015). https://doi.org/10.1145/2676723.2677273
9. Kennedy, S.M., Hodgson, M., et al.: Subjective assessment of listening environments in university classrooms: perceptions of students. J. Acoust. Soc. Am. **119**(1), 299–309 (2006)
10. Lu, J.J., Fletcher, G.H.: Thinking about computational thinking. ACM SIGCSE Bull. **41**, 260–264 (2009)
11. Porter, L., Guzdial, M., McDowell, C., Simon, B.: Success in introductory programming: what works? Commun. ACM **56**(8), 34–36 (2013). https://doi.org/10.1145/2492007.2492020
12. Rode, J.A., et al.: From computational thinking to computational making. In: Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing, pp. 239–250 (2015)
13. Romero, M., Lepage, A., Lille, B.: Computational thinking development through creative programming in higher education. Int. J. Educ. Technol. High. Educ. **14**(1), 1–15 (2017). https://doi.org/10.1186/s41239-017-0080-z
14. Shipman, P., Bull, R.: Lab on a stick. In: Proceedings of the 16th Annual Conference on Information Technology Education. SIGITE 2015, pp. 33–38. Association for Computing Machinery, New York, NY, USA (2015). https://doi.org/10.1145/2808006.2808034
15. Wagner, P., Johnson, C.: MOBILE. In: Proceedings of the 13th Annual Conference on Information Technology Education, pp. 115–120. Association for Computing Machinery, New York, NY, USA (2012). https://doi.org/10.1145/2380552.2380590

The Importance of Scaffolding When "Building" Information Systems Specialists

Ingrid Siebörger(✉)

Department of Information Systems, Rhodes University, Makhanda 6140, South Africa
ingrid.sieborger@ru.ac.za

Abstract. University students are expected to be able to employ higher levels of cognitive engagement and solve problems. However, as a result of the massification of higher education, there are increasingly more students in university classrooms who tend to adopt surface level learning approaches to their education. The challenge that university teachers face is to find ways to teach which will encourage the students in their classrooms to adopt more deep learning strategies. Consequently, more university teachers are adopting a constructivist approach to teaching and learning in order to encourage greater active participation and deeper approaches to learning from the students However, without instructional support students are still able to adopt surface level approaches to learning and potentially construct misconceptions; thus they are not able to enact their knowledge in a meaningful way in order to solve problems. This paper discusses the use of constructivist teaching together with instructional support (scaffolding) within teaching in Information Systems and notes the importance of including high instructional support verse low instructional support.

Keywords: Constructivist teaching and learning · Instructional support · Scaffolding · Information systems teaching

1 Introduction

University students are expected to be able to employ higher levels of cognitive engagement and solve problems. However, because of the massification of higher education, there are increasingly more learners in university classrooms who tend to adopt surface level learning approaches to their education. The challenge that university teachers face is to find ways to teach which will encourage the students in their classrooms to adopt more deep learning strategies [4]. A deep approach to learning is one where the student has an intention to understand what they are learning and employs a learning strategy to support that. Conversely, a student who has adopted a surface approach seeks to cope minimally with the requirements of the course and typically engages in rote learning/memorization of facts and procedures [2].

G. Wells et al. (Eds.): SACLA 2020, CCIS 1518, pp. 104–113, 2021.
https://doi.org/10.1007/978-3-030-92858-2_7

Mohtashami and Scher [18] note that technical courses in Computer Science and Information Systems are typically taught in an "objectivist model of learning" or didactic instruction, where vast quantities of technical knowledge are imparted from the teacher to the listening learners. Learners are thus passive listeners, typically resulting in surface approaches to learning and knowledge reproduction; the traditional transmission mode of teaching encourages students in our classrooms to remain surface learners and never truly engage with the concepts being covered in order to construct their own understanding.

Conversely, Connolly and Begg [6] like Biggs [4], Hunt, Chalmers and Macdonald [13] and Ashwin et al. [2], argue for a constructivist approach to teaching and learning where learners are encouraged to take part in an active process of constructing their own knowledge and understanding (rather than being passive hearers of the body of knowledge) based on what they already know, coupled with the new information being presented to them by the teacher.

This paper discusses the constructivist teaching and learning theory together with problem-based learning and highlights the value of scaffolding the learning process. It goes further to describe two Information Systems courses, one that adopted a high instructional support approach to scaffolding and one that adopted a low instructional support approach and compares the different set of summative results in each of the courses. The differences are attributed to the varying instructional support and the paper attempts to highlight and argue for the importance of instructional support in teaching in Information Systems when undertaking a constructivist teaching and learning approach.

2 Literature Review

Constructivist learning theory states that people construct their own meaning or understanding (knowledge) of the world around them through experience and reflection. The use of constructivist learning theories in the classroom setting utilizes active techniques like problem solving to help learners build their own knowledge and understanding and then to reflect on their understanding [2,6,12,23,26]. The focus of teaching thus shifts to considering what activities students could be asked to do in order to result in the appropriate levels of learning taking place. Instead of teaching focusing on facts, concepts and principles that need to be understood, the emphasis is on what it means to know and understand the concepts and which activities can be utilized to gain understanding [4,6,7,12,13,24]; the focus is on application of knowledge in order to solve meaningful, real-world problems.

Constructivism has its foundations in the work of Dewey, Piaget, Vygotsky and Bruner [23]. Dewey argued that students learn through building their own knowledge; knowing and doing are closely linked and that it is through action that students are able to put into practice what they know and develop a deeper understanding [7,8,23]. Similarly, Piaget [20] described how children make meaning from their experiences and ideas; they interpret new information in terms of already existing concepts and in turn revise pre-existing concepts in terms of new information or experiences [23,25]. While, Vygotsky [27] focused on how a student's thinking can be influenced by relationships with others who are more knowledgeable [23,25]. Vygotsky emphasized the importance of social learning through what he called the Zone of Proximal Development (ZPD) where the interaction of the student with experts and peers (now also including cognitive tools [14], for example, mind maps, computer games, calculators, etc.) supports them in internalizing knowledge and constructs [23,25]. Learning is thought of like a coach training an athlete - the expert offers the student guidance for practicing new skills or constructing new knowledge such that the student assimilates the skill or knowledge [23]. Similarly to Vygotsky, Bruner [5,29] argued that students could more readily assimilate new skills and knowledge if provided with appropriate guidance and resources. He referred to such support as instructional scaffolding, providing guidance in the right way and at the right time.

Building on the original "approaches to learning theory" [17] which has been influential in changing higher education practitioners' understandings of how different students learn, Biggs [4] describes two types of students: 'Susan', the academically committed, bright, diligent student who is interested in her studies; and 'Robert', who is less committed, less interested in his studies and unless pushed will only put in enough effort to pass. Susan utilizes a deep approach to learning and is self-motivated, while Robert employs a surface approach to learning and only wishes to do enough to obtain his degree. Biggs suggests that the number of Roberts in university classes today is higher than the past as a result of the massification of higher education [4]. He argues that the more active the teaching method and the more engaged students are required to be, the more likely they are to progress from adopting surface learning approaches to using deeper learning approaches [4].

Biggs [4] suggests that one way of encouraging deeper approaches to learning is through employing a problem-based learning strategy because it requires that all students question, speculate and attempt to generate solutions; resulting in more students utilizing higher order thinking skills. Problem based learning (PBL), as defined by Ashwin et al. [2] is a form of teaching where students are given carefully chosen, typically real world, problems to solve while being guided by a facilitator (instructional support) and is becoming increasingly important in STEM education [3]. Problem based learning approaches vary but at their essence they are about tackling ill-structured problems for which there are a number of possible solutions [3]. Belland [3] argues that determining the best means to support students through this process has great potential to prepare students as problem solvers. This approach to learning also lends itself to a constructivist approach to teaching and learning; encouraging the student to do more, while scaffolding the learning and, allowing for social learning through collaboration.

Increasingly, the value of instructional support within learning environments is highlighted across numerous research projects; extending and enhancing the capabilities of students in problem solving [1,3,10,11]. Hardy, Jonen, Möller and

Stern [10] describe two key elements to instructional support, namely (1) carefully structuring the task students undertake in order keep the focus on the important aspects, and (2) supporting and encouraging students to reflect on their insights. They argue that instructional support should assist students to progress a sequential conceptual development where they are engaging in active discovery [10]. Wood, Bruner and Ross [29] describe scaffolding as dynamic support intended to extend a student's abilities by helping to fill the gaps in their abilities and knowledge such that they are able to complete a task. Scaffolding is meant to be temporary support that enables the student to gain independent skills or knowledge [29]. Wood, Bruner and Ross [29] argue through scaffolding the student is able to achieve more than when attempting unassisted, eventually resulting in task competence sooner than had the student progressed completely unassisted. Similarly, Belland [3] describes scaffolding as interactive support that utilizes what the students already know to assist them in meaningfully participating and gain new skills or knowledge which they would not have been able to do unassisted, with a view to being able to perform the new skill independently in the future. Scaffolding is said to require "intersubjectivity", meaning an understanding of what is required to be successful. This aspect is key in helping students to understand when they have accomplished a task successfully such that they are able to perform tasks independently in the future [3,29]. Scaffolding can be provided by teachers, peers and computer-based tools [3,25].

The level at which instructional support is offered can vary between high instructional support and low instructional support. Hardy et al. [10] undertook to investigate the effects of different levels of instructional support within a constructivist learning environment. They maintained the same instructional time, teacher, and teaching materials across two groups of students (in the same grade) and only varied the instructional support offered. The instructional support offered to the high instructional support group was to focus the students on the relevant aspects of the task by carefully sequencing the tasks from basic to more complex and encouraging the students to reflect on their insights. The discussions were deliberately prompted by the teacher by either asking questions or making contrary statements in order to engage the students higher order thinking skills. While the low instructional support group were free to proceed with the material in any order that they wished, and the work was not segmented into smaller conceptual units. Group discussions were far more student centred; students were expected to challenge each other's hypotheses and the teacher offered mainly organizational supervision [10]. Hardy et al. [10] found that students who had received the high instructional support had a more coherent understanding of the concepts they had been studying as compared to their classmates who received low instructional support. Students in the high instructional support group were also better able to transfer their new skills and knowledge to other domains and problems as compared to their classmates who received low instructional support.

Luckett and Sutherland [16] argue that the role or purpose of education is to produce capable individuals, where they define capability to be the "integration of knowledge, skills, understanding and personal qualities" that are used to solve problems that are both known and new/unfamiliar [16, p. 99]. Similarly, Rogoff and Lave [22] and Lave and Wenger [15] hold that learning is a result of activity, context and culture. Wenger [28] further refined his definition of learning to be the development of one's contextualized knowledge, understanding and actions/interactions within a particular community, in order to become full participants within the knowledge or practice community. Arguably, Information Systems and Computer Science graduates need to both know and understand the theory behind the disciplines but also be able to put that theory into practice in order to contribute meaningfully in industry once they leave university; they need to be what Luckett and Sutherland [16] refer to as "capable individuals".

3 High Instructional Support Vs Low Instructional Support

Coupled with the push to utilize a constructivist approach to teaching and learning is the knowledge that in the modern world there is a burgeoning availability of theory and content from YouTube videos and other online resources which students can easily access. This then begs the questions: what can university lecturers offer their students that they cannot get from YouTube and other online resources? The clearest answer is access to a 'master' of the discipline. This reframing of the role of the lecturer is useful and allows one to consider the students as 'apprentices' [2,9]. Furthermore, the concept of the student as the apprentice and the lecturer as the master is at its essence how Vygotsky saw the learning process taking place; the novice or less experienced student learning from the expert [25].

Thus, the Information Systems 3 Database Theory and Structured Query Language (SQL) and the Information Systems (IS) 2 Object-oriented design courses were both reformed from a more didactic approach to teaching and learning to encompassing constructivist teaching and learning practices specifically taking a problem-based learning approach. For both courses the theory components of the courses were compiled into a series of lecture videos together with support PowerPoint slides, relevant textbooks and other online materials in order to facilitate self-study and multiple learning styles. This allowed the lecture times to be freed in order to allow for collaborative problem-based learning exercises with the lecturer and students. This entailed providing typical problems that students might encounter in the real world and then explicitly teasing out problem solving approaches, utilizing theoretical knowledge, which they had access to via videos and slides provided to them via the university's learning management system. In line with the apprenticeship model [2,9] and the principal tenets of scaffolding in constructivist learning [3,25], as the course progressed the students were expected to contribute more to helping to solve the problems being discussed.

The IS 3 course ran earlier in the year than the IS 2 course. During the IS 3 course the course practical sessions were structured to provide high instructional support. The students were broken up into smaller groups and had to work with their tutors collaboratively in order to solve problems posed. Tutors were instructed to help the students problem solve (not give away the answer) and share openly with them their own problem solving techniques in order to promote a collaborative learning community of practice [15,22,25,28]. The IS 3s complained bitterly about having to work on the tutorials together, commenting “The tutors went through questions but we didn’t get to sit in front of a computer and do it ourselves. We couldn’t ask questions about homework questions because more time spent on tutorial questions” and “I had an issue with the structure of the practical. We were put in tutorial groups and focused on work that some students had knowledge of and this meant no time to look at and understand the homework questions. I would have liked it to have more of a practical structure than a tutorial structure where we ask the tutors about questions relevant to our understanding.” to provide two examples. They appeared to resent having to work together and collaborate. Furthermore, students felt that because they were provided with some of the answers in the tutorial (in order that as a group they could focus on understanding the problem solving process as well as how theory is enacted – praxis [21]) that they “knew” the content, one student said “Would prefer to ask questions on things I do not understand and the homework section than be tutored on content that I know.” Not all of the students felt this way, but because of the strong resentment to the idea, the tutorial component of group work and peer learning was dropped for the IS 2 course. In so doing, the practical sessions for the IS 2 s had less instructional support than it did for the IS 3 s; providing less structure and guidance with respect to how to engage in the praxis of the knowledge areas within the IS discipline and only offering problem solving examples during the formal lectures.

The summative performance of the IS 3 students improved significantly from 2018 to 2019; the 2018 averages for the theory and practical examinations were 41% and 71% respectively, while the 2019 averages for the theory and practical examinations were 64% and 73% respectively. The improvement in results reflects a deeper understanding by the students [7] and a shift from surface learning to a deeper form of learning and engagement [4].

The summative performance of the IS 2 students however, significantly worsened. The average for the theory examination for object-oriented design in 2018 was 40.7%, worsening to 34.6% in 2019. The results for the IS 2 students in 2019 had a bimodal distribution; approximately one third of the class did very well achieving an average exam mark in the 60 to 70 percentile, however, the remaining two thirds of the class were in the 30 to 40 percentile or below. The impact of the instructional support in the two courses and the resulting assessment results are discussed in greater details in the discussion below.

4 Discussion

The importance of theory in the discipline of Information Systems has been emphasized since the discipline’s very beginning. Mueller and Urbach [19] explain that theory helps us to understand the how, what and why of various phenomenon; “Such an understanding helps researchers and practitioners to move beyond rote routines [...] and allows them to uncover underlying conceptual structures that enable purposeful and meaningful actions" [19, p. 350]. As such, it is not enough that our students "know" theory, they need to understand in such a way in order to enact it [21] in practice and solve real world problems if they are to offer meaningful contribution as practitioners.

Constructivist teaching and learning approaches place an emphasis on the student to be an active learner and construct their knowledge, but without high instructional support (HIS) students can potentially construct incorrect knowledge constructs, or just avoid taking part in the course altogether in favour of attempting to "cram" the information before the examination (summative assessment). The IS 3 students complained about being coerced into working actively in the Database Theory and SQL course because of the tutorials during the practical sessions together with not being provided with any of the notes or examples covered in class (they were expected to take their own notes). They wished to have greater control of how and when they were learning. However, when this was allowed with the IS 2 students, lecture attendance suffered (roughly 30–40% lecture attendance) and the students did not appear to gain the same benefit from the practical sessions as the IS third year students did, as reflected by the final summative assessment marks. In line with the key instructional support elements as specified by Hardy et al. [10], the third year student practical sessions were more carefully structured to keep the students' focus on the important aspects to be understood, while encouraging them to reflect on their insights through collaboration and group discussion. Furthermore, the tutorial group work with their tutors helped to fill the gaps in their abilities and knowledge such that they were able to complete tasks. This support was later removed in order that they might gain independent knowledge and skills.

Wood, Bruner and Ross described a good tutor as someone who does a good job of scaffolding the student. They noted that "well executed scaffolding begins by luring the child into actions that produce recognizable-for him solutions. Once that is achieved, the tutor can interpret discrepancies to the child. Finally, the tutor stands in a confirmatory role until the tutee is checked out to fly on his own" [29, p. 96].

The IS tutors tutoring in the IS third year database course were encouraged to support the students in their tutorial groups in problem solving. They were encouraged not to merely give away the answers to the questions but to encourage the tutees in their groups to work together with them to solve the problems collaboratively. Furthermore, the tutors were told to share with the students in their tutorial groups their approaches to problem solving and demonstrate to the students how they enact their knowledge. Despite the complaints from some students about not being given enough freedom to decide how and when they wished to work and learn several students found the tutorial experience valuable. Some students commented "The tutors hosting the practical as a tutorial session was more beneficial than tutors invigilating sessions", while another student said that the practicals were useful in preparing them for the exams because they were "made into tutorials and we were able to discuss questions and have solutions". It is worth noting at this point that a number of the students who complained about the tutorials within the practical sessions said they did not like the tutorials because work that they (the tutees) "already knew" was covered. It would appear that students naively equate having access to the answers as "knowing"; they have misunderstood what it means to know in our disciplines and going forward it would be prudent to spend some time making sure that they have a better understanding of what it means to know in the discipline.

The second year students were conversely given more latitude and control over their learning - more student focused as they had to drive the process, particularly within the practical sessions, while tutors and the lecturer were available to field questions and encourage independent thinking. Students were therefore also free to not actively take part in the course, despite the constructivist nature of the course, albeit with low instructional support. The second-year students thus seemed to construct an inaccurate or incorrect understanding of the knowledge area under study and thus were not able to enact their knowledge (correctly).

Through the use of high instructional support students were inducted into the praxis of working with and designing databases, as well as able to interpret real world requests for information, interpret those requests into the required data housed within a relational database and correctly extract the relevant data. This was achieved through a Master-Apprentice approach to the lecture time slots together with peer learning through their tutorial groups during the practical time slots each week - learning from each other and their more experienced tutors - and the use of high instructional support elements. While a number of the students within the third year class did not appreciate the structure of the scaffolding provided in the database course, their deeper levels of understanding, as compared to the second year students who received less instructional support and reflected in the summative assessment speaks to the importance of instructional support in a constructivist learning environment. Problem-based learning approaches in the classroom are not enough, especially in large classes where not all students are forced to engage with the material being studied. Without high instructional support in a constructivist learning environment the "Roberts" (surface learners) in our classrooms are not coerced into employing deeper approaches and thus not employing higher cognitive thinking skills.

5 Conclusion

Considering the argument by Luckett and Sutherland [16] that the purpose of education is to produce capable individuals, who are able to think and solve unseen problems together with the knowledge that more of the students in universities today adopt surface level approached to learning, it is imperative to transform our courses and teaching practices. Constructivist learning theories and approaches such as problem-based learning offer meaningful ways to engage learners and encourage deeper approaches to learning. That said, however, in large classrooms they are not enough to encourage surface level learners to adopt deeper approaches without including high instructional support mechanisms that coerce students into enacting their knowledge in solving authentic problems such that they become capable individuals.

References

1. Alrawili, K.S., Osman, K., Almuntasheri, S.: The impact of scaffolding strategies on higher order thinking and attitudes for saudi middle school science students. Int. J. Educ. Pedagogy **1**(2), 1–7 (2019). http://103.8.145.246/index.php/ijeap/article/view/7028
2. Ashwin, P., et al.: Reflective Teaching in Higher Education. Bloomsbury, London (2015)
3. Belland, B.R.: Instructional Scaffolding in STEM Education. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-02565-0
4. Biggs, J.: What the student does: teaching for enhanced learning. High. Educ. Res. Develop. **31**(1), 39–55 (2012). https://doi.org/10.1080/07294360.2012.642839
5. Bruner, J.: The Culture of Education. Harvard University Press, MA (1996)
6. Connolly, T.M., Begg, C.E.: A constructivist-based approach to teaching database analysis and design. J. Inf. Syst. Educ. **17**(1), 43–53 (2006)
7. Dewey, J.: Art as Experience. Capricorn Books, New York (1958)
8. Dewey, J.: How We Think. Houghton Mifflin, Boston (1998)
9. Falchikov, N.: Rethinking Assessment in Higher Education. pp. 128–143 Routledge, London (2007)
10. Hardy, I., Jonen, A., Möller, K., Stern, E.: Effects of instructional support within constructivist learning environments for elementary school students' understanding of floating and sinking. J. Educ. Psychol. **98**(2), 307–326 (2006). https://doi.org/10.1037/0022-0663.98.2.307
11. Hsiao, E.L., Mikolaj, P., Shih, Y.T.: A design case of scaffolding hybrid/online student-centered learning with multimedia. J. Educ. Online **14**(1), 1–9 (2017)
12. Huang, H.M.: Toward constructivism for adult learners in online learning environments. British J. Educ. Technol. **33**(1), 27–37 (2002). https://doi.org/10.1111/1467-8535.00236
13. Hunt, L., Chalmers, D., Macdonald, R.: University teaching in focus. pp. 21–37. Routledge, London (2012)
14. Jonassen, D.H.: What are cognitive tools? In: Kommers, P.A.M., Jonassen, D.H., Mayes, J.T., Ferreira, A. (eds.) Cognitive Tools for Learning, pp. 1–6. Springer, Berlin Heidelberg, Berlin, Heidelberg (1992)
15. Lave, J., Wenger, E.: Situated learning: Legitimate Peripheral Learning. University of Cambridge Press, Cambridge (1991)
16. Luckett, K., Sutherland, L.: Improving Teaching and Learning in Higher Education. pp. 98–130. Witwatersrand University Press, Johannesburg (2000)
17. Marton, F., Säljö, R.: On qualitative differences in learning: i-outcome and process. British J. Educ. Psychol. **46**(1), 4–11 (1976). https://doi.org/10.1111/j.2044-8279.1976.tb02980.x
18. Mohtashami, M., Scher, J.M.: Application of bloom's cognitive domain taxonomy to database design. In: Proceedings of ISECON (Information Systems Educators Conference) (2000)

19. Mueller, B., Urbach, N.: Understanding the why, what, and how of theories in IS research. Communications of the Association for Information Systems **41**(1), 17 (2017). https://doi.org/10.17705/1CAIS.04117
20. Piaget, J.: The Psychology of Intelligence. Routledge, London (2001)
21. Quinn, L., Vorster, J.A.: Pedagogy for fostering criticality, reflectivity and praxis in a course on teaching for lecturers. Assess. Eval. Higher Educ. **41**(7), 1100–1113 (2016). https://doi.org/10.1080/02602938.2015.1066756
22. Rogoff, B., Lave, J. (eds.): Everyday Cognition: Its Development in Social Context. iUniverse (1999)
23. Seifert, K., Sutton, R.: Educational Psychology. Global Text Project, 2nd edn. (2009)
24. Shuell, T.J.: Cognitive conceptions of learning. Rev. Educ. Res. **56**(4), 411–436 (1986)
25. Tudge, J., Rogoff, B.: Peer influences on cognitive development: Piagetian and Vygotskian perspectives. In: Interaction in human development., pp. 17–40. Crosscurrents in contemporary psychology, Lawrence Erlbaum Associates Inc, Hillsdale, NJ, US (1989)
26. Ultanir, E.: An epistemological glance at the constructivist approach: constructivist learning in dewey, piaget, and montessori. Int. J. Ins. **5**(2), 195–212 (2012)
27. Vygotsky, L.: Mind in society: The Development of Higher Psychological Processes. Harvard University Press, Cambridge (1978)
28. Wenger, E.: Communities of Practice: Learning, Meaning, and Identity. Cambridge University Press, Cambridge (1998)
29. Wood, D., Bruner, J., Ross, G.: The role of tutoring in problem solving. J. Child Psychol. Psychiatry **17**(2), 89–100 (2006). https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1469-7610.1976.tb00381.x

Using Technology to Teach a New Generation

James Connan(✉), Dane Brown, and Caroline Watkins

Rhodes University, Grahamstown 6140, South Africa
{j.connan,d.brown,c.watkins}@ru.ac.za
http://www.ru.ac.za/computerscience

Abstract. Introductory programming courses attract students from diverse backgrounds in terms of ability, motivation and experience. This paper introduces two technological tools, Thonny and Runestone Academy, that can be used to enhance introductory courses. These tools enable instructors to track the progress of individual students. This allows for the early identification of students that are not keeping up with the course and allows for early intervention in such cases. Overall this leads to a better course with higher throughput and better student retention.

Keywords: Introductory programming course · Thonny · Runestone academy · Computer science · Problem solving

1 Introduction

Introductory programming courses are often taken by a diverse group of students. While some students may be computing enthusiasts who have had both formal and informal exposure to computers and even programming, many students may have extremely limited, if any, exposure to computers or programming as illustrated in Fig. 1. This makes teaching introductory programming courses both a challenge and very rewarding.

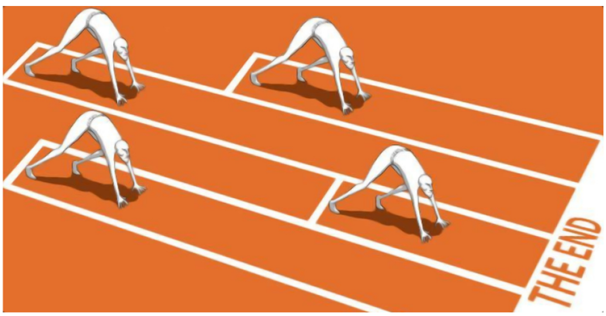


Fig. 1. The challenge of teaching first years is that not everybody starts from the same place.

G. Wells et al. (Eds.): SACLA 2020, CCIS 1518, pp. 114–128, 2021.
https://doi.org/10.1007/978-3-030-92858-2_8

This diversity is amplified in a country like South Africa that has lowered university entry level requirements [5] and has a very diverse schooling system.

This paper looks at the role technology can play in facilitating and managing this process as well as leveling the playing field for all participants in the course. Technology is used to quickly identify students who are not keeping up and who may need additional assistance. Interventions can then be arrange to prevent them from falling behind.

The technology also mitigates against the effects of changing lecturing personnel regularly. Practical matters such as staff sabbaticals, resignations, retirements, illness and scheduling conflicts means that it is very unusual to have the same lecturing personnel presenting the same content in consecutive years. The structure provided by the technology negates many of the negative impacts normally associated with these changes.

The remainder of the paper is organised as follows: Sect. 2 provides and overview of the environment that is used by investigating the constraints on placed on course by those who enrol in it while Sect. 3 and Sect. 4 introduces the two tools that are used to better achieve this. Section 5 looks how these technologies are applied in the laboratory environment. Section 6 discuss ways to analyzes the impact that this has had on the course and Sect. 7 will present the results. Section 8 concludes the paper and outlines future work.

2 Programming Environment

The choice of programming language and environment can be both controversial and problematic. Every instructor or student has their preferred language. This can lead to endless debates about language selection.

Rather than examining languages and debating their strengths and weaknesses, language selection for an introductory course can be done by selecting a language that meets the needs of the students in the course.

2.1 Students

Students who typically subscribe to introductory courses have diverse motivations for doing so. As a first step towards a formal Computer Science degree, many of the students who enrol for introductory courses do so with the intention of doing a full Computer Science degree. Some of these students will have done either a programming course or some sort of computer literacy course at school, but some of the students in this category will have had very little to no exposure to computers.

The next group of students are those who do not intend to continue with Computer Science, but would like some exposure to programming before continuing into other fields. This group may also contain a group of students who are misinformed about what exactly an introductory programming course is and may mistake it for a computer literacy course.

The last group are often senior, sometimes postgraduate, students, who have realised that they need some exposure to programming in order to do their studies in fields outside of Computer Science. This may include students from backgrounds as diverse as Linguistics and Bioinformatics.

Ideally, the introductory course should offer something to each of these groups. The Computer Science student should get a solid foundation on which to build the rest of their Computer Science education. The other two groups should get enough exposure, in terms of depth and breadth, that they can write basic computer programmes and explore the field to discover more knowledge about programming, should they feel the need.

2.2 Content

Both the programming language and the content of the course are influenced by the motivation students have to enrol for the course. The course needs to teach the fundamentals of programming and problem solving in order to prepare those students who want to go on to complete a Computer Science degree. It needs to be comprehensive enough that those who will only ever do this one course in programming leave with enough knowledge to be able to write basic programs and it needs to give those students who have come in from other areas of study with enough tools to enable them to better perform their tasks.

The course therefore focuses on five basic principles of programming: input, output, branching, repetition and calculations. While it might touch on higher order concepts such as objects, data structures and graphical user interfaces, these are not core concepts within the course and the focus is on teaching students to solve basic problems and to implement solutions for those problems using the five basic concepts.

2.3 Platform

The last consideration for selecting the programming environment is the platform to be used. As stated previously, the experience and exposure that students have when they enrol for an introductory course is very diverse. This can lead to quite an imbalance in both the lecture venue and practical laboratory sessions. Those students who have had limited exposure to computers can feel very intimidated by those who have had a lot of exposure. One way to limit the impact if this is to put everybody into a new environment. This reduces the gap between those who do not have exposure and those who have been formally and/or informally exposed to computing technologies.

Students are therefore put into a GNU/Linux environment. Those already familiar with Microsoft Windows are exposed to a new environment with their peers and those who have not been exposed to any computing environments do not feel intimidated because their peers seem to "just know everything".

This has the further advantage that it allows an exact duplication of the environment to be given to each and every student without any concerns about proprietary licenses. The student can therefore duplicate the exact environment used in the classroom and laboratory on their personal machine, if they have one.

While great strides have been made in releasing free student versions of various Integrated Development Environments (IDE), it still often happens that the student version is not identical to the licensed version used in the laboratories. While it would seem simple enough to use the student edition in the laboratories, there are practical reasons while this is sometimes not practical.

Students are therefore provided with a flash drive that contains a GNU/Linux environment with the exact same software that is used in the laboratory. This is done as either a virtual machine or as a bootable usb drive. The exact nature of this environment is the subject of another paper by the same authors.

2.4 Language

The language selected for this course is Python [9]. Python is a relatively simple interpreted language that is widely used (nationally and internationally) for scripting and rapid prototyping. This popularity is illustrates in Fig. 2. It is used in both the scientific and non-scientific community. This means that there are libraries available for many different fields of study, ranging from the Natural Language Toolkit (NLTK) [7] for linguistics to the scikit-learn [8] for machine learning.

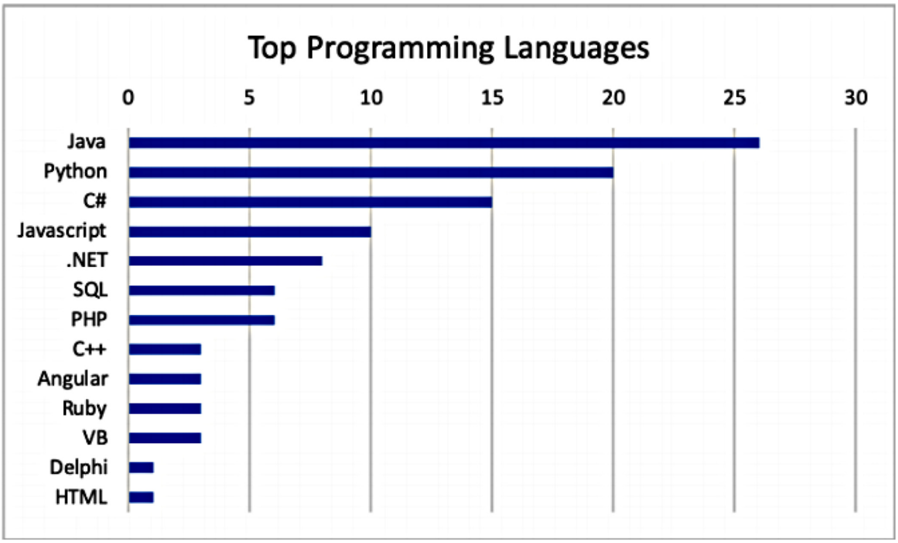


Fig. 2. Programming languages in demand in SA [3].

A student that therefore only does this one programming module with the aim of applying programming to a different field is likely to find Python libraries to support their field of study.

Python has simple syntax that is easily learned by novice programmers, both native English speakers and those with different home languages. It is a dynamic language, which means that it does not require users to declare variables and specify their types before using them. From a teaching perspective, this can be both an advantage and disadvantage [6]. It uses indentation to specify blocks and nesting. While this can be frustrating for experienced programmers, it does force novice programmers into some good habits.

As an interpreted language, Python is slow, but very portable. It is supported on most platforms and devices and there are many web based compilers for it, meaning that it can be used without installing any additional software on the local device. This means that code can be produced on almost any device, though the process might be quite cumbersome. It is not unusual to be approached by a student who has a question with a code snipped running on their smartphone.

From the perspective of a Computer Science student, it supports a lot of modern programming concepts and importantly it has very good support for the five basic principles of programming: input, output, branching, repetition and calculations.

3 Integrated Development Environment

There are many good IDEs available to programmers, both in the commercial space as well as in the Open Source space. Developers are spoilt for choice and companies with commercial interests in specific programming languages have realised that providing a good development environment is almost just as important as an efficient compiler. There are therefore IDEs that can assist in making commercial programmers more efficient and increasingly, these IDEs have productivity tools integrated into them.

Education is not training. It is much more. There is a lot of value in providing training in a production grade IDE that can be used in a commercial setting. For education purposes though, it is more important that the IDE, should one be used, aids learning and understanding over productivity. While a specific IDE might be useful in improving the productivity of a commercial programmer, it does not follow that the same environment aids understanding of the programming process. In the strive for increased productivity, some IDEs automate tasks that the programmer would normally do. Using such an IDE might mean that the novice programmer is never exposed to these tasks, hindering the learning process.

For this reason, an IDE was selected that aids learning rather than productivity. The IDE needs to enhance the learning experience of the learner, without overwhelming the novice.

An important observation that has been made both in the laboratory as well as in the feedback from students was that those who were the least computer literate upon starting the course, found comprehensive production quality IDEs the most intimidating. Rather than assisting in the learning process, these IDEs obscure many of the key concepts that novice programmers need to learn.

Many IDEs were investigated with the aim to find one suitable for an introductory course such as this. While they all have strengths and weaknesses, it is not always possible to come up with an objective measure of how good a particular IDE is. The IDE that was therefore selected is not necessarily the best IDE ever produced or even the best IDE for learning to program. It is an IDE that is well suited to novice programmers and that meets the objectives of this course.

Thonny [1] calls itself "The IDE for beginners" and is developed by the University of Tartu in Estonia. It is developed specifically for teaching Python.

Thonny runs on Windows, Linux and Mac and is free. Its features include:

- A Clean and simple Graphical User Interface
- Variable visualisation
- A simple debugger
- The ability to step through code
- Expression evaluation
- Faithful representation of function calls
- Syntax error highlighting
- Scoping
- A mode to explain references
- Code completion
- Integrated shell
- It is also extendable and supports the Python package installer pip

What it does not have is breakpoints, though those are currently under development. This means that the only way to step through code is to step through the entire program, though it does support "step in" and "step out", which means that the user can avoid certain chunks of code.

Thonny is not intended to be a production class IDE and will probably never be. It is a very good teaching tool.

Feedback from students has also been very positive. Students that start off as computer illiterate are not intimidated by Thonny. Those who are computer literate are often less enthusiastic about it, but it is interesting that very few of them switch to other IDEs, even though there is no explicit requirement to stay with Thonny.

One of the best features of Thonny is its ability to step through statements and evaluate expressions. Students can therefore observe how the code is executed on the machine as illustrated in Fig. 3. It is very pleasing to walk through a practical and observe students stepping through code in a line by line fashion and seeing exactly how each line is interpreted, evaluated and executed.

4 Interactive Textbook

The content for the course has evolved over several years. This led to the adoption of an openbook, "How to think like a Computer Scientist [4], five years ago.

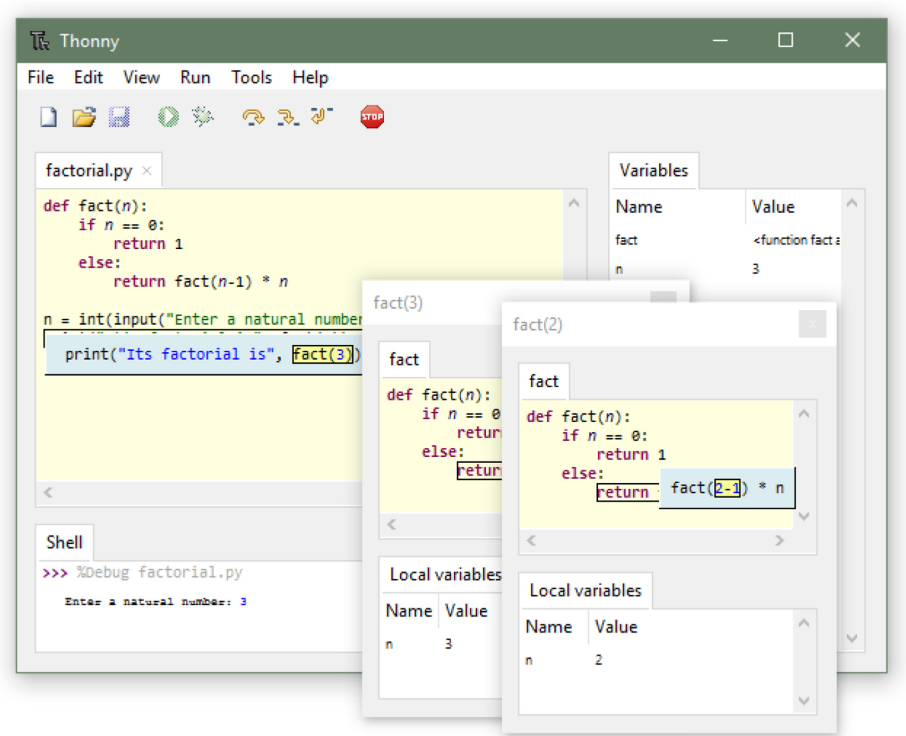


Fig. 3. A screenshot of the Thonny IDE stepping through a recursive function.

As this book is open, it was modified to fit the course content. The original book was for the Python language, but it has been translated [10] into other languages and was used with `C#` for an earlier iteration of this course. Many of the edits and changes made to the book have made their way into the current version of the book.

However, five years on this textbook has also evolved, and thanks to the efforts of Brad Miller at Luther College it is now a fully fledged interactive textbook.

Runestone Academy provides a framework for presenting interactive courses. An instructor can create a custom online course that is only available to their class. When a student points their web browser at this course, they are presented with an interactive online version of the book.

Each chapter generally starts with a short video explaining some of the key concepts of the chapter. Students can work through the content of the chapter. When they get to coding sections of the chapter, the code can be written and executed in the web browser. The student can therefore get real time live feedback on what they have done. This is illustrated in Fig. 4.

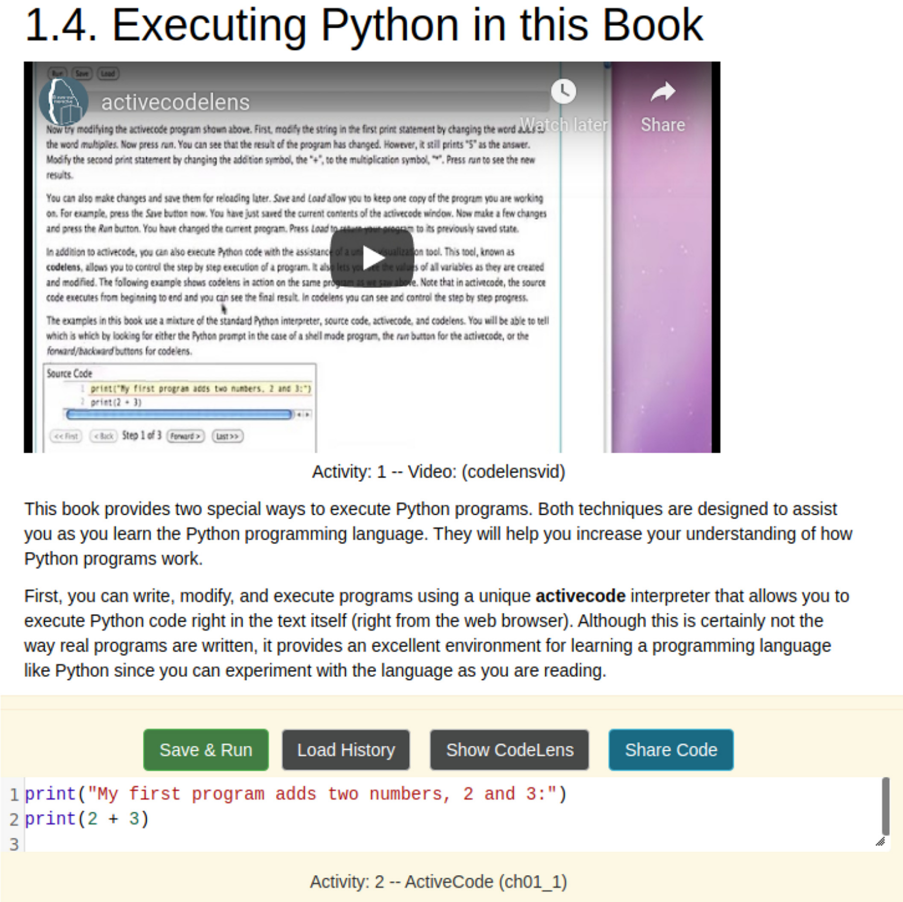


Fig. 4. A screenshot from Runestone Academy showing a video that introduces the chapter as well as an example of executing code in the book.

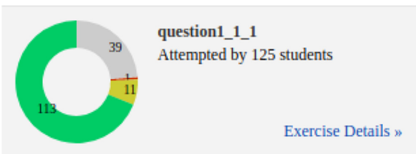


Fig. 5. A summary of a test performed in Runestone Academy.

Some chapters also contain small tests to test some of the concepts covered. Student engagement and competence can me seen at a glance by looking at the results summarised in a little chart as shown in Fig. 5.

The entire system is open source, available on GitHub and customisable.

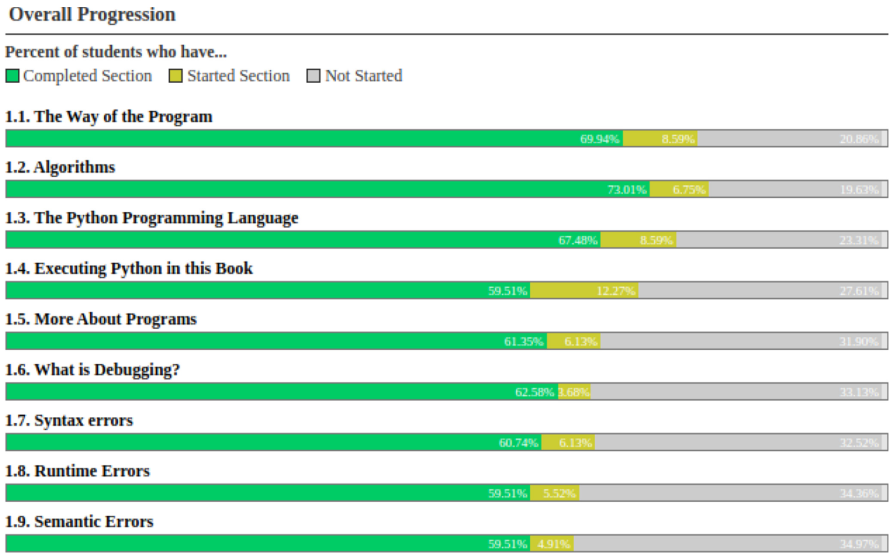


Fig. 6. Runestone Academy progress summary for class.

From an instructor's perspective, the system not only allows for customization, but also logs both individual student activity as well as a summary for the class as illustrated in Fig. 6. This allows the instructor to track student engagement. It is possible to see how many students have read a section or attempted an activity as well as which activities and sections each student has engaged with.

Students are also provided with a pdf version of the book and have the option to print and bind it.

5 Application of the Technology

The application of the Thonny IDE is straightforward as it is used in both the classroom and laboratory to step through code and illustrate how the code is parsed, expressions evaluated and instructions executed. The many different views allows for students to explore code execution well beyond the scope of this course.

Runestone Academy is used to track student progress. This is done in two ways. The instructor can use at anytime look at a student's progress. They can also look at the class' progress. If the class or individual students are not keeping up with the course, this can be addressed.

However, this is most useful in the laboratory environment.

5.1 Academic Development

The Academic Development Programme (ADP) is integrated into the planning and structure of the course. If a problem is identified that affects a group of students, there are ADP lectures and individual tutoring.

For practicals the ADP the system on its head. People who get over 75% are removed from the practical and do their practical in a separate venue with a floating tutor. The rest of the class share a venue and tutors. Thus everybody is automatically in ADP.

This has several advantages. Students who perform well feel like they are being rewarded. More tutors are available to assist the rest of the class. Students who struggle do not feel isolated from the class.

Without the top performers in the venue, the difference between remaining stronger and weaker students is not as big. This means that the weaker students are not as easily discouraged.

Feedback from students has been very positive on this.

5.2 Practicals

Students are expected to have the Runestone Academy site open at all times during practicals. This means that tutors or lecturers that assist students can quickly and easily assess whether the student has engaged with the relevant section of work. Readings can be set and enforced. This has resulted in a significant increase in engagement with the textbook and has made it possible to quickly and accurately identify students that have not engaged with the relevant work.

6 Evaluating the Effectiveness of the Enhancements

Brené Brown [2] starts her Tedx talk titled The Power of Vulnerability with a quote from one of her mentors.

If you can not measure it, it does not exist.

It is difficult to measure how well a course works. Traditional measures might include pass rates, class averages, student satisfaction, peer assessment, enrollment numbers and trends, and many more. While all of these measures can provide valuable feedback to the instructor, they are all open to interpretation, abuse, manipulation and debate.

Course evaluation are often forced on instructors and students as part of review processes, but they may also be used by instructors that are genuinely trying to improve their courses, both in terms of content and delivery. In both cases there are strong arguments to be made for and against some of the performance measures already mentioned.

This paper will however also propose a novel way to measure the effectiveness of a course. Instead of looking at the course itself, it will track the progress of students until graduation in an attempt to see if they were adequately prepared.

If it is difficult to measure how well a course works, it is even more difficult to measure the impact of individual components within the course. Realising that these measures are all limited, especially in terms of objectivity, there are some trends that have emerged from all of them.

7 Results

Assessment for this course includes tests, practicals, a practical exam and a theory exam. There are three lecturers involved in delivering the course and each assessment is peer reviewed by all of them. Exams are also internally reviewed by someone not involved in the course. This reviewer is also asked to classify each question as being easy, medium or hard. Each exam strives to have 40–50% easy questions that students should be able to answer by just learning the theory, 40–50% medium questions where students need to be able to apply knowledge and not just recall it, and 10–20% hard questions, that really test a thorough understanding of the concepts and work covered in the course.

The changes mentioned in this paper were first introduced two years ago. Looking back to over the last four years, the class averages have not changed significantly. The two years preceding the introduction of these new technologies had averages of 70% and 67% respectively. Since changes, the averages have been 68% and 66%. This shows that these changes have not had a clear negative impact on the course.

Pleasingly, the class marks (50% tests and 50% practicals), the practical exam marks, the theory exam marks and the overall average for the course are all very similar at 60% (classmark), 64% (practical exam), 67% (theory exam) and 66% (final mark). This means that students are doing similarly in all forms of assessments and therefore getting comprehensive coverage of the curriculum.

Pass rates have also remained steady, showing a slight increase from 73% in 2017, 72% in 2018, 83% in 2019 and 78% in 2019. This is despite a significant increase in enrollments from 82 in 2017, 81 in 2018, 93 in 2018 and 126 in 2019.

The increase in enrollments for the course is well ahead of the institutional and faculty average.

The proportion of students retained into the second year has also increased slightly.

Figure 7 shows one cohort of 95 students moving through a Computer Science degree programme. Each student is represented by a unique colour. Points that do not have lines continuing from them indicate students that either failed or opted not to continue with Computer Science. In this cohort, there were 95 students in CS1, 46 in CS2 and 37 in CS3. As stated on Sect. 2.1, part of the purpose of this course is to prepare students for a degree in Computer Science. The thick yellow line in Fig. 7 shows the class average for each year and the long yellow line shows the trend line for the class average over the three years. From Fig. 8 we can see that almost all students who pass CS3 score at or above the class average for CS1. Closer inspection shows that 100% of students from this specific cohort that scored at or above the class average for CS1 and attempted

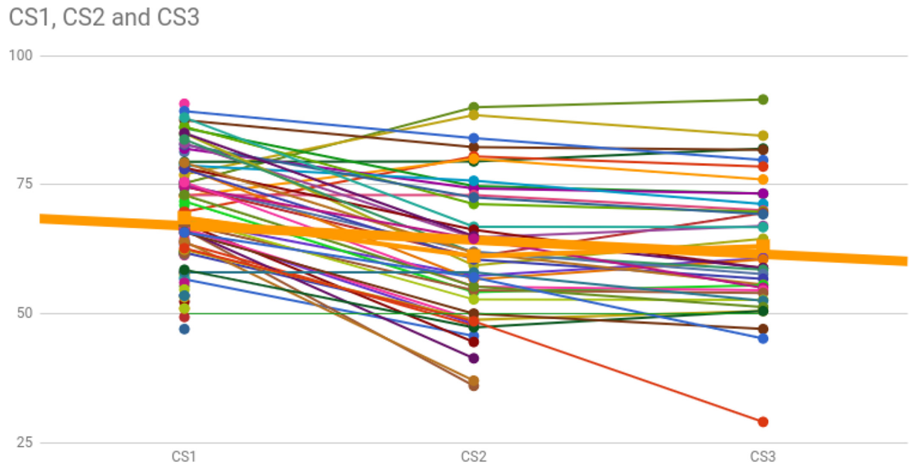


Fig. 7. Mark deviation as the same students (colour lines) progress through first, second and final year.

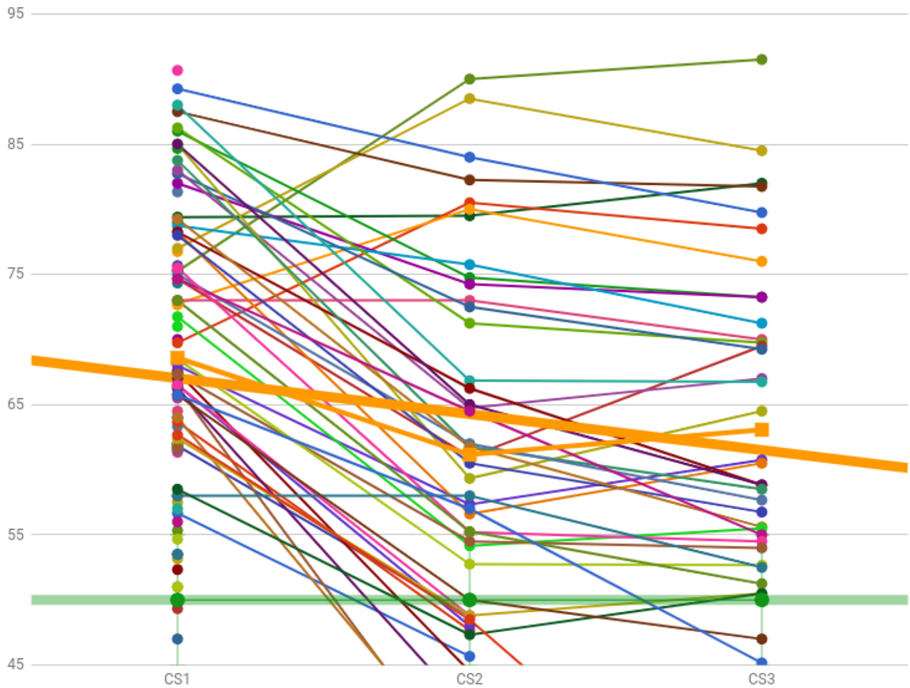


Fig. 8. A closer look at the students that went on to do and pass CS3

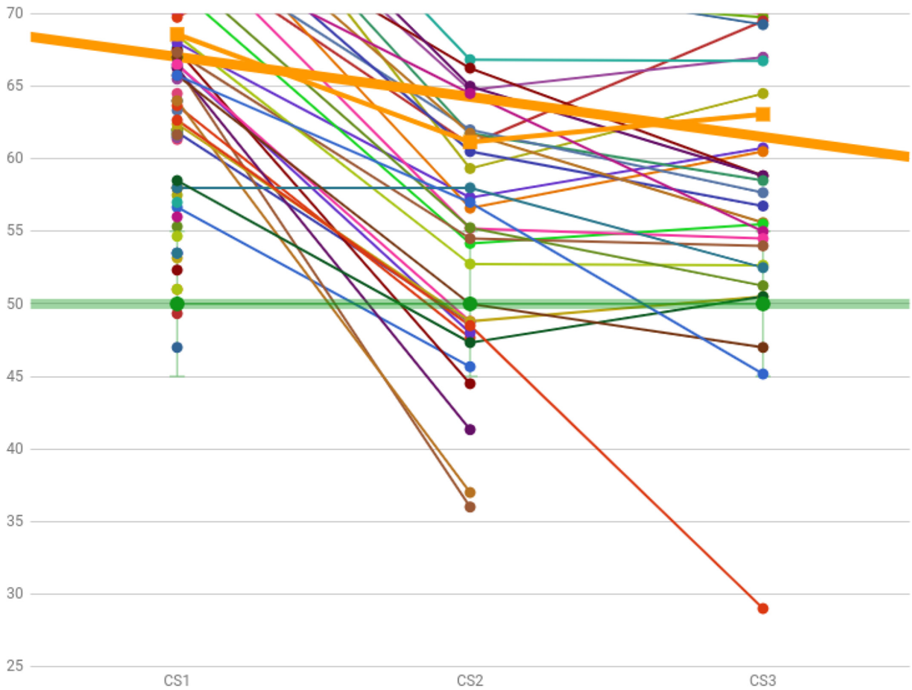


Fig. 9. A closer look at the students who scored below the class average for CS1

CS3 managed to pass CS3. Of those who passed CS1 but scored less than the class average (Fig. 9), only 16% went on to pass CS3. This needs to be investigated further.

8 Conclusion

Integrating more technology into this course has made the management of the course better and more efficient. It has had small but significant positive impact on the students and their marks, pass rates and enrollments for the course. Current enrollments have shown a strong upward trend, above the faculty average. It has also resulted in more students being retained into subsequent courses.

The technology allows the quick and early identification of students who are struggling or not engaging with the course material. This in turn allows for early intervention which means that students can get assistance before they fall behind too far. Identification and intervention does not currently automatically lead to success. How to effectively intervene is a topic that is currently being investigated.

It is highly recommended that course managers in charge of introductory courses seriously consider this or similar technologies to help with the management and delivery of their courses.

Instead of looking at how students perform in the current course, it might be more effective to use performance in subsequent courses to evaluated the effectiveness of the current course. A good course needs to provide adequate coverage of material and challenge students to perform at the correct level to prepare them for what lies ahead.

A course that can give students who score at or above the class average a high degree of certainty that they will be able to complete subsequent courses needed for their degree provides adequate coverage of concepts at a sufficient level to prepare students for these courses.

Students failing courses is a waste of resources. Ideally, all students that register for a course should therefore be able pass the course. For CS1 it can be seen from Fig. 9 that this ideal was almost achieved as very few students failed CS1. This is a notable result, given the heterogeneous nature of CS1 classes.

It is debatable whether all students who enroll for a degree can or should complete that specific degree. Once students have done an introductory course such as CS1, some may opt not to continue with a degree in Computer Science. Others may not have engaged with the material at a sufficient level to gain a proficiency that adequately prepares them for future courses in Computer Science. It may be in the best interest of these students to either repeat CS1 or to be encouraged to enroll for a different degree. These are major decisions that need to be informed by both qualitative and quantitative evidence. This study aims to eventually provide the evidence. This was the first cohort investigated. Preliminary investigations into subsequent cohorts are showing the same trends discussed here. Once validated, the sample set should exceed 500 students which will allow for objective conclusions.

Tracking student progress therefore seems like a good measure that can be used to objectively evaluate the effectiveness of a course and also to fine tune a course to meet the course objectives.

9 Future Work

The developers of Runestone Academy are currently looking at integrating unit testing into the system. This can be done in either an explicit or transparent way. Integrating unit testing into both Runestone Academy as well as Thonny will improve both products, especially if this can be done in a transparent way.

Another useful feature to add to Thonny would be a tool that counts how many compilations are made and how many of them are successful. It is sometimes difficult to know whether a student is actively or passively failing when sitting in a practical. Being able to analyse compilation records will help to assess this.

Lastly, using cohort analysis to objectively evaluate courses shows a lot of promise and is definitely worth pursuing further.

References

1. Annamaa, A.: Introducing Thonny, a Python IDE for learning programming. In: Proceedings of the 15th Koli Calling Conference on Computing Education Research, pp. 117–121 (2015)
2. Brown, B.: TedX: The Power of Vulnerability (2010). https://www.ted.com/talks/brene_brown_on_vulnerability?language=en. Accessed 2019
3. BusinessTech: The top programming languages and tech skills needed in South Africa right now (2019). https://businesstech.co.za/news/it-services/345582/the-top-programming-languages-and-tech-skills-needed-in-south-africa-right-now/. Accessed 2019
4. Elkner, J., Downey, A.B., Meyers, C.: How to Think Like a Computer Scientist: Interactive Edition (2019). https://runestone.academy/runestone/books/published/thinkcspy/index.html. Accessed 2019
5. Griffiths, D.: #FeesMustFall and the decolonised university in South Africa: tensions and opportunities in a globalising world. Int. J. Educ. Res. **94**, 143–149 (2019)
6. Holkner, A., Harland, J.: Evaluating the dynamic behaviour of python applications, vol. 91, pp. 17–25 (2009)
7. Loper, E., Bird, S.: NLTK: the natural language toolkit. In: Proceedings of the ACL Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics. Association for Computational Linguistics, Philadelphia (2002)
8. Pedregosa, F., et al.: Scikit-learn: machine learning in python. J. Mach. Learn. Res. **12**, 2825–2830 (2011)
9. Van Rossum, G., Drake, F.L., Jr.: Python tutorial. Centrum voor Wiskunde en Informatica Amsterdam, The Netherlands (1995)
10. Wentworth, P.: Think Sharply with C# (2015). https://www.ict.ru.ac.za/Resources/ThinkSharply/ThinkSharply/index.html. Accessed 2019

Opening Pandora's Box: An Active Learning Approach to Teaching Project Management

Marié Hattingh(✉), Komla Pillay, and Sunet Eybers

Department of Informatics, University of Pretoria, Private Bag X20, Hatfield 0028, South Africa
{marie.hattingh,komla.pillay,sunet.eybers}@up.ac.za

Abstract. Given the global skills requirement for competent project managers, Higher Education Institutions are forced to adapt their pedagogical practices in order to deliver "industry-ready" graduates. A number of approaches have been adopted by HEI in preparation for competent graduates such as self-directed learning, problem-based learning, collaborative work or active learning. This paper reports on an active learning approach that was taken to teach second year, information systems design (ISD) students' project management skills. This is in preparation for the capstone project, a collaborative turnkey project for a real-life client, where it was noticed that students struggle with project managing the project, their other academic commitments and life. This study adopted an active learning approach by constructing a project management tutorial in such a way as to simulate challenges students might experience when completing their capstone project. Students, in groups of five, each received a box of 20 challenges that they first had to prioritise individually, and then as a group. Students were asked to complete three tutorial questions following this activity which were thematically analysed. They were also asked to complete a short survey on the principles of active learning. The findings suggested that an active learning, simulation based approach was valuable to the students and exposed them to real-life project management challenges. Students were able to identify key activities that caused strain in setting up a common schedule. Tools/techniques that were used to compromise in setting up a common schedule are communicated. Lessons learnt by the students and reflection from the lecturers provide recommendations to ISD lecturers on teaching project management.

Keywords: Active learning · Information systems design · Undergraduate teaching · Simulation-based teaching

1 Introduction

The global skills gap is a phenomena that refers to the deficiency between the skills required by employers and the skills (or lack thereof) of young graduates [1]. This gap is estimated to grow as the skills demanded by the introduction

G. Wells et al. (Eds.): SACLA 2020, CCIS 1518, pp. 129–144, 2021.
https://doi.org/10.1007/978-3-030-92858-2_9

of the Fourth Industrial Revolution is expected to grow [1]. Many researchers have proposed practical approaches towards addressing this problem such as self-directed learning [2], problem-based learning (also referred to as case-based learning) [3,4], collaborative or group work [3] and technology-enhanced learning [3] (to name a few). In the majority of studies the need is for tertiary education institutions to supply in demand, 'work ready' young graduates. Given the old, passive instructional approaches followed by tertiary educational institutions, this seems like a challenging demand to meet.

Constructivism is classified as a traditional learning theory that refers to learning as an active process during which learners construct their own, new ideas based on their current or past knowledge and experience [3,5]. The learner is actively involved in the learning process, as opposed to passively listening. One method often adopted by educators to actively involve learners is active learning. As a result, students engage in the classroom with fellow learners with the objective of solving practical problems. Learners often acquire new knowledge and skills without realizing it. The end result is the construction of their own knowledge.

Studies focusing on constructivism, in particular active learning approaches, are not just used in IT education. In fact, research on the topic is interdisciplinary applied in many faculties such as cognitive psychology, physiology, educational psychology, and learning sciences [3].

A major challenge for information systems design (ISD) undergraduate students is the application of their theoretical knowledge to the management of project constraints i.e. applying theory to practice. Students must learn to be able to systematically identify constraints in a project environment and be able to decisively plan how to manage and schedule the constraints. Typically the opportunity for students to develop these attributes before undertaking their third year capstone project are limited. Simulation can assist students develop the required attributes. The aim of this study was to evaluate the use of simulation-based teaching in the ISD second year undergraduate curriculum in the context of project management of constraints using active learning's decision-making activity and case based learning. In a similar study, a research project titled "Quality of Norwegian Higher Education: Pathways, Practices and Performances" highlighted the importance of simulation in the higher education sector. The study focused on how simulations are designed and used and subsequently outlined various different ways for simulation to be carried out. For example, healthcare made use of high-tech mannequins, education management made use of role playing for interviews and the simulation for law included the creation of a stage courtroom with students role playing the various officers of the court [6]. The main aim of simulations in the Norwegian study, as well as this study, was to bridge the gap be-tween theory [7] and practice which was the main aim for this research.

In this study, the active learning is adopted to expose second year learners to practical project management concepts. Project management as a discipline is known for its dynamic, agile requirements demanding more than just theoretical knowledge of the key principles of managing scope, time, budget and quality. In an attempt to provide structure to the research question, namely - how effective is the adoption of an active learning approach (in particular case based simulation) towards bridging the gap between the project management theoretical concepts and the practical application thereof in second year students - five principles of active learning, as proposed by Michael [3] was adopted. The five principles are discussed in Sect. 3.

2 Literature Review

2.1 Active Learning

Active learning focuses on the fostering of students skills as opposed to the passive transmission of information. The student must engage on a higher level of thinking which is developed through the immersion in an activity. Students are encouraged to explore their own opinions and tenets. Active learning can range from modest class tasks like briefly stopping the class to enable students to dis-cuss with their peers to more involved tasks like immersing students into a case study [8]. Brame [9] states that providing an environment for students to think about their own learning process facilitates the linking of action and learning. Active learning has demonstrated its success in many disciplines and has proven to foster an inclusive environment for student learning. Freeman et al. [10] conducted a meta-analysis of 225 studies in STEM disciplines. The studies examined the design of class lessons with at least some active learning versus traditional lecturing. The meta-analysis compared the failure rates and student scores on examinations, concept inventories, or other assessments. The findings revealed that students in traditional lectures were 1.5 times more likely to fail than students in courses with active learning. The results were consistent across the disciplines of biology, chemistry, computer science, engineering, geology, math, physics, and psychology. Another research project involved the analysis of 166 studies by Ruiz-Primo et al. [11]. Published studies were examined to determine the effects of active learning approaches in undergraduate biology, chemistry, engineering and physics courses [11]. The finding indicated that active learning approaches improved student outcomes, although there are important limitations to deliberate.

Brame [9] collated the work of various authors to summarize the active learning techniques and activities that can be used in higher education lectures. Some of these techniques include the pause procedure, retrieval practice, demonstrations, think-pair-share, peer instruction, minute papers and strip sequence. Table 1 briefly summarizes these techniques. This study used the active learning techniques, decision-making activity and case based learning. The decision-making activity for this research required the students to imagine they are working on a project in a five member team. Each group member had to justify their decision and explain their reasoning for arriving at their prioritized set of constraints. The group then had to critically discuss and formulate a group prioritized list of constraints. This engaging technique assisted students to critically analyse a perplexing problem and encouraged them to find creative solutions. This provided students with a feel of real-world challenges and complexity involved in resolving conflict. The research can also be categorized as the active learning technique, case-based learning. Similar to decision-making activities, case-based learning immerses students in situations from the "real" world that require students to apply their understanding to make a decision. The students were provided with a case, they had to then decide what is relevant to the case and what other information they may need. They considered what impact their decisions may have taking into account the broader implications of their decisions. The best value obtained from case-based learning is from the complexity and myriad of answers that were generated.

Table 1. Active learning techniques

Technique	Brief description
The pause procedure	Pause for two minutes every 12 to 18 min, encouraging students to discuss and rework notes, question and clarify in pairs
Retrieval practice	Pause for two or three minutes every 15 min, allow students to write everything they can remember from the 15 min of class time. This method encourages students to retrieve information from memory enabling them to digest the content before proceeding to the next level
Demonstrations	Ask students to predict the result of a demonstration, briefly discuss it with their peers. After demonstration, ask them to discuss the actual result and how it may have differed from their prediction. The lecturer can thereafter provide an explanation
Think-pair-share	Ask students a question from the application, analysis, and evaluation levels of Bloom's taxonomy. The students must write down their answer for one minute and thereafter discuss the answer with their peer for two minutes. This will ensure the students think critically about their answers
Peer instruction	This is a modification of the think-pair-share using technology (e.g., clickers). Pose a conceptually based multiple-choice question. Ask students to think about their answer and thereafter respond. They can then discuss with a peer and possibly change their answer. Following this, the lecturer shares the results and discusses with the class
Minute papers	Ask students a question that requires them to write a reflection on their learning for one minute. Lecturers can use responses to inform future classes
Strip sequence	Give students the mixed up steps of a process on pieces of paper. They must work together to reconstruct the proper sequence fostering logical thinking

2.2 Project Management

A project is a set of activities with a defined scope and resources aimed at achieving an objective. Project management is the application of knowledge, skills, tools, and techniques to project activities to meet the project requirements [9]. Successfully realizing the project objective can be constrained by many factors, scope, quality, budget, time, risks, customer and stakeholder expectation [10]. Many projects have under-performed by exceeding costs or not meeting expectations. The core function for project management is to manage resources within the constraints encountered in the project environment [11]. For this reason, this study chose to develop a simulation exercise on project constraints. Goldratt (1990; cited in [12]) put forward the theory of constraints which is a systems thinking approach, that a complex system at any point in time has one or more constraints that restrict it from achieving its goal. Project management which is relevant across industries and sectors use this theory extensively. More specific to project management is the theory of triple constraints which states that the triangle of time, cost and scope must be in balance in order to achieve the project goal. The trade-off includes increased scope resulting in increased time and/or budget, decreased time resulting in decreased scope and/or increased budget. The cause and effect of the triple constraints are negotiated and trade-offs made throughout the life of the project.

Project scheduling is a key management activity if not the most crucial. Scheduling specifies the timeline for project completion, the budgeting of the resources and the sequencing of the tasks to be accomplished. Project scheduling is defined as the process of ascertaining when project tasks will take place based upon circumscribed periods and industry standards. Schedule constraints involve specifying when an activity should begin or end, based on attributes such as resource accessibility, target dates or other time constraints. Project scheduling is a multifaceted and iterative process that involves determining dependencies be-tween tasks to ensure the right order is scheduled, planning realistic start and end dates and determining the tasks that are critical for successful project completion [9]. This study explored the challenges experienced by students when setting up a project schedule.

3 Theoretical Foundation

Evidence suggests that active learning can be successfully applied across disciplines [3,13]. Seminal work conducted by Michael and Modell (2003; cited in [3]) proposed generic principles towards successfully adopting an active learning approach. These principles were considered in conjunction with research on the topic of constructivism by Olusegun [5], focusing on active learning. The five principles are:

Principle 1: (a) "learning involves the active construction of meaning by the learner." This can be achieved by employing self-organizing project teams where the focus shifts to self-governing teams. In this approach, team members take psychological ownership to complete tasks [14]. This is further supported by the constructivism theory whereby the lecturer acts as facilitator rather than instructor [5]. (b) "If previous knowledge is flawed then it will take long-er to fix errors." Individual team members' personal frame of reference, which are formed by previous exposure to a project management environment, are imperative towards the successful acquisition of new knowledge [15].

Principle 2: "learning facts and learning to do something are two different processes." This is particularly true in an environment where learners are exposed to a project management environment. Many authors have acknowledged this challenge and adopted innovative approaches in an attempt to over-come the challenge, for example a game-action learning approach (GAL) whereby knowledge can be applied in different scenarios [16] and the use of simulations and real world projects [17].

Principle 3: "Some things are learned specific to the domain or context in which they are learned." From a practical project management application point of view this contributes to the challenge of exposing learners to real-world projects that can facilitate realistic real world environments. Adding to the challenge is the need for learners to apply their knowledge across disciplines, in this instance computer science, information systems or management sciences. This adds to the project complexity of managing projects in a diverse environment [18].

Principle 4: "Individuals are more likely to learn from one another than when they learn alone." Gil and Mataveli [19] support this principle and refer to peer learning as project learning.

Principle 5: "Meaningful learning is facilitated by articulating explanations by one's self, peers or teachers." This principle is related to principle 4 and supports the concept of project learning by Gil and Mataveli [19].

4 Case Study

One of the aspects that is included in the second year ISD curriculum, is project management. Theoretical concepts related to project management such as feasibility analysis, triple constraint of project management are covered in class and students are taught how to use a project management tool such as Microsoft Project. Project management is a necessary skill that students need to learn in preparation for the capstone project in their third year. In order to prepare them for the possible issues they might encounter in their capstone project, a tutorial was designed to make them aware of the issues they might have to "manage". The tutorial was designed as such:

- Obtain a list of issues that capstone project students have encountered when completing their project. The issues ranged from academic commitments such as tests (for core modules and electives) that need to be written on a specific date, an assignment submission, group meeting to work on an assignment, social commitments (meeting up with the boyfriend/girlfriend) and family commitments such as attending your grandfather's birthday. This was obtained from the assistant lecturers (fourth year students that completed the capstone project the previous year).
- Print these issues on different colours of paper and put it in the "box of challenges", as illustrated in Fig. 1.

Fig. 1. Box of challenges.

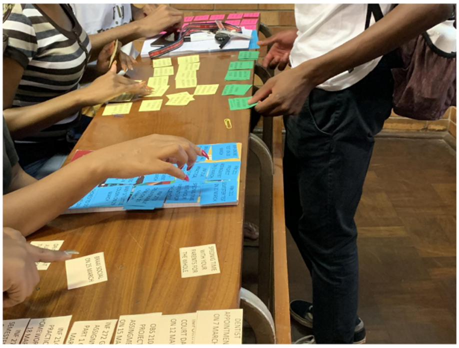


Fig. 2. Completing the activity.

– Students were encouraged to attend the class in their project groups as far as possible (so that team members can learn about one another - as an effort to improve team dynamics).
– Each group of five members (the majority of groups were the team members) each got a set of 20 commitments/challenges (a mixture of academic, personal and social) which they had to individually prioritise, as illustrated in Fig. 2.
– After individual prioritisation, the group needs to compare their priorities and work out a schedule that will work for the group.

5 Methodology

In order to assess the success of the activity the authors followed an interpretive approach, acknowledging the subjective nature of the data that was collected from the students [20]. The interpretive approach further suited the nature of Active Learning [3] where students interpreted the situation individually, discussed it as a collective group before reflecting on the process. Following this, the groups had to answer the following questions:

– What were the main areas of conflict?
– How did you as a group come to a compromise?
– What did you learn from this exercise?

Thematic Content Analysis [21] was used to analyse the tutorial answers to the above questions. One researcher read through the tutorial responses (45 in total) and identified the themes that emerged for each question. These themes are presented in Figs. 3, 4 and 5 below.

For the second part of the study, students were asked to complete a survey based on the principles of active learning [3] to measure the extent to which the exercise was successful. The questions and the results are presented in Sect. 6.4.

Inferential statistical analysis was conducted on the Principles of Active Learning survey. The survey was answered by 187 students, with 185 complete responses. Results of the analysis are presented in Sect. 6.

6 Analysis and Discussion

The aim of the paper is to report on the level of success of an active learning tutorial to teach second year students about how to manage "real-life" constraints when setting up a common project schedule. This tutorial was set up with the help of fourth year ISD students who were able to suggest common challenges they struggled with whilst working on their third year capstone project. As the objective of the second year ISD module is to prepare the students for their capstone project (which involves extensive and intense group work), the tutorial was designed to assist with that.

6.1 Main Challenges in Creating a Joint Schedule

Thematic analysis has revealed the challenges students experienced. The major challenges are illustrated in Fig. 3 below. At the core of project management teaching, is the concept of the project management triple constraints: balancing performance, cost and time [9]. The simulation exercise attempted to address the performance and time aspect. In a simulated university environment the main trade-offs existed between balancing social and academic commitments, and the extent to which that can be balanced will affect the performance of the group. The major challenges identified by the students were:

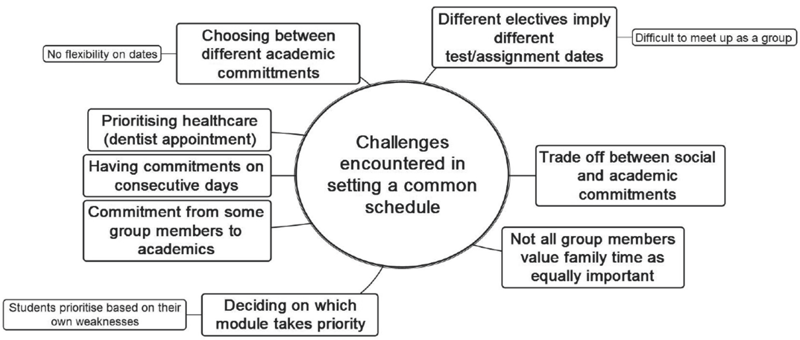


Fig. 3. Challenges encountered in setting a common schedule

Balancing Different Academic Commitments. The students struggled to compromise on different academic commitments. A group often consists of members having different electives, and that implies that each member will have different assessment dates for their respective electives. As the dates are usually very close to one another, it is difficult for the students to find common ground in choosing between the "semester test" of one module and the need to work as a group on a "group assignment" for another module.

Furthermore, students indicated that depending on what is their "problem" module, they might require to spend more time in preparing for a test, which might not be in balance with the rest of the group's capabilities.

Balancing Social and Academic Commitments. The second major challenge highlighted by students is the ability to attend to social events (one of the examples given as a challenge card was "going out with your BFF") and one response was that you could not "sideline your BFF". Other approaches to this dilemma were that it was obvious that academic work always take precedence over social commitments. Although not on a social level, students in the study by Geithner and Menzel [22] mentioned that they learned how to structure tasks and subsequently prioritize commitments.

Students Value Family Time Differently. It was evident from the responses that the fact that some students in the group valued spending time with family as more important than other group members, and this created a challenge when creating a common schedule. One of the examples given in the challenge card was "attending your elderly grandfather's birthday".

The Importance of Healthcare. One of the challenge cards indicated that you have to attend a dentist appointment. A number of the 45 groups referenced the dentist appointment challenge, indicating that healthcare should take precedence over academic/social commitments, whilst other responses focused on the discourse surrounding the nature of the dentist appointment and whether it was an emergency or not. Although the scenarios presented through the challenge cards were not exhaustive, it attempted to simulate the "real-life" aspects that will impact a group's ability to perform within a time constrained environment (something which they will experience in their capstone project).

6.2 Compromises Made in Creating a Joint Schedule

The next part of the tutorial was to learn how the students compromised given the challenges experienced, in order to create a common schedule between the group members. Thematic analysis revealed a number of ways students compromised, as displayed in Fig. 4. The most important compromises will be discussed in turn:

Priority Depends on Context/Seriousness of Condition. This aspect is related to the fact that at least one of the students in the group received a card that indicated that s/he had to attend a dental appointment. Students had a discourse on the nature of the appointment, whether it was an emergency or not, can you move it? Although the example of a "dentist appointment" is simple, the discussions surrounding the challenge that this appointment provides to the group and their ability to perform well, are important in negotiating the group's expectation from the specific group member depending on how well they understand the nature and context of his/her challenge. Group members need to decide on the negotiation process that will be followed for the duration of the project. The choice will be between hard, soft or principled with the objective being to minimize loss for the least negative effect [23].

Make Sacrifices for the Future. One group (from 45) indicated that it helps to be "forward thinking" in realizing that the sacrifices they have to make now, will be worth it for any future rewards.

Activities with No Dedicated Time Slot Need to Be Managed Personally. A number of commitments such as "going to the gym", "going on a date with your boyfriend/girlfriend", and "babysitting your cousin's child" were included in the box of challenges. The majority of groups have demoted these commitments as less important than academic commitments. However, a discussion surrounding these types of commitments highlighted the extent to which academic and social life need to be balanced. One group suggested that "extended family" should help with "babysitting duties" which emphasized the support students might need in order to balance their commitments. In Pretorius and Hattingh [24] it was argued that students do not work in isolation and that their learning environment is both impacted by their personal and academic environment.

Develop a Hierarchy of Work. One group suggested that as a group, they need to develop a hierarchy of work. This particular group prioritized group work at the top, followed by individual assignments, then studying for any tests and then social commitments. Group members need to establish ground rules as part of their common schedule negotiation [23].

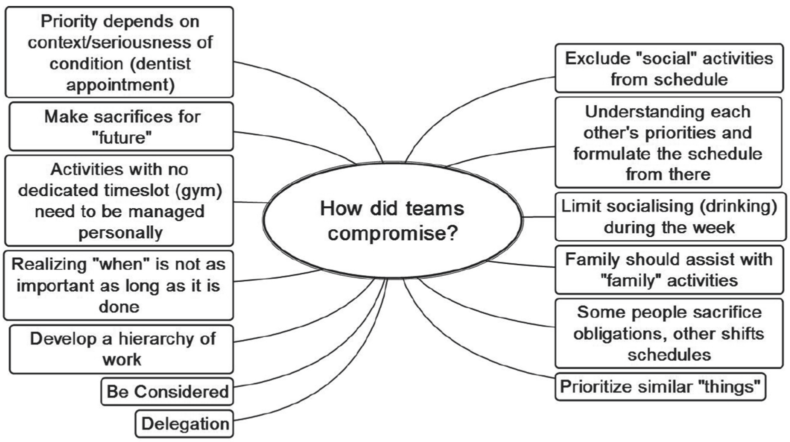


Fig. 4. Compromises made by students

Sacrifice vs Re-schedule. Groups have realized that in order to compromise, they had to either sacrifice an activity (usually social activities), or that they can/should re-schedule activities as to not to impact on the performance of the team. This is an interesting distinction made by the students and might be due to the fact that students developed a better understanding of the needs of fellow team members, similar to a study by Geithner and Menzel [22].

Prioritise Similar Things. The final strategy that was put forward by a number of groups is to prioritise similar things. This is similar to point four where a hierarchy of work needs to be developed. By prioritizing similar things, it is less likely that group members might feel that they compromise the whole time whilst other members do not have the same level of commitment. Again, this is in line with the study by Geithner and Menzel [22] where team members developed better understanding of fellow students' needs.

6.3 Lessons Learnt by the Students from the Tutorial

Although the true benefit of this active learning exercise will only be evident when the students complete their capstone project, some of the lessons learnt is transferable to other aspects of life. This will be discussed further in Sect. 6.4, principle 3. Thematic analysis revealed that the students have learnt a number of lessons by completing this tutorial. Figure 5 outlines the lessons learnt, and the most important lessons learnt will be discussed in turn:

The Need to Plan Ahead. The Core of Project Planning is to Plan Ahead. However, students often struggle with that. As the "box of commitments" included a variety of commitments, both academic and social but also a number of academic commitments within a module and across other modules (electives), students were able to see that if they want to find time to work on a group project, they will have to plan ahead.

Completing Activities on Time (Time Management). Another key aspect of project management which was recognised by the students. Students are also exposed to a project management tool, such as Asana which will allow them to see the dependencies between activities if deadlines are missed.

Consequences of Making Hard Choices. The activity allowed students to consider sacrifices and compromises they had to make in order to reach their academic goals. The activity questioned their levels of commitment in meeting those goals. This "lesson" is hopefully key in them realizing that hard work now will pay off later. The use of the simulation exercise enabled the integration of the knowledge the student gained during the theoretical lecture with the practice of constraint decision-making [8].

Individuals' Value Systems Differ and Prioritise Differently. The subjectivity of the human has transpired through this lesson learnt. Group members got to know themselves in relation to their group members and learnt that as each of them are individuals they have different priorities, different value systems, different ways in which they handle different levels of pressure and that they have different levels of commitment in reaching their goals.

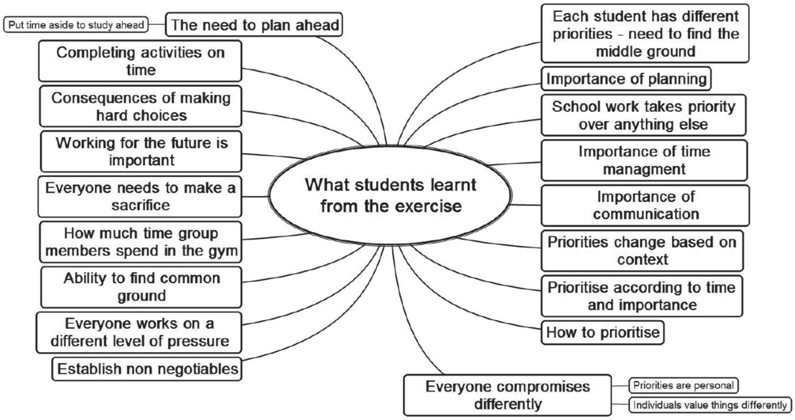


Fig. 5. Lessons learnt by students from tutorial

How Much Time Individuals Spend in the Gym. Going to the gym is a metaphor for all those "other important" commitments that group members might have.

Importance of Communication. Communication is a significant knowledge area according to PMBOK [9]. This exercise encourages communication between group members about their individual priorities. It was evident that with better communication group members might understand the context of some challenges experienced by group members and as a consequence might compromise on their contribution to the combined schedule.

6.4 Extent to Which Active Learning Principles Were Met

A survey of 13 questions was released to the students. 188 students completed the survey with 186 complete answers. The results indicated that students agree that the principles of active learning have been met: 59.8% strongly agreed/agreed that principle 1 has been met, 45.2% strongly agreed/agreed that principle 2 has been met, 71.8% strongly agreed/agreed that principle 3 has been met, 58.9% agreed that principle 4 has been met and 71.5% agreed that principle 5 has been met.

It is not surprising that principle 2's scores were lower, as students still have to practically apply these skills they have learnt through the tutorial. Table 2 provides the items against which the extent to which the principles of active learning were met. Questions 2, 3 and 11 were omitted in the correlation analysis. Question 5 was reversed scored. See Table 3 for the results of the reliability test.

Cronbach alpha values are in indication of construct reliability. The instrument questions were formulated based on the five theoretical principles of active learner and no further consideration was given to this result. One explanation for the low values might be due to the low number of questions asked and the heterogeneity of the questions due to the diversity of the five principles.

Table 2. Questionnaire for principles of active learning

Principle	Question
1	1. The class exercise assisted me in understanding the theoretical concept of managing project constraints better
1	2. Prior to the class, I had no prior knowledge of how to manage project constraints
1	3. Prior to the class, I had some knowledge of how to manage constraints in projects but it was not 100% correct
1	4. The class exercise enabled me to understand how to manage project constraints better
2	5. At the end of the theoretical explanation of project constraints, I was still a bit unsure of how to practically apply it*
2	6. At the end of the practical exercise I had a clear understanding of how to manage project constraints.
3	7. I will be able to practically apply this approach to handling constraints in my personal life
3	8. I will be able to practically apply this approach to my current, study-related IT projects
4	9. I learned a lot from my fellow students (team members)
4	10. I learned more from my fellow students than what I would have learned when studying alone
5	11. The explanation of concepts by my lecturer assisted in my understanding of project constraints
5	12. The explanation of concepts by my fellow students assisted in my understanding of project constraints
5	13. Explaining to my fellow students, the reasoning for the prioritization of my allocated constraints, assisted in my understanding of project constraints

Table 3. Reliability analysis

Principle	Chronbach alpha
Principle 1	0.63
Principle 2	0.497
Principle 3	0.659
Principle 4	0.620
Principle 5	0.554

Table 4 presents the correlation results between the principles. Due to space constraint only the most prominent correlation will be discussed. The strongest correlation existed between principles 4 and 5. Both of these principles are related to "how" learning takes place: in a group (principle 4) and through the peers

Table 4. Correlation analysis results between active learning principles

	Principle 1	Principle 2	Principle 3	Principle 4	Principle 5
Principle 1	1	0.313**	0.364**	0.285**	0.372**
Principle 2		1	0.173*		0.281**
Principle 3			1	0.146	0.255**
Principle 4				1	0.503**
Principle 5					1

(principle 5). From the previous sections it was evident that group members had to articulate how they prioritise, which allowed group members to understand their context and situations. Through this, they learnt about challenges working in a group, which are not unique to undergraduate students that are forced to work in groups [25].

Although there were significant correlations between the other principles, none of them were very strong. This might be noteworthy and can subsequently be investigated in future studies.

6.5 Reflection from the Lecturers

From a lecturing perspective, the active learning tutorial was regarded as a success. As supported by the findings discuss, the lecturers observed active class discussion, were called in to clarify given "commitments" which is important to understand the context of the potential challenge which will enable the group members to prioritise accordingly. Feedback from the students (gathered at the time of the active learning principles survey) were overwhelmingly positive. Therefore, the positive aspects from the active learning tutorial can be summarized as (1) an enjoyable activity for the students; (2) fostering group dynamics for groups that managed to attend in their project groups; (3) students came to the realization of managing available resource (them) versus time, and the potential implication for group performance.

Following the completion of the activity, the lecturers have decided that when this activity is implemented again next year the following changes will be made: (1) encourage well in advance group members to attend in their project group as the activity can be expanded to induce reflection on team dynamics; (2) Don't assign dates to all social/non-academic commitments. It was observed that in certain instances assigning a date constrains the students' planning capability.

7 Conclusion

This study determined that the adoption of active learning simulation involving a decision-making activity and case study simulation was effective in teaching project management to second year students. This study used Michael's five principles of active learning to confirm that an active learning approach to second year project management is valuable. The exercise used for the case study was successful in highlighting the different challenges students experience with regards to balancing different academic commitment - especially if their electives differ, balancing academic work and social commitments, and the importance of family and healthcare. The exercise was further successful in identifying strategies students can employ to compromise on developing a joint schedule for the members of the team. Furthermore, an important part of the exercise was that the students could reflect about the exercise and discuss what they learnt from it. The lecturers found this exercise valuable and insightful and keeping the recommendations of the previous section in mind, will repeat this exercise in the future and compare the results with the findings of this cohort.

References

1. Shango, D.: Why the skills gap remains wider in Africa (2019). https://www.weforum.org/agenda/2019/09/why-the-skills-gap-remains-wider-in-africa/. Accessed 25 Mar 2021
2. Marino, M.D.: Self Directed Learning with Hidden Action: A Quadratic Model with an Implication for the Skills Gap. Social Science Research Network, Rochester, NY, SSRN Scholarly Paper ID 3373752, April 2019. https://papers.ssrn.com/abstract=3373752. Accessed 08 Mar 2020
3. Michael, J.: Where's the evidence that active learning works? Adv. Physiol. Educ. **30**(4), 159–167 (2006). https://doi.org/10.1152/advan.00053.2006
4. Martz, B., Hughes, J., Braun, F.: Creativity and problem-solving: closing the skills gap. J. Comput. Inf. Syst. **57**, 39–48 (2016). Accessed 08 Mar 2020
5. Olusegun, B.S.: Constructivism Learning Theory: A Paradigm for Teaching and Learning, pp. 66–70
6. Nordkvelle, Y., Stalheim, O.R., Fossland, T., de Lange, T., Wittek, A.L., Nerland, M.B.: Simulating: bridging the gap between practice and theory in higher professional education. In: Trimmer, K., Newman, T., Padró, F.F. (eds.) Ensuring Quality in Professional Education Volume I, pp. 53–72. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-01096-6_3
7. Walters, B., Potetz, J., Fedesco, H.N.: Simulations in the classroom: an innovative active learning experience. Clin. Simul. Nurs. **13**(12), 609–615 (2017). https://doi.org/10.1016/j.ecns.2017.07.009
8. Brame, C.: Active Learning. Vanderbilt University, 20 June 2016. https://cft.vanderbilt.edu/guides-sub-pages/active-learning/. Accessed 08 Mar 2020
9. Project Management Institute: PMBOK® Guide - Sixth Edition (2017), 6th edn, Newton Square, PA, United States (2017)
10. Gido, J., Clements, J., Baker, R.: Successful Project Management. Cengage Learning (2017)
11. Rugenyi, F.: Assessment of the triple constraints in projects in Nairobi: the project managers' perspective. Int. J. Acad. Res. Bus. Soc. Sci. **5**(11), 1–16 (2015)
12. Naor, M., Bernardes, E., Coman, A.: Theory of constraints: is it a theory and a good one? Int. J. Prod. Res. **51**(2), 1–13 (2012)
13. Ruiz-Primo, M.A., Briggs, D., Iverson, H., Talbot, R., Shepard, L.A.: Impact of undergraduate science course innovations on learning. Science **331**(6022), 1269–1270 (2011). https://doi.org/10.1126/science.1198976

14. Juli, T.: Power and Illusion of Self-Organizing Teams, North America, Vancouver, British Columbia, Canada (2012). https://www.pmi.org/learning/library/power-illusion-self-organizing-teams-5970. Accessed 08 Mar 2020
15. Genwalal, C., Bekker, M.C.: Project manager attributes influencing project success in the South African construction industry (2015). https://repository.up.ac.za/handle/2263/51988. Accessed 08 Mar 2020
16. Law, K.M.: Teaching project management using project-action learning (PAL) games: a case involving engineering management students in Hong Kong. Int. J. Eng. Bus. Manag. **11**, 1847979019828570 (2019). https://doi.org/10.1177/1847979019828570
17. Collingbourne, L., Seah, W.K.G.: Teaching project management using a real-world group project. In: 2015 IEEE Frontiers in Education Conference (FIE), Camino Real El Paso, El Paso, TX, USA, pp. 1–8, October 2015. https://doi.org/10.1109/FIE.2015.7344301
18. Cristóbal, J.R., Carral, L., Diaz, E., Fraguela, J.A., Iglesias, G.: Complexity and project management: a general overview. Complexity **2018**. https://doi.org/10.1155/2018/4891286
19. Gil, A.J., Mataveli, M.: Learning opportunities for group learning: an empirical assessment from the learning organization perspective. J. Workplace Learn. **29**(1), 65–78 (2017). https://doi.org/10.1108/JWL-02-2016-0009
20. Avison, D., Fitzgerald, G.: Methodologies for developing information systems: a historical perspective. In: The Past and Future of Information Systems: 1976–2006 and Beyond, vol. 214, pp. 27–38 (2006). https://doi.org/10.1007/978-0-387-34732-5
21. Braun, V., Clarke, V.: Using thematic analysis in psychology. Qual. Res. Psychol. **3**(2), 77–101 (2006). https://doi.org/10.1191/1478088706qp063oa
22. Geithner, S., Menzel, D.: Effectiveness of learning through experience and reflection in a project management simulation. Simul. Gaming **47**(2), 228–256 (2016). https://doi.org/10.1177/1046878115624312
23. Enguland, R.L.: Negotiating for success: are you prepared? Presented at the PMI® Global Congress 2010, EMEA, Milan, Italy (2010). https://www.pmi.org/learning/library/negotiating-project-outcomes-develop-skills-6781. Accessed 08 Mar 2020
24. Pretorius, H.W., Hattingh, M.J.: Factors influencing poor performance in systems analysis and design: student reflections. In: Liebenberg, J., Gruner, S. (eds.) SACLA 2017. CCIS, vol. 730, pp. 251–264. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-69670-6_18
25. Isaac, S., Tormey, R.: Undergraduate group projects: challenges and learning experiences. QScience Proc. **2015**, 19 (2015). https://doi.org/10.5339/qproc.2015.wcee2014.19

Factors Determining the Intention of Using Online Learning Videos: A Study from Uzbekistan

Anchal Garg[1(✉)] and Jean-Paul Van Belle[2(✉)]

[1] School of Creative Technologies, University of Bolton, Bolton BL3 5AB, UK
a.garg@bolton.ac.uk

[2] Centre for IT and National Development in Africa, University of Cape Town, Cape Town 7701, South Africa
jean-paul.vanbelle@uct.ac.za

Abstract. In recent years, ICTs have been used increasingly in teaching and learning. With the onset of e-learning, m-learning, MOOCs, SPOOCs, ICT is becoming an imperative part of the education domain. Organizations such as Khan Academy, Coursera, edX, and Udacity have produced and made thousands of educational videos available to the masses globally. The number of users using these online platforms have been increasing steeply. Although it is known that the online learning videos are popular among the students, little is known about the drivers of the user's intentions and use of these videos. The paper intends to answer this gap. The purpose of this study is to identify the factors that influence the students' behavioral intentions to adopt online learning videos in an economically developing country like Uzbekistan. Not only does our study inform educators about what motivates student to use educational videos, it is also one of the very rare studies to explore ICT use in Uzbekistan.

Keywords: Online learning videos · Educational videos · E-learning · Unified Technology and User Technology Acceptance Model (UTAUT) · ICT use in Uzbekistan

1 Introduction

Information and Communication Technologies (ICTs) have made inroads into every sector and become indispensable to their day-to-day operations. Subsequently, there is a growing interest on the use of ICTs in the education sector due to the many advantages that it offers. ICTs have been shown to improve access, equity, quality of teaching and learning as well as helping to achieve overall educational excellence [1]. With the onset of e-learning, m-learning, MOOCs, SPOOCs, ICTs are becoming an imperative part of the education domain. Organizations such as Khan Academy, Coursera, edX, and Udacity have produced and made thousands of educational videos available to the masses globally [2].

G. Wells et al. (Eds.): SACLA 2020, CCIS 1518, pp. 145–155, 2021.
https://doi.org/10.1007/978-3-030-92858-2_10

The number of users using these online plat-forms have been increasing steeply. Popular MOOC platforms such as edX and Coursera are reported to have nearly four and half million enrolled students [3]. Khan Academy's YouTube channel that started in 2006, now has 7517 videos as opposed to 1499 in 2014 [4,5]. In 2020, the channel has 5.3M subscribers and over 1.7 billion views [6]. Furthermore, the studies reveal that the students mainly watch videos and skip online discussions and other interactive media [3,5,7].

Although it is known that the online learning videos are popular among the students, little is known about the drivers or factors which drives the (intentions to) use of those videos. The paper intends to answer this gap. Only one such study has been conducted in Germany [8]. Also, it is not known how prevalent the adoption of online learning videos is among students in under-researched economically developing country like Uzbekistan, and what aspects contribute towards its adoption. Moreover, it might be useful to explore the issues in using online videos in Uzbekistan and those might be different from that of other developed or in developing countries. Furthermore, the studies on MOOCs reveal that there is low student engagement and lower completion rates with high dropouts [9–11]. Although, online learning videos are very popular among the students, it does not mean that those are successful. Hence, it is essential to know the behavioral intentions on using those online learning videos [12].

It is vital to determine the factors that encourage the students to watch online learning videos so that such factors can be taken into consideration while creating online courses. Such a study will also help us identify the issues with the online learning platforms and accordingly take necessary action. The purpose of this study is to identify the factors that influence the students' behavioral intentions to adopt online learning videos in a very under-researched economically developing country like Uzbekistan.

This research questions of this research are (a) What are the factors that affect the use of online learning videos by students in an economically developing country? (b) What are the reasons of using online learning videos? (c) What are the popular plat-forms used by students for watching online learning videos? (d) What is the average length of videos that the students' watch? (e) What are the issues with the online learning videos?

The paper is structured as follows: Part 1 included introduction this is followed by Methodology in Sect. 2, Sect. 3 consists of Result and Analysis, Sect. 4 covers the Discussions and Conclusion.

2 Methodology

This study uses Unified Theory of Acceptance and Use of Technology (UTAUT) model proposed by Venkatesh et al. (2003) [13]. Since its introduction it has been employed in number of studies to identify factors affecting technology acceptance by users. For instance, UTAUT has been a practical tool to investigate technology acceptance in context to various technologies such as webinar technology [14], academic e-learning technologies [15], e-commerce [16], Internet banking

[17], open access [18], e-government [19], social networking [20], and m-learning [21]. The reason for choosing UTAUT model over Technology Acceptance Model (TAM) is that we were interested also interested in exploring attributes such as gender, educational background, computer experience and the students' voluntariness to use the system along with other behavioral factors. Moreover, UTAUT model is better at predicting user behaviour compared to TAM model [13].

The UTAUT model enables us to assess dependent variables (DVs), i.e. behavioral intention (BI) and user behavior, by estimating the influence of four core independent variables (IVs) namely performance expectancy (PE), effort expectancy (EE), social influence (SI), Perceived Usefulness (PU) and facilitating conditions (FC). Besides, UTAUT considers the influence of four moderators (M): age, gender, prior experience with technologies, and voluntariness of use. PE describes how much advantage an individual hope to obtain; EE explains how much the same individual expects to reduce costs. FC and SI refer to contextual concepts relating to the institutional and social surroundings of an individual. While, FC are the conditions set by the institutional surrounding, SI is drawn from people within the direct social environment of an individual and their attitudes towards the technology in question.

The current study used college and university students based in Uzbekistan as the sampling frame. Very little, if any, research has been internationally published about individual's ICT use in Uzbekistan. For this study, questionnaire survey was used to conduct this study. The first part of the questionnaire consisted of questions on demographics, frequency of using online learning videos, average length of video watched, and one on commonly used platform. The second part included questions on each determinant PE, EE, SI, FC, PU, Satisfaction and BI. Each question was measured using a 5-point Likert scale (1 = strongly agree, 2 = agree, 3 = neither agree nor disagree, 4 = disagree, 5 = strongly disagree). The questionnaire was created in Google forms and the link was distributed to around 500 students through email, LinkedIn and Telegram.

For the survey, only students who were well-versed in English were included. Total of 254 responses were obtained in three weeks (response rate = 50.8%). Out of total responses, total of 202 responses were taken for analysis. All those who answered 'Yes' to the use of online learning videos were included and all with 'No' were excluded as they did not complete the other part of the questionnaire. The responses were then analyzed to find the effect of IV on DV using statistical methods such exploratory factor analysis and multiple regression using Statistica software. Descriptive analysis was performed to provide insights into some of the queries posed during the survey.

3 Result Analysis

The analysis is discussed under several headings: first a descriptive analysis of the sample of respondents as well as their use of online learning videos is given. Then a qualitative analysis of some of the interesting questions is given. We conclude with inferential statistics based on the UTAUT model's hypothesized relationships.

3.1 Descriptive Analysis of the Sample

Out of total 254 responses, 202 respondents (79.5%) used online learning videos, while 56 respondents (22%) did not use those at all. Of the total respondents 173 (68.1%) were males, 78 (30.7%) were females and 3 (1.2%) of them did not prefer to respond to this. 91.4% of the respondents were in the age group of 17–23. As can be assumed from the age, 92.2% of the students were either pursuing foundation program or a bachelor's program. 54.7% of students were pursuing a degree in IT, 17% in Economics, 10.5% in Travel Tourism, 7.7% in Management, and others from law, psychology, politics and language. Several other questions were posed and the responses to those are given below.

Question 1: What Is Your Experience in Using Computers?
Response: 98% of the respondents had an experience in using computers as apparent from Fig. 1.

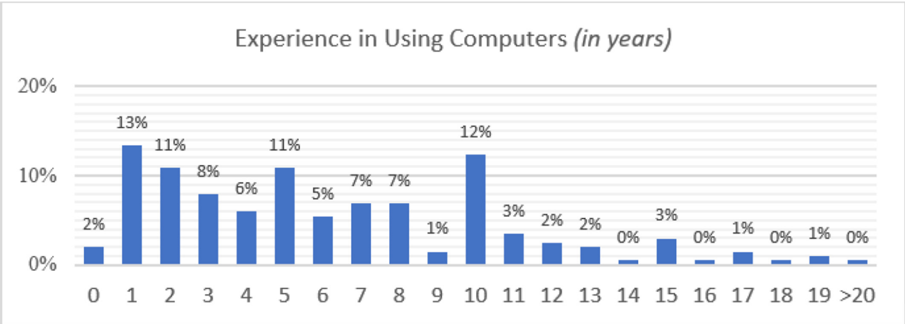


Fig. 1. Experience in using computers (in %).

Question 2: Have You Used Online Learning Videos?
Response: Around 70% of the respondents used online learning videos, 20% did not watch and about 10% were not sure about it.

Question 3: How Frequently Do You Use Online Educational Videos?
Response: The results indicate that more than half of the students generally regularly watch online learning videos (Fig. 2)

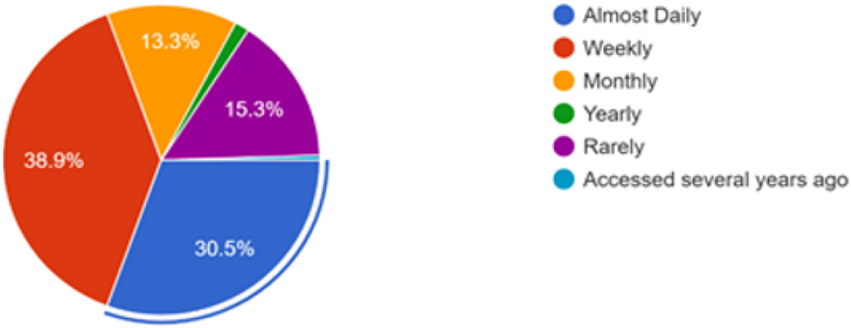


Fig. 2. Frequency of using online learning videos.

3.2 Their Use of Online Learning Videos

Question 4: What Is the Average Length of the Videos You Watch?
Response: Fig. 3 shows the average length of the videos watched by the students. As seen from Fig. 3, respondents generally watch it between 5–30 min.

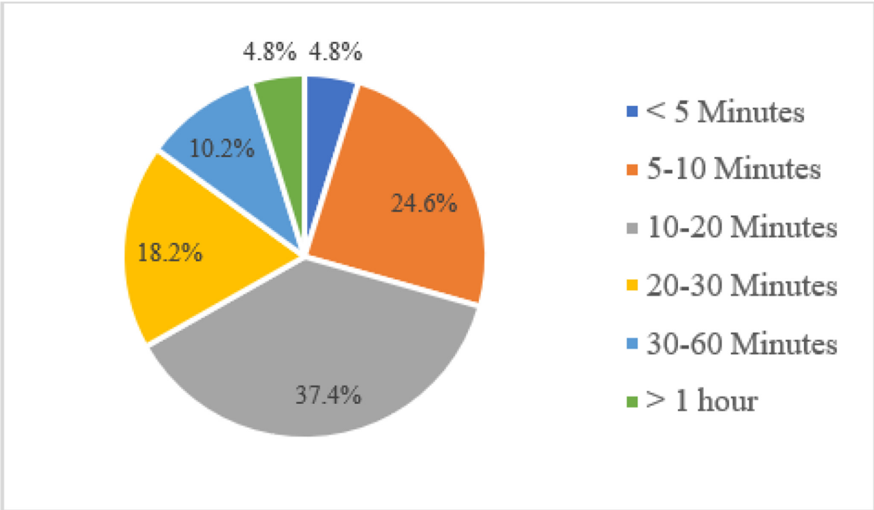


Fig. 3. Average length of videos watched.

Question 5: Which Platform Do You Use to Access Online Educational and Learning Videos?
Response: As apparent from Fig. 4, majority of the respondents watch videos on YouTube.

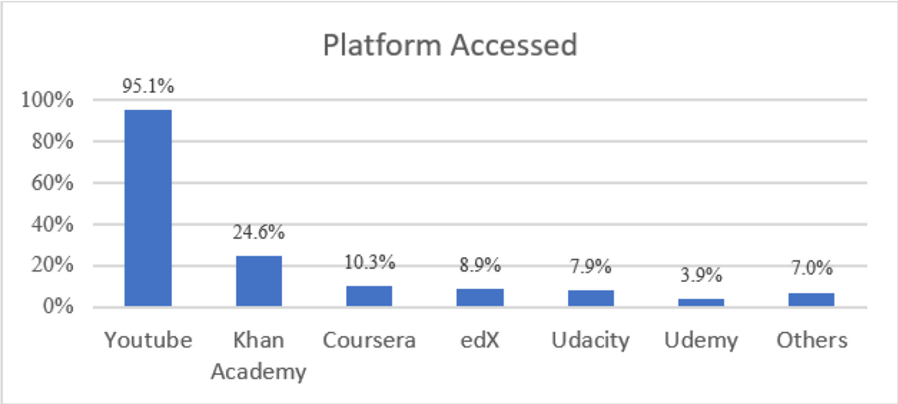


Fig. 4. Average length of videos watched.

Question 6: Please List the Three Main Topics or Subjects for Which You Watch Educational/Learning Videos

Response: Commonly listed are the course subjects such as mathematics, economics, business statistics, marketing, finance, accounting, chemistry, physics, psychology, tourism, law, geography, probability and statistics, while others watch those for their career enhancement English, IELTS, Programming subjects such as JAVA, Python, C, HTML, C++, Android development, Game development, latest technologies such as artificial intelligence, IoT, Robotics, data science, blockchain, cybersecurity, Machine learning, farming technology, social media marketing, NodeJS, video editing, cinematography, photoshop, scriptwriting, or for helping in their hobbies such as playing guitar and piano, cooking, dancing, foreign language, beauty and yoga.

3.3 Drivers of and Obstacles to Use of Online Learning Videos

Question 7: In Your Opinion, What Are the Two or Three Biggest Benefits or Ad-Vantages of Using Online Learning Videos (When Compared to Using Other Types of Learning Resources)?

Responses: The respondents feel that watching the videos online is beneficial to them. Watching videos online is comfortable, easy to understand and learn, useful, faster, can be watched repeatedly, offers anywhere anytime access, affordable, it is easy to find information, offers flexibility, has good quality, portable, use animated videos, variety of options and resources, visualization, time efficiency, listening is better than reading, more examples, effective and efficient lectures, easy to find, easy to use, fits your time, improves knowledge, self-paced, easy to search, helps in completing homework, quick and clear explanation, improves retention, seriousness, cannot be downloaded, affects health and causes laziness, not all videos are free, and many display advertisements. Overall, it appears that Internet connectivity and language, pronunciation, not getting (the right?) answers to questions, and too many videos (too much choice?) are the major issues.

3.4 Validity and Reliability Analysis of UTAUT Constructs

We now turn to the UTAUT model. First it is necessary to establish the reliability and validity of the hypothesized constructs. Test items were grouped using exploratory factor analysis. When this is completed, multiple regression will be done to assess the hypothesized relationship between IVs and DVs.

Exploratory factor analysis with a principal component analysis (PCA) was conducted on the 23 items with orthogonal rotation (varimax). The Kaiser-Meyer-Olkin (KMO) measure verified the sampling adequacy for the analysis, KMO = .941. Bartlett's test of sphericity - (234) = 3137.074, p ¡ .001, indicated that correlations between items were sufficiently large for PCA. Table 1 shows the factor loadings after rotation. Results from the rotated component matrix (Varimax rotation) indicate an item, The OLVs are compatible with the other systems I use (FC4), had to be omitted due to weak cross-loading. Six factors identified and confirmed by way of PCA are: Effort Expectancy (EE), Performance Expectancy (PE) and Satisfaction, Social Influence (SI), Facilitating Condition (FC), Perceived Usefulness (PU), as well as Voluntariness of Use (VU). The variables that were grouped with PE are OLVs helps me understand subject easily (PE1), OLVs enhances my learning effectiveness (PE2), OLVs accomplish my learning tasks more quickly (PE3). The variables grouped with EE are Interaction with OLVs is clear and understandable (EE1), Easy for me to be skillful using OLVs (EE2), I find OLVs easy to use (EE3), Operating OLVs is easy for me (EE4), Easy to learn what I need using OLVs (EE5). While those under SI include People who influence my behavior think I should use OLVs (SI1), People who are important to me think that I should use OLVs (SI2), People who are important to me have been helpful in using OLVs (SI3). FC included I have the re-sources necessary to use OLVs (FC1), I have the knowledge necessary to use the OLVs (FC2), my university supported the use of OLVs (FC3). PU were grouped into increase my chances of better academic performance (PU1), Improves knowledge retention (PU2), OLVs are useful (PU3), OLVs not compulsory for my learning (VU1), OLVs not compulsory by university (VU2), and Satisfaction.

This was followed by multiple regression. The SPSS Output in Table 2 shows the model statistics. The model explains 58% of the variance in BI1, 63% variance in BI2 and 60% variance in BI3 can be explained by all the factors. This is a very satisfactory result since we are explaining well more than half of the variance. The ANOVA results (Table 2 and Table 3) confirm that all the models are statistically significantly.

Table 1. Results of PCA

Rotated component matrix[a]						
	Component					
	1	2	3	4	5	6
EE5 easy to learn what I need using OLVs	0.691					
EE4 operating OLVs is easy for me	0.689					
EE3 I find OLVs easy to use	0.674					
EE2 easy for me to be skilful using OLVs	0.663					
EE1 interaction with OLVs is clear and understandable	0.641					
SI1 People who influence my behavior think I should use OLVs		0.846				
SI2 People who are important to me think that I should use OLVs		0.833				
SI3 People who are important to me were helpful in using OLVs		0.725				
PE1 OLVs helps me understand subject easily			0.751			
Satisfaction			0.672			
PE2 OLVs enhances my learning effectiveness			0.658			
PE3 OLVs accomplish my learning tasks more quickly	0.511		0.558			
FC3 my university supported the use of OLVs				0.731		
FC1 I have the resources necessary to use OLVs				0.692		
FC2 I have the knowledge necessary to use the OLVs				0.677		
FC4 The OLVs are compatible with the other systems I use						
PU1 increase my chances of better academic performance					0.625	
PU3 OLVs are useful					0.574	
PU2 improves knowledge retention					0.573	
VU1 OLVs not compulsory for my learning						0.873
VU2 OLVs not compulsory by university						0.836
Eigen values	11.12	1.686	1.195	0.887	0.712	0.658
% of the total variance	52.86	8.03	5.69	4.235	3.39	3.13
% Total variance	77.434					
Extraction method: principal component analysis						
Rotation method: varimax with Kaiser normalization						
a. Rotation converged in 8 iterations						

Table 2. (A) Multiple regressions for BI1 to BI3 (summarized)

Dependent variable	Model	R^2	Adjusted R^2 e	Std. error of the estimate
BI1	1	0.58	0.52	0.512
BI2	1	0.636	0.584	0.483
BI3	1	0.598	0.54	0.505

Table 3. (B) Multiple regressions for BI1 to BI3 (summarized)

Model BI1		Mean square	F	Sig.
BI1	Regression	2.522	9.615	$.000^b$
	Residual	0.262		
Model BI2		**Mean square**	**F**	**Sig.**
BI2	Regression	2.835	12.165	$.000^b$
	Residual	0.233		
Model BI3		**Mean square**	**F**	**Sig.**
BI3	Regression	2.633	10.337	$.000^b$
	Residual	0.255		

4 Discussions and Conclusion

The objective of this study was to find the factors influencing the use of online learning videos by the students of Uzbekistan. To accomplish that, the UTAUT model was used. The results revealed that majority of the respondents watch online videos for learning, and that YouTube is the most popular platform. Students generally watch videos that are 10–20 min long (37%) or 5–10 min (25%). Moreover, anywhere anytime access and possibility of watching the video any number of times are the common benefits of using those videos. The internet connection is an issue is several regions of Uzbekistan and therefore poor internet connectivity was the most common problem listed by the respondents. Also, language was another barrier listed by the respondents. Since, Russian and Uzbek is the language spoken in Uzbekistan, not all people are able to take the advantage of the rich universe of online learning videos, since most of them are in English. The huge amount of learning videos also confused respondents. Respondents view videos not only for help in their core subjects but also as an aid for career enhancement and for pursuing their hobbies and other interests.

The results of multiple regression analysis indicate that all the factors and other variables such as age, gender, experience in using computers and voluntariness of use have dependency on all the behavioral intentions. This study focused only on English speaking students. Future research could include people using online learning videos using languages other than English to see the intentions of using those and how those benefits them. Also, the study could be expanded by increasing the sample size and including both the students and professionals in the survey.

This paper is one of the first to report on the use of ICTs in Uzbekistan, a developing country which features very rarely in the international academic literature. It lends further support to the UTAUT model, explaining between 58% and 64% of intended use of online learning videos. Educators wishing to supplement their curricula with online learning materials should heed the issues experienced by students such as slow bandwidth, dominance of English materials (some platforms offer real-time automatic translation) and overwhelming amount of material (a pre-selection of materials is recommended). Hopefully some of our findings can be relevant to similar contexts i.e. other small non-English-speaking developing nations.

References

1. Garg, A., Shukla, B.: IT implementation in the education sector: a review. In: International Conference of Electronics, Communication and Aerospace Technology, pp. 604–610. IEEE, Coimbatore (2017)
2. Cross, A., Bayyapunedi, M., Ravindran, D., Cutrell, E., Thies, W.: VidWiki: enabling the crowd to improve the legibility of online educational videos. In: 17th ACM Conference on Computer supported Cooperative Work and Social Computing, pp. 1167–1175. ACM, Baltimore (2014)
3. Breslow, L., Pritchard, D., DeBoer, J., Stump, G., Ho, A., Seaton, D.: Studying learning in the worldwide classroom: research into edX's first MOOC. Res. Pract. Assess. **8**, 13–25 (2013)
4. Social blade Homepage. https://socialblade.com/youtube/user/khanacademy. Accessed 11 Feb 2020
5. Korkut, S., Dornberger, R., Diwanji, P., Simon, B.P., Maerki, M.: Success factors of online learning videos. Int. J. Interact. Mobile Technol. **9**(4), 17–22 (2015)
6. Youtube Homepage. https://www.youtube.com/user/khanacademy/about. Accessed 11 Feb 2020
7. Guo, P.J., Kim, J., Rubin, R.: How video production affects student engagement: an empirical study of MOOC videos. In: Proceedings of the first ACM Conference on Learning @ Scale Conference, pp. 41–50. ACM, Atlanta (2014)
8. Pedrotti, M., Nistor, N.: User motivation and technology acceptance in online learning environments. In: Verbert, K., Sharples, M., Klobučar, T. (eds.) EC-TEL 2016. LNCS, vol. 9891, pp. 472–477. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-45153-4_45
9. MOOC Research Initiative (2014)
10. Roy, S., Garg, A.: Analyzing performance of students by using data mining techniques a literature survey. In: 17 4th IEEE Uttar Pradesh Section International Conference on Electrical, Computer and Electronics, pp. 130–133, IEEE, Mathura (2017)
11. Upadhyaya, D., Garg, A.: Leveraging on gamification and learning analytics for improved student learning. In: 2019 International Conference on Computing, Power and Communication Technologies, pp. 132–138. IEEE, Greater Noida (2019)
12. Persada, S.F., Miraja, B.A., Nadlifatin, R.: Understanding the generation Z behavior on D-learning: a unified theory of acceptance and use of technology (UTAUT) approach. Int. J. Emerg. Technol. Learn. **14**(05), 20–33 (2019)

13. Venkatesh, V., Morris, M.G., Davis, G.B., Davis, F.D.: User acceptance of information technology: toward a unified view. MIS Q. **27**, 425–478 (2003)
14. Khechine, H., Lakhal, S., Pascot, D., Bytha, A.: UTAUT model for blended learning: the role of gender and age in the intention to use webinars. Interdiscipl. J. E-Learn. Learn. Obj. **10**(1), 33–52 (2014)
15. Uğur, N.G., Turan, A.H.: E-learning adoption of academicians: a proposal for an ex-tended model. Behav. Inf. Technol. **37**(4), 393–405 (2018)
16. Pahnila, S., Siponen, M., Zheng, X.: Integrating habit into UTAUT: the Chinese eBay case. Pacific Asia J. Assoc. Inf. Syst. **3**(2) (2011)
17. Chian-Son, Y.: Factors affecting individuals to adopt mobile banking: empirical evidence from the UTAUT model. J. Electr. Comm. Res. **13**(2), 104–121 (2012)
18. Dulle, F.W., Minishi-Majanja, M.K.: The suitability of the unified theory of acceptance and use of technology (UTAUT) model in open access adoption studies. Inf. Dev. **27**(1), 32–45 (2011)
19. Alzahrani, M.E., Goodwin, R.D.: Towards a UTAUT-based model for the study of E-Government citizen acceptance in Saudi Arabia. Int. J. Soc. Behav. Educ. Econ. Bus. Ind. Eng. **6**(64), 376–382 (2012)
20. Pardamean, B., Susanto, M.: Assessing user acceptance toward blog technology using the UTAUT model. Int. J. Math. Comput. Simul. **1**(6), 203–212 (2012)
21. Badwelan, A., Drew, S., Bahaddad, A.A.: Towards acceptance m-learning approach in higher education in Saudi Arabia. Int. J. Bus. Manag. **11**(8), 12–30 (2016)

A Cybersecurity Curricular Framework for IT Undergraduates in South Africa

Lynn Futcher(✉), Kerry-Lynn Thomson, and Apelele Mbuqe

Nelson Mandela University, Port Elizabeth, South Africa
{lynn.futcher,kerry-lynn.thomson,s213317400}@mandela.ac.za

Abstract. Many organisations have reported a dramatic shortage of cybersecurity skills to counteract cyberattacks. Academia and industry should be responsible for working together to investigate possibilities in closing this cybersecurity skills gap. The IT curricula of many South African universities are based on the outdated IT2008 curricular guidelines, which make no mention of cybersecurity. In order to address this increasing cybersecurity skills gap, universities offering computer-related qualifications need to adapt their curricula accordingly. This paper proposes a framework for integrating cybersecurity into IT undergraduate curricula in South Africa. As the cybersecurity needs of industry continue to evolve, the proposed framework will provide a structure that aligns IT curricula to the cybersecurity workforce needs by considering the NIST NICE framework, CSEC2017, and IT2017. Furthermore, the proposed framework discussed in this paper will serve as an academic guide for communicating cybersecurity content to address specific industry work roles. The implementation of this IT Cybersecurity Curricular Framework aims to impact the development of the cybersecurity workforce positively and reduce the cybersecurity skills gap between academia and industry.

Keywords: Cybersecurity skills · IT2017 · CSEC2017 · NICE · IT curricula

1 Introduction

The newly proposed Cybercrimes and Cybersecurity Bill will soon bring South Africa in line with international laws dealing with cybercrime. Once enacted, it will necessitate alignment with the South African data protection Acts, RICA and POPIA. However, Communications Deputy Minister, Pinky Kekana, warns that it will be impossible to implement this Bill without a fully capable cybersecurity work force. She further stresses that a rigorous and coordinated effort between the private sector, government and academia is required to close the cybersecurity skills gap [19].

The explosive growth of cyberspace has provided an extended playground for hackers to test their skills by deceiving users, stealing data and committing

G. Wells et al. (Eds.): SACLA 2020, CCIS 1518, pp. 156–170, 2021.
https://doi.org/10.1007/978-3-030-92858-2_11

fraudulent activities. To counteract this, organisations must engage in cybersecurity practices. The International Telecommunication Union (ITU) [8] defines cybersecurity as *"the collection of tools, policies, security concepts, security safeguards, guidelines, risk management approaches, actions, training, best practices, assurance and technologies that can be used to protect the cyber environment and organization and user's assets."* However, small and large organisations continue to suffer at the hands of cybercriminals. Attackers ambush these organisations through cyberattacks that include, but are not limited to, ransomware, phishing, and malware. 71% of data breaches recorded in 2019 were financially motivated [8], and it is estimated that the average cost of a data breach was $3.92 million in 2019, up from $3.86 million in 2018. Further, the cost of a mega breach, where more than 1 million records were compromised, averaged $42 million per breach [21].

Even though cybersecurity is essential for organisations, 33% of cybersecurity professionals indicated that their organisations are significantly impacted by the cybersecurity skills shortage, while 41% stated that their organisations are somewhat impacted [7]. In 2015, a job forecast by Symantec predicted that the huge demand for cybersecurity professionals would increase to six million globally by the end of 2019. Jobs forecast for cybersecurity have failed to keep up with the rise of cyberattacks, which is estimated will have a global cost of $6 trillion annually by 2021 [15]. Even though there seems to be an increasing pool of Information Technology professionals, there remains a shortage of cybersecurity professionals to close the cybersecurity skills gap. Organisations are desperate for professionals that can create secure software, design safe computer code and design tools to prevent, mitigate and detect cyberattacks [6].

A recent study revealed that 79% of cybersecurity professionals began their careers in IT and then transitioned to cybersecurity over time [7]. The study further advised organisations to start recruiting cybersecurity talent from IT departments inside or outside their organisation [7]. If this is the case, why do the majority of South African IT curricula contain limited cybersecurity content?

Various ACM/IEEE curricular documents, that provide computing curricular guidelines and recommendations, emphasise that institutions offering computing qualifications have a huge responsibility to integrate cybersecurity throughout the curricula of their programs. It can, therefore, be concluded that cybersecurity is essential in order to ensure that computing programs meet the needs of industry. It can also be argued that failure to integrate cybersecurity within computing programs might result in students lacking the cybersecurity knowledge, skills and abilities needed in industry. Such negligence means that network systems, information systems, IT infrastructure and other IT assets designed by these students after graduating, could have security misconfigurations and vulnerabilities [2].

After a preliminary content analysis conducted of the IT curricula in five South African universities, it was concluded that even though security and cybersecurity are addressed in some IT curricula in South Africa, it is insufficient. This apparent lack of cybersecurity education across South African IT curricula could be detrimental to organisations that hire IT graduates, and thus continue to con-tribute to the cybersecurity skills gap that is currently on the rise between academia and industry.

Many of the IT curricula of South African universities are based on the outdated IT2008 curricular guidelines, which makes no mention of cybersecurity. This leads to the identified problem that *"currently many universities in South Africa have not yet adapted their IT undergraduate qualifications to address the changing cybersecurity demands of industry"*. The purpose of this paper, therefore, is to propose a Cybersecurity Curricular Framework to facilitate the education of IT undergraduates in South Africa.

This has been achieved by addressing three research questions:

1. What comprises the South African IT undergraduate curricular landscape?
2. What are the ACM/IEEE cybersecurity recommendations for IT curricula?
3. What aspects of international cybersecurity documents should be considered in IT undergraduate curricula in South Africa?

The methodology followed for this research study consisted of a literature review, a content analysis, a critical analysis, modelling and argumentation. The literature review identified current literature relevant to IT curricula that seeks to address the cybersecurity skills gap between academia and industry, while a con-tent analysis was conducted to understand the South African IT undergraduate curricula landscape. Furthermore, a critical analysis was conducted on various ACM/IEEE and NIST publications to determine the cybersecurity components relevant to IT curricula and the cybersecurity knowledge, skills and abilities required of various IT roles in industry. The cybersecurity components that were identified were used to model and argue towards the proposed framework.

The rest of the paper is outlined as follows. Section 2 provides a brief overview of the South African IT undergraduate curricular landscape. Section 3 discusses related literature, while Sect. 4 provides a high-level overview of the relevant ACM/IEEE and NIST publications and presents the identified key cybersecurity components from these documents. Sections 5 and 6 propose the IT Cybersecurity Curricular Framework, and its implementation within a South African university, and Sect. 7 concludes this paper.

2 The South African IT Undergraduate Curricular Landscape

The South African Higher Education system is divided into three categories; namely, Universities of Technology, Comprehensive Universities and Traditional Universities. While Universities of Technology focus on vocationally orientated education, Comprehensive Universities offer a combination of academic and vocational diplomas and degrees, and Traditional Universities offer theoretically orientated university degrees. South Africa has eight Universities of Technology, six Comprehensive Universities and twelve Traditional Universities.

Of these 26 South African universities, 23 offer computing qualifications, including Computer Science (CS), Information Systems (IS) and Information Technology (IT); 18 universities offer Computer Science; 14 offer Information Systems; and 13 offer Information Technology. In terms of this research, the focus is on South African IT undergraduate curricula.

In order to understand the South African IT undergraduate curricular landscape, a content analysis was conducted on the curricula documents taken from various university websites. Only five of the 13 universities offering IT were included in the study as the research only considered those universities that had detailed curricular documents available on their websites.

Table 1 provides a summary of the results of the content analysis conducted for the five South African universities, indicating that many of their undergraduate IT modules address one or more of the IT pillars as presented in IT2008. The main focus of the various IT modules analysed was on Programming, Databases and Networking, with lesser focus on Web Systems and Human Computer Interaction. In addition, the coverage of Information Assurance and Security (IAS) across these modules varied vastly, with U2 and U5 showing a coverage of 53.33% and 50% respectively, while U1, U3 and U4 showed less than 20% coverage. However, in all cases the actual breadth and depth of the various security topics could not be determined, but only that the various modules addressed security to some extent. In addition, all of the universities, except U4, appeared to offer at least one dedicated security-related module at undergraduate level. While it is clear that there is an attempt to cover certain security aspects in the South African IT undergraduate curricula, the coverage of security is not sufficient to address the cybersecurity skills gap.

Table 1. Summarised results of content analysis.

University	#modules covering ≥1 of 5 pillars	#modules integrating IAS	%IAS coverage	#dedicated security modules
U1	25	3	12.00	3
U2	15	8	53.33	0
U3	21	2	9.52	1
U4	19	3	15.79	1
U5	22	11	50.00	2

3 Related Literature

In order to address the global cybersecurity crisis, academic institutions worldwide are required to address the demand for cybersecurity professionals [13]. According to [3], *"cybersecurity is both a discipline in its own right and that cybersecurity should be taught within computer science and related degrees"*, including IT. In addition [3] mentions that all people working in IT need varying levels of cybersecurity knowledge, skills and abilities.

This need is partially addressed by [5] who proposed an IT Student Learning (Security-related) Taxonomy by combining Bloom's Taxonomy's six levels of thinking with Webb's Depth of Knowledge Model addressing the breadth and depth of learning required at various levels. In addition, they developed an IT Security-related and Cybersecurity Curriculum Taxonomy showing the integration of security-related topics throughout the curriculum. However, a study in South Africa by [20] noted a number of perceived challenges for integrating security into undergraduate computing curricula, including: computing curricula already being overloaded, lecturers' resistance to change and lecturers lacking the relevant 'know-how'.

In South Africa, several sectors suffer from cybersecurity negligence. For example, [9] asserts that many small, medium enterprises (SME) in South Africa have experienced a spate of cyberattacks and that they do not have sufficient knowledge, skills and abilities to protect their organisations against such cyberattacks. In addition, [12] revealed that the South African banking industry has experienced a rapid increase in cyberattacks due to their affiliation with various third party systems, which often introduce vulnerabilities. IT professionals in these organisations have a responsibility to protect their organisations when such cyberthreats arise.

[16] recently conducted a study to determine the knowledge, skills and qualifications required of cybersecurity professionals in South Africa. Their study found that 52% of job advertisements for cybersecurity professionals required a Bachelors' degree in Computer Science, Information Systems, Engineering or a similar field. Furthermore, the highest demand was found to be for technical skills (42%), followed by interpersonal skills (21%) and business skills (16%). In addition, the greatest demand for cybersecurity professionals was found to be in the Financial Services industry, followed by the IT Services industry. However, their study did not align the required knowledge, skills and abilities to specific work roles. As a result, this research relied on NIST's NICE framework as it identifies the cybersecurity knowledge, skills and abilities required in industry.

According to [11], the NICE framework *"provides the basis to link knowledge-based (breadth) learning with the skills-based (depth) making it possible to map a set of learning objectives and experiences that lead to a high level of expertise"*. However, for many lecturers of cybersecurity, knowledge-based learning is most appealing as it fits easily into a traditional classroom and can be taught in an instructor-centred fashion. Skills-based learning, on the other hand, requires one to apply one's knowledge and skills effectively in a real-world environment [11].

[17] asserts that all computing and technology related programs have a responsibility to ensure that security is thoroughly and pervasively integrated into their curricula. In addition, they state that IT programs are well-suited to incorporating cybersecurity through the five pillars as presented in IT2008. Therefore, it is argued that current South African IT undergraduate curricula, which are based on these five pillars, can incorporate cybersecurity.

4 Identified Key Cybersecurity Components

Considerable attention has been given to cybersecurity by key role players for recommendations and guidance in various computing fields. Multiple organisations have developed cybersecurity guidelines and recommendations for universities and the workforce. These key role players represent a community effort with input from academia and industry [4]. These key role players include NIST and the ACM/IEEE. This study focused on various documents produced by these role players, and deemed relevant to the South African context, and thus shaped the key components of the proposed IT Cybersecurity Curricular Framework.

4.1 The ACM/IEEE IT Curricular Component

The ACM/IEEE IT2008 document outlines the pillars of the IT discipline including Programming, Networking, Human-Computer Interaction, Databases, and Web Systems. In the ACM/IEEE IT2008 curriculum document, security was included as a knowledge area called Information Assurance and Security (IAS) [10]. The need and importance for security in the IT discipline is emphasised in this report. Furthermore, the IT2008 [10] report states that security has to be addressed across the full scope of the IT discipline as indicated by the above-mentioned IT pillars.

Since the release of IT2008, the demand for IT capabilities has increased and has become embedded in everything around us. In 2017, Information Technology 2017 (IT2017) was released by the ACM/IEEE [18]. One of the most notable transformations in this curricular report was the inclusion of cybersecurity, included as Essential and Supplemental Domains. While the Cybersecurity Essential Domain refers to the cybersecurity competencies that all students should achieve while completing the IT program, the Cybersecurity Supplemental Do-mains consist of cybersecurity competencies in which students perform more specialised work that is in line with the program goals [18].

This research organised the IT discipline into pillars, as suggested by IT2008, as most South African universities have used this as a basis for their IT qualifications. As already stated, although the IT2008 [10] document included IAS, cybersecurity was not part of this report. However, there has recently been an increase in the cybersecurity skills gap between academia and industry. To counteract this, the newly updated ACM/IEEE IT2017 [18] curricular document, has included cybersecurity. In the IT2017 curricular report, the ACM/IEEE identified IT cybersecurity domains that are both essential and supplemental to an IT curriculum.

The Cybersecurity Essential Domain refers to the cybersecurity competencies that all students should achieve while completing the IT program. Therefore, Fig. 1 depicts the Cybersecurity Essential Domain as an umbrella that overarches the IT discipline, similar to that of IAS within IT2008. This means that the Cybersecurity Essential Domain must be integrated across the IT curriculum regardless of the IT pillar being addressed.

Figure 1 also shows the Cybersecurity Supplemental Domain, which consists of various cybersecurity competencies in which students perform more specialised work according to the specific program goals. The Cybersecurity Supplemental Domain helps define the purpose and goal of what each IT pillar aims to address.

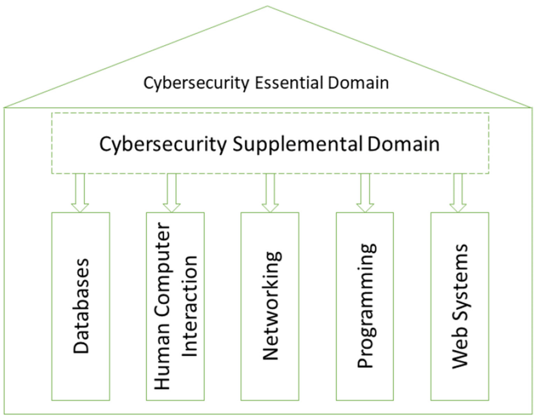


Fig. 1. Cybersecurity essential and supplemental domain within the IT discipline (Adapted from [10,18])

4.2 The NICE Component

The National Initiative for Cyber Security Education (NICE) [14], led by NIST of the U.S Department of Commerce, is the result of a combined effort from the government, academia and the private sector. This publication provides a fundamental reference resource to reinforce a workforce that is able to meet the cyber-security needs of the organization.

The NICE framework establishes a scheme that divides work into categories, speciality areas, work roles, knowledge, skills and abilities, and tasks. These NICE framework core components assist different organisations in managing cybersecurity roles and responsibilities. In terms of industry, NIST's NICE framework [14] presents the cybersecurity skills, knowledge and abilities needed by the workforce. In addition, the various components of the NICE framework help improve the communication required to identify, recruit and develop cybersecurity skills and talent.

Figure 2 illustrates the connection between the various NICE components, namely:

- *Categories* that provide the general structure of the NICE framework.
- *Speciality Areas* that represent a group of specialised work areas within cybersecurity for each category identified.
- *Work Roles* that provide a grouping of related cybersecurity work within each speciality area identified.

– *Knowledge, Skills and Abilities (KSAs)* which are needed to perform each work role and
– *Tasks* which include pieces of work that are specifically defined.

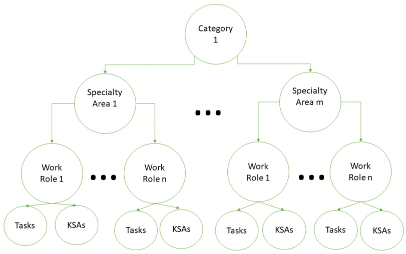


Fig. 2. NICE components [14]

The NICE framework's cybersecurity knowledge, skills and abilities (KSAs) enable employees to perform specific tasks and to connect those tasks with work roles associated with various disciplines. For example, in the case of an IT qualification that focuses on the Programming pillar, there is a related *Software Development* speciality area and a *Software Developer* work role.

Furthermore, according to NIST [14], a student pursuing an IT Software Development qualification should possess, for instance:

– Knowledge of secure coding techniques,
– Skills in conducting vulnerability scans and recognising vulnerabilities in security systems, and
– The ability to develop secure software according to secure software deployment methodologies, tools, and practices

In addition, relevant cybersecurity knowledge, skills and abilities will enable, for example, the IT Software Development graduate to carry out tasks related to the Software Developer work role. According to NIST [14], the tasks related to the Software Developer work role include, but are not limited to:

– Applying secure code documentation,
– Identifying basic common coding flaws at a high level,
– Capturing security controls used during the requirements phase to integrate security within the process, to identify key security objectives, and to maximise software security while minimising disruption to plans and schedules.

In essence, and according to this example, the NICE framework serves as a valuable reference and would enable IT graduates to satisfy the industry requirements for a Software Developer, if the identified KSAs are integrated into the IT curriculum.

4.3 The ACM CSEC2017 Component

Alongside the various ACM/IEEE curricular documents that promote the integration of cybersecurity into computing curricula is the Cybersecurity Curricular (CSEC2017) [1] document. The CSEC2017 document presents a cybersecurity model that comprises of three parts, namely: knowledge areas, cross cutting concepts and disciplinary lenses.

Knowledge areas provide a structure that organises the content of cybersecurity. Each knowledge area consists of critical and important knowledge that is essential across computing disciplines, while cross-cutting concepts provide an underlying basis for the student's ability to understand the knowledge area. Furthermore, cross-cutting concepts promote the security mindset carried out by each knowledge area. As seen in Fig. 3, knowledge areas consist of knowledge units that comprise related topics and their associated learning outcomes. The disciplinary lens drives the application of specific knowledge from knowledge areas and cross-cutting concepts. Moreover, the disciplinary lens explores the depth of content, learning outcomes and approach from knowledge areas and cross-cutting concepts, and how they will be tailored to fit a particular discipline.

Figure 3 shows that the various computer disciplines can address the CSEC2017 cybersecurity content differently through the identified knowledge areas. In terms of this paper, the focus is on Information Technology.

The CSEC2017 knowledge areas are flexible structures in the thought model that permit contraction and expansion of cybersecurity content as required. This body of knowledge is structured according to knowledge areas that could be used to provide cybersecurity concepts filtered according to the different IT pillars. Cross-cutting concepts can be applied to a wide range of programs within the IT discipline. An IT program should include content from the Cybersecurity Essential Domain, regardless of the program's focus. Similarly, cross-cutting concepts should be addressed throughout the IT program. Figure 4 shows that the various cross-cutting concepts should overarch every knowledge area. Within the IT discipline, knowledge areas can reflect on the goals and purpose of the IT program, while cross-cutting concepts provide individuals with the ability to understand those knowledge areas. It can be asserted that since cross-cutting concepts help in understanding the knowledge areas, it is essential that an IT program should address these concepts regardless of the pro-gram focus. As previously stated, the Cybersecurity Supplemental Domain helps define the purpose and goal of what each IT pillar aims to address. For example, the Connection Security knowledge area offers knowledge units and topics applicable to the Networking pillar, and the Software Security knowledge area has knowledge units and topics targeting the Programming pillar. It is important to note that these examples do not imply that an IT pillar can only extract cybersecurity content from one knowledge area. This is evident in the Data Security knowledge area which encompasses knowledge units and topics that apply to several IT pillars.

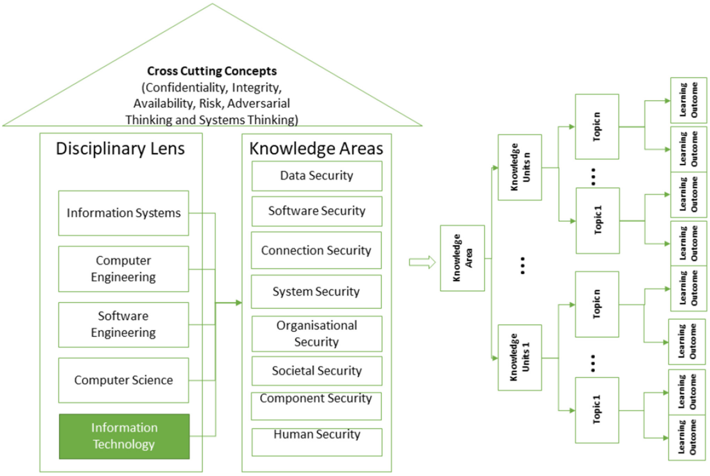


Fig. 3. The CSEC2017 component (adapted from: [1])

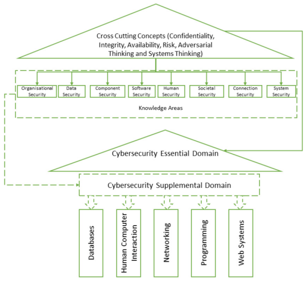


Fig. 4. The CSEC2017 component mapped against the IT discipline

5 The Proposed Framework

In order to produce IT graduates who are capable of making conscious cybersecurity decisions, a Cybersecurity Curricular Framework is proposed and presented in Fig. 5. The proposed framework acknowledges that there are various computing disciplines that should incorporate cybersecurity education, including Information Systems, Computer Science, Computer Engineering, Software Engineering and Information Technology. However, the focus of this study is specifically on IT.

Furthermore, it acknowledges that academia is responsible for addressing the cybersecurity skills gap in industry. This skills gap can be identified by understanding the various NICE categories, speciality areas and work roles within each speciality area, together with the associated tasks, knowledge, skills and abilities required to satisfy these work roles. However, many IT academic programs in South Africa typically focus more on specific work roles rather than on the broad range as provided by NICE (for example, Software Developer and Network Operations Specialist work roles). As academics, it would therefore be important to address the cybersecurity needs of these work roles by considering the guidelines as provided by CSEC2017 and IT2017.

CSEC2017's cross-cutting concepts of confidentiality, integrity, availability, risk, adversarial thinking and systems thinking need to be integrated into all aspects of cybersecurity education thus forming part of the Cybersecurity Essential Domain as presented in IT2017. Depending on the focus of the specific IT module being taught, specific cybersecurity topics and learning outcomes can be drawn from CSEC2017's knowledge areas, thus fitting into the Supplemental Domain as presented in IT2017. For example, a databases module will primarily draw cybersecurity related topics from the Data Security knowledge area of CSEC2017. Similarly, a programming module will primarily draw related cybersecurity topics from the Software Security knowledge area, but could also draw relevant topics from the Component Security knowledge area.

Universities offering IT qualifications in South Africa (and globally) should thus address the cybersecurity skills gap and the high industry demand for these skills by considering the IT2017 curricular guidelines in conjunction with the NICE framework and CSEC2017. The proposed Cybersecurity Curricular Framework aims to help educators prepare IT graduates for work roles defined by the NICE framework. In addition, CSEC2017 serves as the bridge that presents topics and learning outcomes for preparing IT graduates for specific work roles, ensuring the integration of relevant knowledge, skills and abilities into the IT curriculum.

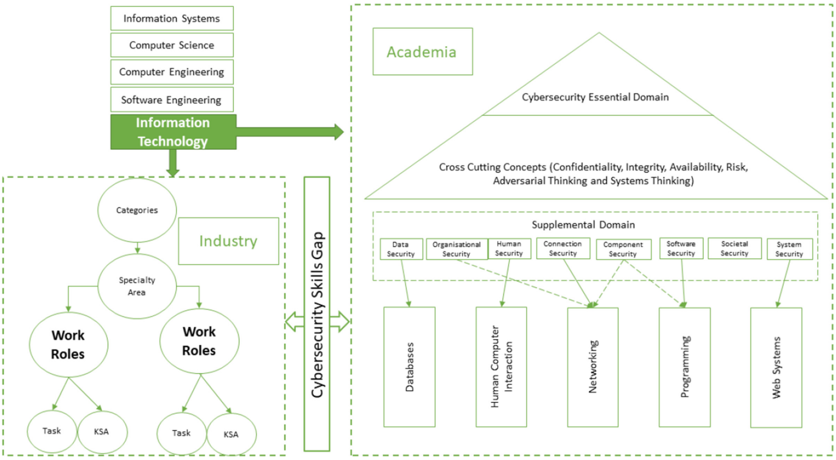


Fig. 5. The proposed cybersecurity curricular framework

A critical success factor for the implementation of such a framework would be the buy-in from educators who are already facing several challenges. [20] refer to such challenges for integrating information security into undergraduate computing curricula. These include the limited time that educators have to work through the existing curriculum without the additional burden of considering information security. In addition, it was found that educators are often resistant to change and would have to be motivated to integrate information security into their particular modules. This motivation could be provided by Heads of Departments who should show support and provide directives to educators within the department. It was also noted by [20] that computing educators may lack the necessary information security "know-how" to enable them to integrate information security into the curriculum. These challenges would most likely also hold true for integrating cybersecurity into the IT curriculum.

The following section describes the implementation of the proposed framework within typical South African IT programs.

6 Implementation of the Framework

From the preliminary content analysis conducted of five South African IT curricula, it was evident that South African IT undergraduate curricula are dominated by Programming and Networking. For this reason, these two IT pillars were chosen for contextualising the implementation of the proposed framework. The NICE work roles that were chosen for the contextual examples include Network Operations Specialist (WORK ROLE:OM-NET-001) and Software Developer (WORK ROLE:SPDEV-001).

The Network Operations Specialist work role has a well-defined description in the NICE framework. However, this information alone will not be enough to link the work role to the applicable CSEC2017 knowledge areas, knowledge units and topics. The NICE KSA and task listings provide extensive details regarding NICE work roles; thus, providing a basis for the process of mapping to IT2017 and CSEC2017. Curricular guidance for the Network Operations Specialist can be mapped to various knowledge areas in CSEC2017.

For example, Secure Communication Protocol knowledge units found in the Data Security knowledge area, cover some networking topics, for instance, the Internet/Network layer. Connection Security is, however, the main knowledge area that addresses the Networking pillar, as it focuses on digital communications and networks. The Network Operations Specialist must be able to configure and optimise network hubs, routers, and switches (Task ID: Task ID: T0035). To accomplish this task, an IT graduate should have the ability to operate network equipment that includes hubs, routers, switches, bridges, servers, transmission media, and related hardware (KSA ID: A0052).

To further prepare for this work role, IT graduates must possess skills in installing, configuring, and troubleshooting LAN and WAN components such as routers, hubs, and switches (KSA ID: S0041). In addition, they should know about information technology security principles and methods (KSA ID: K0049).

According to NIST [14] the KSAs and tasks of a Software Developer, also focus on security aspects of software development. CSEC2017's Software Security knowledge area focuses on security aspects during the software development life-cycle and provides curricular guidance for IT graduates studying towards the Software Developer work role. Some Software Developer KSAs are addressed in multiple knowledge areas and therefore may require topics from outside the Software Security knowledge area to address some of its KSAs.

As an example of how an IT graduate could be prepared for the NICE framework role of Software Developer (Work role ID: SP-DEV- 001), topics that address some of the Software Developer work role KSAs have been extracted from CSEC2017's Connection and Component Security knowledge areas. IT graduates targeting the Software Developer position must know software design tools, methods, and techniques (KSA ID: K0080). In addition, they must possess the skills to apply cybersecurity and privacy principles to organisational requirements (KSA ID: S0367) and should have the ability to develop secure soft-ware according to secure software deployment methodologies, tools, and practices (KSA ID: A0047). To achieve these KSAs, in this example, CSEC2017 provides the topics to address techniques for integrating security during software design. These topics can be mapped to the Design knowledge unit, in the Soft-ware Security knowledge area. Once these topics have been integrated into the IT undergraduate curriculum, the Software Development graduate must be able to design, develop, and modify software systems, using scientific analysis and mathematical models to predict and measure outcomes and consequences of design (Task ID: T0057).

This is one of the tasks that IT graduates should be able to accomplish within the Software Developer work role. Furthermore, the Software Development Life cycle/Security Development Life cycle topic discusses the design tools like waterfall model, agile development and security. This topic can provide Software Development students with knowledge of software development models as defined by KSA ID: K0081. In addition, a Software Developer must know computer networking concepts and protocols, together with network security methodology (KSA ID: K0001). This KSA could be mapped to topics in the Connection Security knowledge area, under the Networking Service and Network Architecture knowledge units. From this discussion, it is clear that the Software Security knowledge area alone does not encompass knowledge units and topics addressing the KSAs required of a software developer.

7 Conclusion

As organisations continue to fight against cyberattacks, the cybersecurity skills gap between academia and industry continues to grow. While most organisations believe technical interventions like firewall installations might mitigate the exposure to these cyberattacks, the literature argues that the current cybersecurity workforce does not have sufficient knowledge to counteract these cyberattacks. As a result, the shortage of cybersecurity skills has led to some organisations offering guidance in an attempt to close the cybersecurity skills gap between academia and industry.

To produce IT graduates who are capable of making conscious cybersecurity decisions, this paper proposed a framework for integrating cybersecurity into IT undergraduate curricula in South Africa. Since cybersecurity needs continue to evolve, the proposed framework provides a structure that aligns IT curricula to the cybersecurity workforce needs by considering the NIST NICE framework, CSEC2017, and IT2017. Furthermore, the proposed framework serves as an academic guide for communicating cybersecurity content to address specific industry work roles. The implementation of this IT Cybersecurity Curricular Frame-work could positively impact the development of the cybersecurity workforce in South Africa and reduce the cybersecurity skills gap between academia and industry.

The limitation of this study is that the proposed IT Cybersecurity Curricular Framework has not yet been implemented at a university offering an IT qualification. However, this could provide interesting future research to verify whether it would practically be feasible. We also acknowledge the need for further research to identify cybersecurity industry needs within the South African context.

References

1. Burley, D., Bishop, M., Buck, S., Ekstrom, J., Futcher, L.: ACM/IEEE 2017 Cybersecurity Curricula 2017. Technical report, December 2017. https://cybered.hosting.acm.org/wp-content/uploads/2018/02/newcover_csec2017.pdf
2. Cobb, S.: Sizing the cyber security skills gap: a white paper, July 2016. https://www.researchgate.net/publication/309456549
3. Crick, T., Davenport, J., Irons, A., Prickett, T.: A UK case study on cybersecurity education and accreditation. In: 2019 IEEE Frontiers in Education Conference (FIE), pp. 1–9 (2019)
4. Dodge, R.C.: Information assurance and security in the ACM/IEEE CS2013. In: Dodge, R.C., Futcher, L. (eds.) WISE 2009/2011/2013. IAICT, vol. 406, pp. 48–57. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-39377-8_6
5. Harris, M.A., Pattern, K.: Using Bloom's and Webb's taxonomies to integrate emerging cybersecurity topics into a computic curriculum. J. Inf. Syst. Educ. **26**(3), 219–234 (2015)
6. Herjavec Group: Cybersecurity Ventures 2017 Cybersecurity Jobs Report. Technical report (2017). https://www.herjavecgroup.com/wp-content/uploads/2017/06/HG-and-CV-The-Cybersecurity-Jobs-Report-2017.pdf
7. IBM: Security Cost of Data Breach (2019). https://www.ibm.com/downloads/cas/ZBZLY7KL?_ga=2.70882194.329793785.1580983939-1038055541.1580983939&_gac=1.17016139.1580983939.CjwKCAiAj-_xBRBjEiwAmRbqYksOMIXk29yaRNn0LXV2mnVW16Y42yBcqpiMOc2br1AHTPZWhrqXsBoCBlUQAvD_BwE
8. International Telecommunication Union: Overview of Cybersecurity (2019). https://www.itu.int/rec/T-REC-X.1205-200804-I
9. Kabanda, S., Tanner, M., Kent, C.: Exploring SME cybersecurity practices in developing countries. J. Organ. Comput. Electron. Commer. **28**(3), 269–282 (2018). https://doi.org/10.1080/10919392.2018.1484598
10. Lunt, B., et al.: ACM/IEEE 2008 Information Technology 2008 Curriculum Guidelines for Undergraduate Degree Programs in Information Technology. Technical report, Association for Computing Machinery (ACM), IEEE Computer Society (2008). https://doi.org/10.1145/2600000

11. Manson, D., Pike, R.: The case for depth in cybersecurity education. ACM Inroads **5**(1), 47–52 (2014)
12. Mbelli, T.M., Dwolatzky, B.: Cyber security, a threat to cyber banking in South Africa: an approach to network and application security. In: 2016 IEEE 3rd International Conference on Cyber Security and Cloud Computing (CSCloud), pp. 1–6. IEEE (2016)
13. McGettrick, A.: Toward effective cybersecurity education. IEEE Secur. Priv. **11**(6), 66–68 (2013)
14. Newhouse, W., Keith, S., Scribner, B., Witte, G.: NIST 2017 national initiative for cybersecurity education (NICE) cybersecurity workforce framework. NIST special publication 800-181 (2017). https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-181.pdf
15. Oltsik, J., Alexander, C.: Enterprise Strategy Group. The Life and Times of Cybersecurity Professionals. Technical report, November 2019. https://www.esg-global.com/hubfs/pdf/ESG-ISSA-Research-Report-Life-of-Cybersecurity-Professionals-Apr-2019.pdf
16. Parker, A., Brown, I.: Skills requirements for cyber security professionals: a content analysis of job descriptions in South Africa. In: Venter, H., Loock, M., Coetzee, M., Eloff, M., Eloff, J. (eds.) ISSA 2018. CCIS, vol. 973, pp. 176–192. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-11407-7_13
17. Rowe, D.C., Lunt, B.M., Ekstrom, J.J.: The role of cyber-security in information technology education. In: Proceedings of the 2011 Conference on Information Technology Education, pp. 113–122 (2011)
18. Sabin, M., Alrumaih, H., Impagliazzo, J., Lunt, B., Zhang, M.: ACM/IEEE. 2017 Information Technology Curricula 2017: Curriculum Guidelines for Baccalaureate Degree Programs in Information Technology. Technical report (2017). https://doi.org/10.1145/3173161
19. South African Government News Agency: Government's vision to address cybersecurity skills shortage. sanews.gov.za (2019). https://www.sanews.gov.za/south-africa/governments-vision-address-cybersecurity-skills-shortage
20. Thomson, K., Futcher, L., Gomana, L.: Towards a framework for the integration of information security into undergraduate computing curricula. S. Afr. J. High. Educ. **33**(3), 155–175 (2019)
21. Verizon: Data breach investigations report (2019). https://enterprise.verizon.com/resources/reports/dbir/2019/summary-of-findings/

Interactive Learning: Introducing a First-Year Systems' Analysis and Design Course

Adriana A. Steyn(✉), Adriana J. M. Botha, Dané Coetzee, and Marissa de Villiers

University of Pretoria, Pretoria, South Africa
riana.steyn@up.ac.za

Abstract. Education has recognized the need to change its teaching approach. No longer can we only rely on face-to-face classroom interactions, or even PowerPoint slides, to try and get the students to learn. Students entering our campuses do not want to "just" sit and see what information is coming. No, they want to explore, they want to test, they want to experience and, conveniently, using technology they understand. As academics, we need to understand how they learn and perhaps see if technology can be used to enhance the learning process, following the learning science approach. This paper introduced the design of an interactive infographic, which introduces students to the basic system's theory as part of a first-year systems analysis and design course at an urban university. The design of one A4-infographic replaced 91 slides. The infographic, made available through the learning management system of the University, allowed the researchers to track the actual access and usage thereof. A survey distributed to the class of 375 students, resulted in 99 usable responses. A total number of 223 students worked through the infographic spending on average 26.7 min viewing the infographic. The results revealed the students enjoyed the infographic, but, more importantly, students feel they understand the content better and can apply the acquired knowledge on the test. Students further felt that the infographic could be used in explaining the study material, learning activities and summarising the content. Our study suggests that although students all have smartphones, the majority still preferred to open and work through the infographic using a laptop. Another result indicated that students engage with the content on a much deeper level, in their own terms using familiar technologies. They also had more time to work through the content in their own time. Students feel they understand the material better and are able to apply it in various learning activities. Supplementing the PowerPoint slides through the use of an interactive infographic made students engage far better with the content, thus, creating a more in-depth learning experience.

Keywords: Blended learning · Education · Hybrid learning · Higher education · Interactive infographic · Innovative technology · Systems analysis and design · Undergraduate teaching

G. Wells et al. (Eds.): SACLA 2020, CCIS 1518, pp. 171–186, 2021.
https://doi.org/10.1007/978-3-030-92858-2_12

1 Introduction

Students entering our campuses today have a much higher need to be engaged in their world [24, p. 55]. These students are unique, diverse and differ significantly from students that came before them in their needs, aspirations, expectations and perspectives [9,20].

As educators, we have to win back their trust, see the world through their eyes, viewing technology as they do [24,33]. In order to correctly educate these students, we need to understand how they think, learn, and consider these unique characteristics to adapt our educational approaches accordingly. Teaching these students in ways that suit them best should help them engage in the learning process and stay interested in the content that is taught [8,26].

To adapt and engage effectively with these students, as educators, we should lead the changing educational environment [3,17]. As these students are entering our classrooms, we need to understand that they are using multi-modalities to learn, and as lecturers we need to make provision for that through our instruction and facilitation. Despite the recent view that learning styles, per se, should not inform how we teach, research indicates that more and more people are becoming visual learners [7,13], and the need to use more visual aids as a teaching mechanism has increased. Delivering content using more technology and, more effectively, incorporating more visuals should be our focus–a tall order for academics.

In an attempt to solve the problem of learning and teaching in the real world [28,30], Davidson et al. [9, p. 1] speak of getting students to become "active participants in the learning process", linking in with the learning science theories, developed over 30 years ago. This means not just having students in a classroom listening to a lecturer but also learning something. To do this, one could investigate and apply the learning sciences - a field of study related to psychology [14] that investigates and explores teaching and learning in a real-world context [30]. Steyn et al. [32], for example, found that students understood difficult concepts more when creating their own content. The use of technology is becoming more prevalent in both everyday life and educational environments [18] and forms an integral part of the life of these students [36]. They are continually interacting with technological devices and the Internet, hardly knowing life without it [22]. These technologies can be harnessed to enhance and adjust traditional teaching methods to be more effective for their learning styles [11,29]. One of the tools identified that could perhaps help address these needs, are infographics [3,13], also referred to as information graphics. Linking in with the more visual aspects of learning, Yildirim [43] as well as Steyn et al. [33] also mentions the notion of using infographics as a new type of material to assist the student in understanding the material they are working with, providing them first with a'big-picture' of the outcomes to be achieved and secondly giving them the opportunity to work through the different parts that contribute to the build-up of the bigger picture. It was noted by Dunlap and Lowenthal [13], Alyahya [1] as well as Aydin et al. [3] that there is limited literature based on educational infographics, and the application thereof.

The authors of this paper argue that even less literature on the creation of an interactive infographic for the higher education context exist. This paper introduces an interactive infographic to a first-year undergraduate systems analysis and design class. The next section will introduce interactive learning as well as the tool used, the Infographic, in this study through the learning science approach.

2 Literature Review

Hybrid learning, blended learning, and flipped classroom, are some of the terms used in the higher education environment to enhance the learning experience as well as developing life-long learning [5,6,41,43] through various technology initiatives [5,6,41,43]. By combining technology and teaching, we give the students the best of both worlds [5,25,40]. However, the notion of giving technology needs to be carefully planned and designed as we cannot introduce technology into our learning without careful planning and deliberate outcome intentions for its use [13,24]. Many authors have explored increasing visual learning as the need to increase visuals in the education space has increased [1–3,16,23]. Using our eyes are considered the most important learning tool that we have [13] and that visuals are considered as the power tools of teaching. Yildirim [43] states that no matter what visual aid, tool or technology one uses, each aid has its information properties, intended use and learner preferences. A tool that could be used to assist in this hybrid teaching space are infographics which has been extensively investigated and published on various occasions [1,3,7,10,13,27,34,35,38,43]. The notion of incorporating an interactive approach into infographics will now be discussed.

2.1 Interactive Teaching

What Is Interactive Teaching. Interactive learning is the combination of traditional learning with the use and aiding of technology. It is referred as an e-learning environment. Teachers give instructions to students, and the students can use technology in the classroom to promote education [39]. Interactive teaching methods requires a great deal of innovation and technology. An interactive classroom or teaching environment is an excellent way to encourage innovation, creativity, interaction between the students and the teachers but also between the students themselves. It creates an opportunity for both the students and the teachers or lecturers to express their own opinions, thought and ways of thinking with each other. This provides students a chance to learn from each other. But selecting a specific interactive method or approach can be a daunting task, the next section will look more into the methods as well as challenged one typically face with interactive teaching methods and approaches.

Interactive Teaching Methods Used in Education. The focus of modern education should be on "student's independent activity, the skills of self-learning environments and experimental and practical training, where students have a choice of actions and can use initiative. This will allow for flexible training programs where students can work in" [42, p. 75]. These interactive methods of training and teaching, encourages interest in the profession, promote the efficient acquisition of training material, promotes conduct and provides high motivation, strength and knowledge. The most important skill that it will contribute is the intricate ability of future specialists. This method of receiving, processing and retaining information has the "potential to form the competence of future professionals" [42, p. 1]. This paper investigates an interactive infographic which was made available on a Urban University's LSM, where students could work through it in their own time, and pace and where students can decide which part they want to focus on and for how long.

Challenges Faced with Interactive Teaching. With the technology becoming such a big and required factor in our lives, schools and universities have started implementing and incorporating technology into their syllabus. Some schools and universities present certain courses entirely online.

A comparison study of online and traditional student withdrawal rates at a university in the north-eastern United States [4] found that although interactive and online classes have many benefits and is by far the superior way of teaching nowadays, it does however, produce some challenges and disadvantages. One of these challenges is the massive fluctuation in class attendance at university level [4]. Challenges like this caused the higher education institutions to work on and develop online curricula to meet these new challenges and demands.

Interactive and online courses are very impersonal due to the fact that there is a prerecorded lecture that is uploaded online and then viewed by the students at a later stage. This provides very little opportunity for the students to ask questions about the work being presented. The lack of personal interaction between students and the lecturer as well as the "perception that the online class format makes it more difficult for the professor or lecturer to engage with the students creates dissatisfaction" [4, p. 95]. To overcome some of the challenges, the content was only provided after the students were introduced to the content in class. To ensure a more "personal" experience, the voice recordings used during the interactive activities was one lecturer's voice and the examples used was also the examples worked through with the students during classes. The videos designed for the infographic, except for one, was also created by the lecturers to ensure that the content is exactly what the students needed without excessive unnecessary information overload.

2.2 Learning Sciences

The learning sciences focus on the study and design of learning environments [19] and the different contexts in which learning can take place [14]. The field also has a particular interest in the use of technology in education and how it enhances teaching and learning [28]. The learning sciences is a relatively young, interdisciplinary academic community that was formed in the 1970s [28] to "investigate learning and teaching in the real world" [30, p. 320]. It is derived from and closely related to the fields of psychology and cognitive science [21,28]. The interests and goals of the learning sciences include to further the understanding of how learning works and to "shape the ways that learning environments and resources are designed and used" [21, p. 329]. The field also focuses on studying learning as well as learning environments [19,28] and creating new designs for it . The involvement of academics in the learning sciences is diverse [21]. Academics from many fields (for example, computer science, anthropology, education and psychology [19,28]) in institutions worldwide provide their knowledge as a contribution to research on learning. As this field has evolved, many universities worldwide have started offering graduate programs in the learning sciences [30].

The focus areas of learning sciences research are:

1. How people learn: It investigates how people take in information and how knowledge is generated, acknowledging the fact that there are many perspectives on this in the many different fields that the learning sciences originated from [14]. Learning can take place in many different contexts in the real world [30]. One way of addressing this in this paper is through the interactive teaching approach followed.
2. Learning in the disciplines: This aspect is frequently addressed because researchers in the learning sciences ("learning scientists") focus their research on the application of the learning sciences in a wide range of fields, explicitly applying it in the context of their field [30]. This will not be the focus of this paper although it is important to note that all learning applies within context. Although this is not explicitly part of this paper, the content of the infographic was based on the context within which the students are studying as well as within the discipline of teaching business analysis and design.
3. Supporting learning and teaching: The focus is on the enhancement and encouragement of learning along with the "structuring and designing of all kinds of learning environments" [30, p. 322]. By adding the infographic on a familiar LMS platform for the students, ensured that the environment for the students were familiar.
4. Technology-enhanced learning and collaboration: Technological advances have a big influence on learning and teaching methods and enables interaction with others and learning through collaboration [30]. Research in the learning sciences focuses on the use of technology to encourage learning in different environments and looks at implementing these technologies into existing structures, creating new structures or changing existing structures [14].

Although one can argue that all of the above is relevant to this study, only points 1 and 4 are of particular interest, as it focusses specifically on how people learn and how they use technology to enhance their learning. Technology enhanced learning and collaboration was achieved by creating the interactive infographic using technology which can be used on various platforms (when you preview in iSpring, you can see how your design will look on various platforms).

The published iSpring interactive infographic was made available through the Learning Management System (LMS) Blackboard used by the University, by creating a Content Package (SCORM). This allowed the researchers to view which students opened the infographic as well as the duration spent on the infographic. At the moment one cannot see on which component the most time was spend; this will be a follow-up study. The LMS however, has a report function that affords students to view, on a dashboard, how they are performing against the outcomes to be reached by completing quizzes, assignments, and semester test questions, linked to the content and assessment on the use of the infographic. The lecturer's dashboard along with the Retention Centre in the LMS can assist in early identification of students that do not grasp the concepts of the basic theory of systems. The reporting function can be accessed by the lecturer to create additional course coverage, and performance reports to use the actionable data for additional support and improvement on the module towards student learning. In this regard, interactive infographic can be utilised as a remedial intervention.

To explore the notion of how students are learning, a survey was distributed to all students, the results of which will be analysed below.

2.3 Infographic

Dunlap and Lowenthal [13] refer to infographics as a combination of graphic data, visualised data, text, and illustrations that in the end should tell a story which can be used to support student learning.

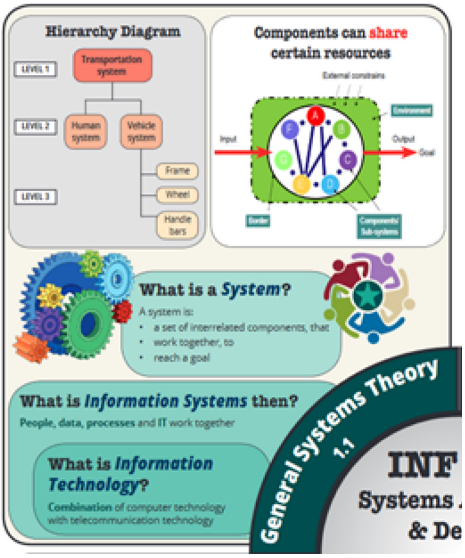


Fig. 1. Hot spot illustration 1

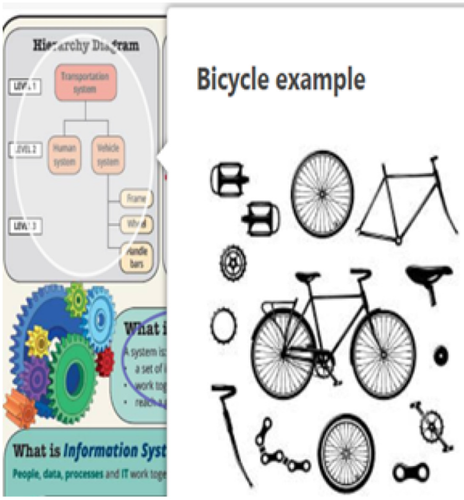


Fig. 2. Hot spot selection illustration

However, these infographics were all static, meaning they were a visual representation of the content that could be printed on one or two pages and, thus not interactive. There is something called a video infographic [3], yet this is purely for showing a video with data flowing. The interactive infographic allows students to stop and navigate back to previous content, selecting a section they wish to view again while listing to narrated voice explaining content. It can also accommodate embedded videos created by the researcher, to name but a few features.

In a paper published on a static infographic [33] the design was based on the 5 principles of the Gestalt theory [31,37]. Some of the constructive feedback obtained from the students were: "Could use more information, e.g. what a concrete and abstract system is."; "It would be great if we could access an online version where we could change the colours to our own preferences." This infographic was changed to become interactive. A static image of the infographic can be viewed in appendix A. The specific infographic introduces the basic systems theory to first-year students, divided into four sub-themes, which is covered in the infographic as "an introduction to systems concepts." Various software such as iSpring [15] and Doodly [12] were acquired to create the interactive infographic. iSpring Suite 9 was used to create interactions on the infographic, such as hot spots, quizzes and embedding YouTube videos. Doodly was used to create video explanations of certain concepts, which was published on a private channel on YouTube. Once a student clicks on a section he can get access to the video either through the link to the YouTube video that was added to the infographic or to the actual video that was embedded into the infographic. An example of how a typical hotspot works, is illustrated below. Once a student clicks on the hot spot (Fig. 1), a interaction screen opens up (Fig. 2), with the lecturer's

Fig. 3. Video hotspot illustration

voice explaining an example of the concept at hand. Students can scroll through the hotspot to see the explanations and examples. The students can select the hot spot as many times as they want to in order to understand the concept and examples. How the videos, created by the lecturer, were embedded to the YouTube link given, can be seen in Fig. 3.

A snippet of the video, which opens up, can be seen in Fig. 4. The lecturer's voice also guides the student through the video and explains the various concepts.

To illustrate how diagrams should be drawn, videos also show a hand drawing of the actual diagram, to mimic the idea of drawing the diagrams with the students (Fig. 5).

Is it necessary to note that when designing an infographic, the focus should be on the student and not the instructor [1]. For this specific reason, the learning sciences were used to investigate the effective use of the interactive infographic.

3 Research Method

A total number of 375 students are enrolled in a first-year business analysis and design course. The first study unit introduces students to systems and all the relevant components. Typically, a lecture is presented using various innovative techniques and students get access to the notes, or Power Point slides, through the LMS. In this study, the students also obtained access to the interactive infographic through the LMS. Students wrote a formative assessment in the form of a semester test. After the test, students were required to complete the survey, made available using a Google-Form. No marks were assigned for completion

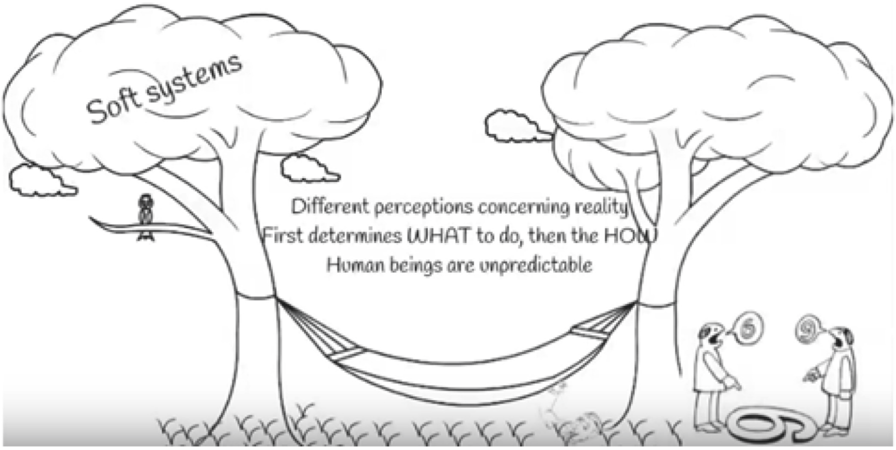


Fig. 4. Video snippet

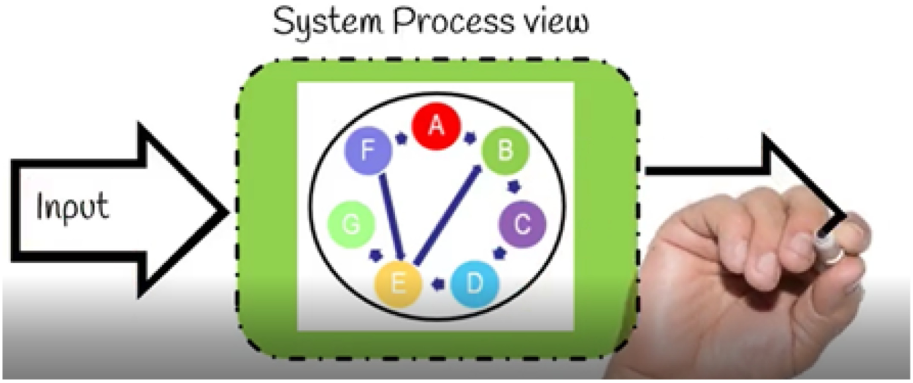


Fig. 5. Hand drawing diagram video snippet

of the survey and the survey was completely anonymous. One hundred and two (102) responses were obtained however, upon further investigation only 99 responses could be used. Tracking the access of the infographics on the LMS revealed a total number of 223 students worked through the infographic spending on average 26.7 min viewing the infographic.

The survey data was exported to Microsoft Excel and statistically analysed using IBM Statistics SPSS version 25. The methods used during the analysis of the data were frequency analysis per question; multiple response frequency analysis; Cross-sectional analysis, and graphical representations such as pie charts and bar charts.

4 Data Analysis and Discussion

To get some background information on the students, they provided the degree for which they are enrolled for. Most of the students were from the BCom (Informatics) (Information systems), BIT (Information technology) and BIS (Information Science) groups, this is not surprising as this course is required for all. A total of 48 respondents are studying BCom - Informatics Information Systems; 17 studies BIT - Information Technology; 15 studies BIS - Information Science; 10 studies BSc - Geoinformatics; four are studying BSc - Information and Knowledge Systems; with three studying BCom - Statistics and 2 studying BSc - Geography.

To determine whether accessing the infographic was possible, the student had to state whether it was easy to access, time-consuming or "I could not access it at all". 94 students said it was easy to access, 5 students said it was time consuming, and 3 said they could not access it at all. Students had to indicate whether they liked the fact that they could access the infographics through the LMS - 92% of the students responded yes. This confirmed that the students were more comfortable using the technology with which they are already familiar.

Students had to indicate how they accessed the infographic, either through a laptop, smart phone, Tablet or PC. From Fig. 6, it is clear that laptops were used the most to access the infographic. This was a surprising result as the lecturers carefully ensured the design of the infographic worked well on various devices, specifically on smart phones. Yet, students still preferred to access it through their laptops.

They had to indicate if they felt that the infographic worked well on the device they used. A total of 91.9% of the students felt the infographic worked well on the device they used. Performing Fisher's exact test on the way they accessed the infographic and whether the infographic worked well on the device they used, it showed that there was significant association between these two questions (p-value = 0.029). However, there was no clear indication where the differences could be.

Connecting the infographic to the learning science, in order to determine whether students felt the infographic did help them or could be applied in various teaching initiatives, students were asked whether they felt the infographics could be used to create new structures or change existing structures [14]. By asking students if they thought the infographic could successfully be used in the following contexts: presenting lectures, explaining study material, used in learning activities, showing examples, summarising content and studying for semester test, the results were rather clear from Fig. 7. Students see the value of using an alternative approach to understand the content of the module in various scenarios.

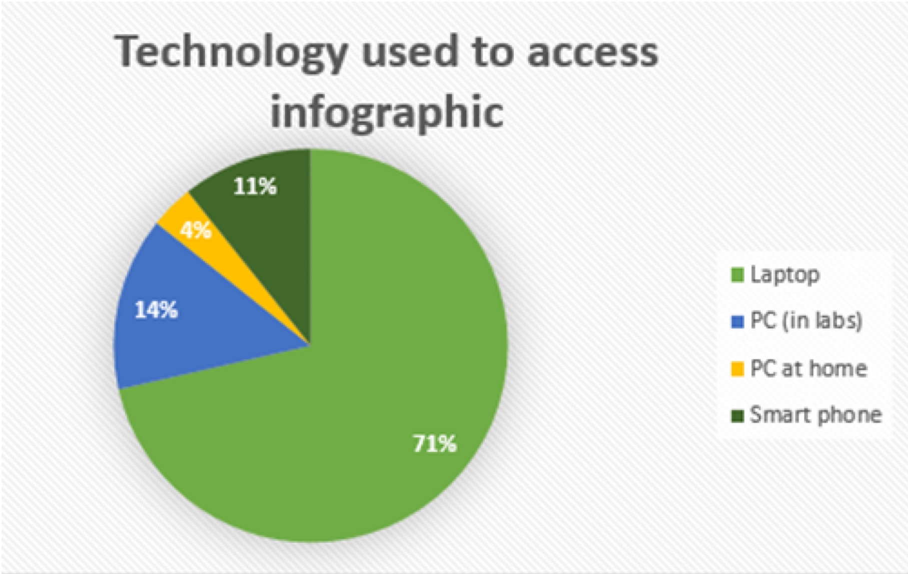


Fig. 6. Technology used to access

As part of the technology, numerous videos were created to explain concepts and to provide examples. Two videos were placed on YouTube, and two were embedded into the Infographic. One first had to determine what the students' perceptions of the videos were by asking whether they thought the videos were appropriate, relevant, fun to watch, and easy to access.

When asked whether they preferred to watch the videos opening YouTube or embedded into the infographic, 60% of the respondents preferred the embedded video, and 40% preferred the YouTube link. Performing Fisher's exact test between the students' perceptions of the video, and how they preferred to access the video, resulted in a significant association between these two questions (p-value = 0.002), with only one significant difference identified and that was the "neutral" response to the statement "video was fun to watch" and "linking through YouTube". More students (24) preferred accessing the video through YouTube and had a neutral response to the video as "being fun to watch" than was expected (16.2) under the assumption of no association.

Throughout the infographic there was a narrating voice, where one of the researchers explained the concepts and gave examples once the students clicked on that section, also referred to as hot spots. Asking the students if the narrations assisted them to understand the concepts better; whether they liked the informal presentations; whether the voice made them feel comfortable and whether the fact that it was their own lecturer's voice made them more willing to use the infographic. None of the students disagreed with the question "Lecturer's voice explaining concepts ...feel comfortable" and "Lecturer's voice ...made more willing to use the infographic" while 91% said their lecturer's voice explaining

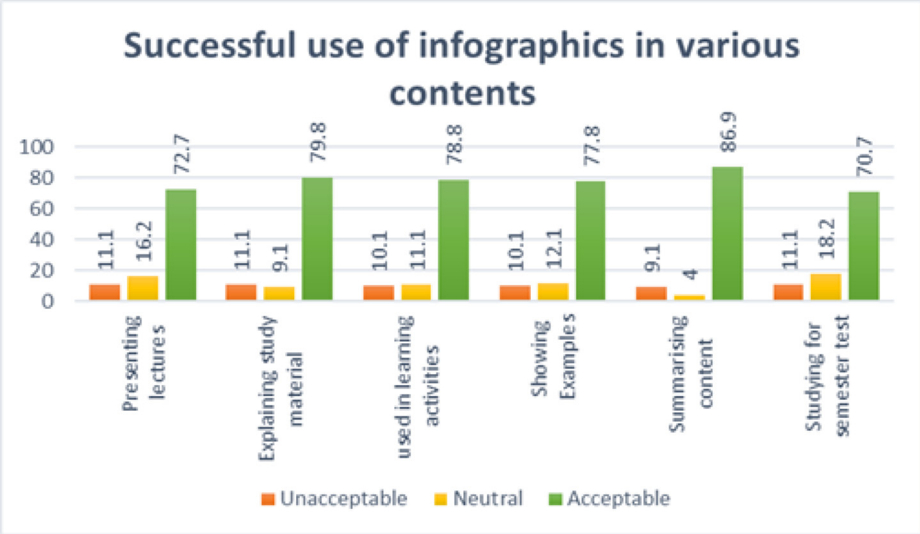


Fig. 7. Successful use of infographic

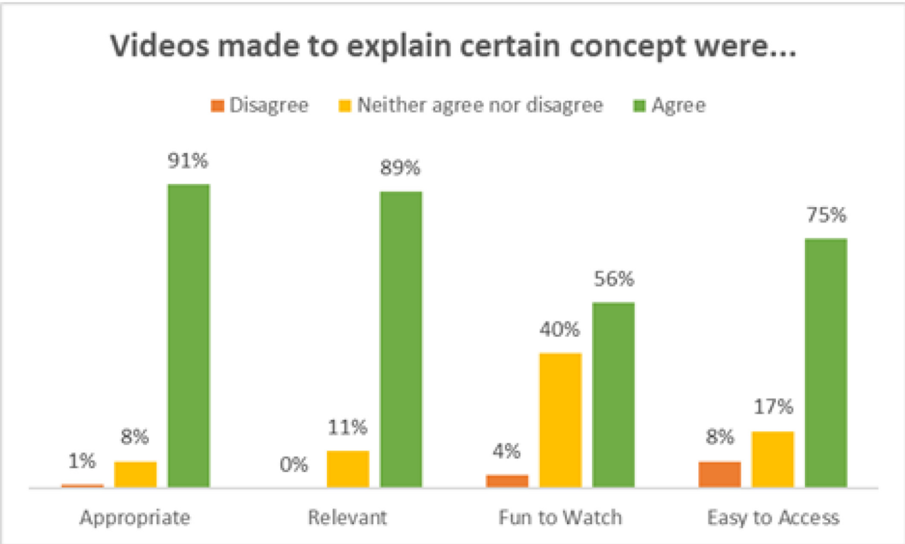


Fig. 8. Videos made were...

concepts made them feel comfortable and 83% agreed that the lecturer's voice made them more willing to use the infographic. Concerning "Informal narrations ...better understood the concepts", 85% agreed with this, with 15% feeling neutral. Asked whether the students liked the informal presentation of the concepts, 89% agreed and 10% felt neutral.

Students were asked if they felt the infographic improved their knowledge of the subject, and whether it was easy to use, useful and can be used to prepare for tests and exams. From Fig. 9, it is evident that students fully agreed with these.

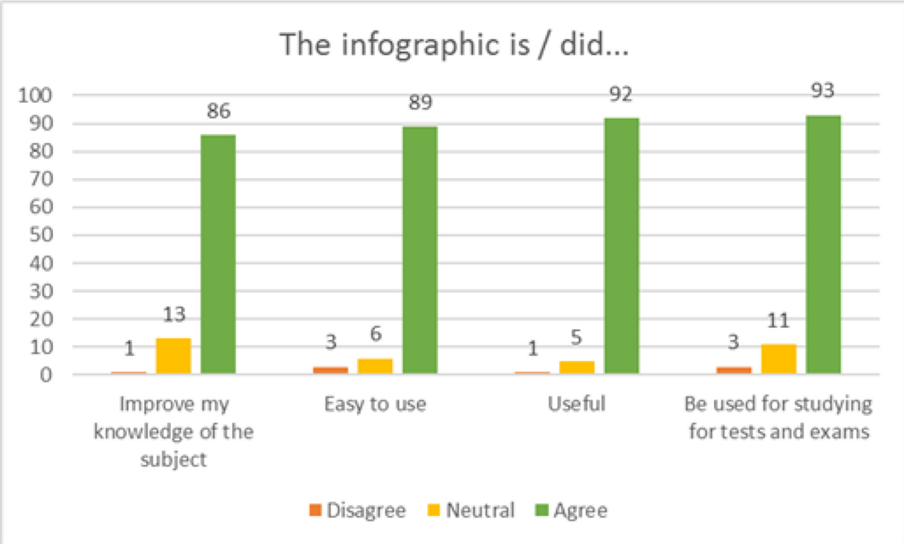


Fig. 9. Infographic is/did

Lastly, students were asked what additional features they would have liked to see in the infographic, some of the responses were: "Some practice questions"; "Everything was perfect, there's nothing to add"; "It was excellent with the voice recording to explain. The visuals helped a lot. I just wished the videos were embedded into the infographic instead of YouTube".

Some more constructive feedback, to improve the next version as well as follow-up infographics: "At the end of explaining the content can there maybe be a short quiz to see if you understand before the next slide opens, or at least obtain 60% of the questions correct"; "more examples" and "maybe some practical questions". Asking students if they had any additional comments: "Thank you very much for all the help, it really helped and was much easier to adapt more complicated content." and "Please design infographics for all the study units." Which emerged a few times. Thus, more infographics are being designed for this module.

In the end, the aim was to see whether the interactive infographic can be used, along with various types of technologies, to improve the student's knowledge and see if it could be used to determine how people learn. The students enjoyed the infographic but more so, they felt they could understand the content better and apply the knowledge in their semester test. Students further felt that the infographic could be used in explaining study material, learning activities and summarising the content. Another interesting find was that although students all have smartphones, the majority still preferred to open and work through the infographic using a laptop. This needs to be considered for future infographic designs and teaching initiatives.

5 Conclusion

This paper explored the notion of creating an interactive infographic to assist student in their knowledge acquisition. Looking though the learning science of "How people learn" and using "Technology-enhanced learning and collaboration", by designing an interactive infographic, students could engage with the content on their terms, using technology familiar to them, and engage with the content on a much deeper level, thus learning in the real world. Students could also work through the content in their own time and in the end, they felt they could understand the work better, and apply it in various learning activities. Supplementing the PowerPoint slides through the use of an interactive infographic made students engage far better with the content, thus creating a more in-depth learning experience.

However, there is a danger or warning with regards to be implementing technology as we cannot introduce technology into our learning without careful planning and deliberate outcome intentions for its use. Incorporating the infographic to the LMS of the university allowed the researchers to not only see if students accessed the infographic and the duration of time spent on the infographic, but also measured the student progress with the learning outcomes for the specific module using the goals Dashboard. This ensures that there is a way in which incorporating technology through the learning science focus into our teaching, can now be measured.

Acknowledgments. The first initial design of the static infographic was created by Mrs Nita Mennaga, and we would like to acknowledge and thank her for her contribution towards the first pilot design which was published in 2018 [33]. We would also like to acknowledge all the academics, who over the last 10 years assisted in creating the content for this module which formed the basis for the content of the infographic.

References

1. Alyahya, D.: Infographics as a learning tool in higher education: the design process and perceptions of an instructional designer. Int. J. Learn. Teach. Educ. Res. **18**(1), 1–15 (2019)
2. Ashraf, B.: Teaching the google-eyed Youtube generation. Educ. Train. **51**(5/6), 343–352 (2009)
3. Aydin, F., Aksüt, P., Demir, N.: The usability of infographics within the framework of learning outcomes containing sociascientific issues. Cumhuriyet Int. J. Educ. **8**(1), 154–171 (2019)

4. Belella, M., Dumbeck, D., Meluskey, M., Seetan, R.: A comparison study of online and traditional student withdrawal rates at a university in the Northeastern United States. J. Comput. Sci. Coll. **34**(1), 93–103 (2018)
5. Bender, T.: Discussion-based Online Teaching to Enhance Student Learning: Theory, Practice, and Assessment, 2nd edn., Stylus Publishing, Virginia (2012)
6. Bonk, C., Graham, C.: The handbook of blended learning: global perspectives. Local designs. Pfeiffer, San Francisco, CA (2006)
7. Bradshaw, M., Porter, S.: Infographics: a new tool for the nursing classroom. Nurse Educ. Technol. Corner **42**(2), 57–59 (2017)
8. Chicca, J., Shellenbarger, T.: Connecting with generation Z: approaches in nursing education. Teach. Learn. Nurs. **13**(3), 180–184 (2018)
9. Davidson, N., Major, C., Michaelsen, L.: Small-group learning in higher education - cooperative, collaborative, problem-based and team-based learning: An introduction by guest editors. J. Excell. College Teach. **25**(3/4), 1–6 (2014)
10. Davidson, R.: Using infographics in the science classroom. Sci. Teach. **81**(3), 34–39 (2014). https://doi.org/10.2505/4/tst14_081_03_34
11. Davies, R.S., Dean, D.L., Ball, N.: Flipping the classroom and instructional technology integration in a college-level information systems spreadsheet course. Educ. Technol. Res. Dev. **61**(4), 563–580 (2013). https://doi.org/10.1007/s11423-013-9305-6
12. Doodly: (2019). www.doodly.com/
13. Dunlap, J., Lowenthal, P.: Getting graphic about infogrpahics: design lessons learned from popular infographics. J. Vis. Liter. **35**(1), 42–59 (2016). https://doi.org/10.1080/1051144X.2016.1205832
14. Hoadley, C., Van Haneghan, J.P.: The Learning Sciences: Where They Came from and What It Means for Instructional Designers, 3 edn., Book Section 6, pp. 53–63. Pearson, New York (2012)
15. ISpring (2019). www.ispringsolutions.com/ispring-suite
16. Jayashanka, R., Hewagamage, K., Hettiarachchi, E.: Improving blended learning in higher education through the synergy between learning design and learning analytics. In: 2018 IEEE 18th International Conference on Advanced Learning Technologies (ICALT) (2018). https://ieeexplore-ieee-org.uplib.idm.oclc.org/document/8433500
17. Kalantzis, M.: Changng subjectivities, new learning. Pedagog. Int. J. **1**(1), 7–12 (2006)
18. Lai, K.W., Khaddage, F., Knezek, G.: Blending student technology experiences in formal and informal learning. J. Comput. Assist. Learn. **29**(5), 414–425 (2013). https://doi.org/10.1111/jcal.12030
19. Margulieux, L., Dorn, B., Searle, K.: Learning Sciences for Computing Education. Cambridge University Press, Cambridge (2019)
20. Mohr, K.A., Mohr, E.S.: Understanding generation Z students to promote a contemporary learning environment. J. Empower. Teach. Excell. **1**(1), 9 (2017)
21. Nathan, M., Alibali, W.: Learning sciences. Wiley Interdiscipl. Rev. Cogn. Sci. **1**(3), 329–345 (2010)
22. Nichols, T., Wright, M.: Generational differences: understanding and exploring generation Z. 2017–2018 OFFICERS President President-Elect, p. 177 (2018)
23. Nja, C., Umali, C.B., Asuquo, E., Orim, R.: The influence of learning styles on academic performance among science education undergraduates at the University of Calabar. Educ. Res. Rev. **14**(17), 618–624 (2019)
24. Puttnam, L.: Creative learning through technology (2012)

25. Rosenberg, M.: Knowledge Management and Learning: Perfect Together, 3rd edn, Book Section, vol. 16. Pearson, Boston (2012)
26. Rothman, D.: A tsunami of learners called generation Z. Public Saf. Online J. **1**(1) (2016). https://mdle.net/Journal/A_Tsunami_of_Learners_Called_Generation_Z.pdf
27. Saunders, D., Horrel, A., Murray, A.: Infographics for students assessment: more than meets the eye. BJSM Online first (2017). http://bjsm.bmj.com/
28. Sawyer, R.: The Cambridge Handbook of the Learning Sciences. Cambridge University Press, Cambridge (2005)
29. Shatto, B., Erwin, K.: Moving on from Millennials: preparing for generation Z. J. Contin. Educ. Nurs. **47**(6), 253–254 (2016). https://doi.org/10.3928/00220124-20160518-05
30. Sommerhoff, D., Szameitat, A., Vogel, F., Chernikova, O., Kristina Loderer, K., Fischer, F.: What do we teach when we teach the learning sciences? a document analysis of 75 graduate program. J. Learn. Sci. **27**(2), 319–351 (2018)
31. Staff: Gestalt theory in typography & design principles. HOW Mag. 11 May 2015. http://www.howdesign.com/resources-education/online-design-courses-education/gestalt-theory-typography-design-principles/
32. Steyn, R., Millard, S., Jordaan, J.: The use of a learning management system to facilitate student-driven content design: an experiment. In: Huang, T.-C., Lau, R., Huang, Y.-M., Spaniol, M., Yuen, C.-H. (eds.) SETE 2017. LNCS, vol. 10676, pp. 75–94. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-71084-6_10
33. Steyn, R., Botha, A., Mennega, N.: Is a picture truly worth a thousand words? infographics for undergraduate teaching. In: Hao, T., Chen, W., Xie, H., Nadee, W., Lau, R. (eds.) SETE 2018. LNCS, vol. 11284, pp. 69–78. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-03580-8_8
34. Sudakov, I., Bellsky, T., Usenyuk, S., Polyakova, V.: Infographics and mathematics: a mechanism for effective learning in the classroom. Probl. Resour. Iss. Math. Undergrad. Stud. **26**(2), 158–167 (2016)
35. Vanichvasin, P.: Enhancing the quality of learning through the use of infographics as visual communication tool and learning tool. In: ICQA 2013 International Conference on QA Culture: Cooperation or Competition, pp. 135–142 (2013)
36. Vikhrova, O.: On some generation Z teaching techniques and methods in higher education. Information **20**(9A), 6313–6324 (2017)
37. Wang, L.: Gestalt principles for information design. HOW, 7 April 2017, http://www.howdesign.com/design-business/design-thinking/gestalt-principles-for-information-design/
38. Williams, F.: Diversity, thinking styles, and infographics. In: 12th International Conference of Women Engineers and Scientists (2002)
39. Williams, S.: Interactive learning: The pros and cons (2019). https://jimmyesl.com/interactive-learning/
40. Wilson, R.: In-class-online hybrid methods of teaching planning theory: assessing impacts on discussion and learning. J. Plann. Educ. Res. **28**, 237–246 (2008)
41. Woods, R., Baker, J., Hopper, D.: Hybrid structures: Faculty use and perception of web-based courseware as a supplement to face-to-face instruction. Internet High. Educ. **7**, 281–297 (2004)
42. Yakovleva, N.O., Yakovlev, E.V.: Interactive teaching methods in contemporary higher education. Pac. Sci. Rev. **16**(2), 75–80 (2014). https://doi.org/10.1016/j.pscr.2014.08.016
43. Yildirim, S.: Infographics for educational purposes: their structure, properties and reader approaches. Turkish Online J. Educ. Technol. **15**(3), 98–110 (2016)

Influence of External Factors on the Attitude of Students Towards Arduino Micro Development Boards

Malie Zeeman(✉) and Romeo Botes

School of Computer Science and Information Systems, North-West University, Potchefstroom, South Africa
malie.zeeman@nwu.ac.za

Abstract. The fourth industrial revolution has a significant impact on the professional skills set of employees. Creativity, critical thinking and problem-solving skills as well as communication and effective teamwork are some of the required skills. The rapid pace of change in technology and consequently, new required skills urge educational institutions to create an inspiring and suitable learning environment for the 21st-century student.

As such, a project-based learning approach was introduced at the School of Computer Science and Information Systems at a higher education institution in South Africa to better prepare students for the current technologically advanced workplace. Students were exposed to hands-on hardware programming using Arduino micro development boards. With no previous experience in the field of hardware programming, students were introduced briefly to some basic concepts and thereafter, they had to develop a solution to solve a specific problem. Apart from hardware programming, skills such as communication, effective teamwork and innovation were required to solve the problem. For Arduino micro development boards to be successfully incorporated in class, it is necessary to determine the student's acceptance of such technology as a teaching tool towards an Industry 4.0 related qualification. Therefore, the attitude of students towards using Arduino technology in class was evaluated using a questionnaire on completion of the project.

The focus of this paper is on measuring the external factors that influence the student's attitude towards the use of the Arduino micro development boards as a teaching tool in class. For this purpose, the Technology Acceptance Model was used. Statistical analysis was done on collected data to derive relations between various external factors and the attitude of students towards the use of the Arduino micro development boards in class. The results of this paper show that external factors such as enjoyment, subjective norms, and focused immersion have a significant influence on the attitude of students towards using Arduino micro development boards as a teaching tool in class. This serves as evidence that the use of Arduino micro development boards as a teaching tool could be implemented successfully in class in the South African context. However, some of the external factors that were evaluated contributed less to the students' attitude.

G. Wells et al. (Eds.): SACLA 2020, CCIS 1518, pp. 187–202, 2021.
https://doi.org/10.1007/978-3-030-92858-2_13

Keywords: Arduino · Fourth industrial revolution · Computer science · Technology acceptance model

1 Introduction

Logical and abstract thinking skills are required to survive in the current complex and challenging digital world [30]. The dawn of the fourth industrial revolution (IR) at the turn of the century introduced big data, interconnectivity and cyber-physical systems as part of the business world and many other spheres of life [14,22]. People skilled in innovative, logical and abstract thinking skills within the field of Computer Science and Information Systems are in high demand to plan, build and manage these cyber-physical systems [4].

Although the learning environment in higher education institutions is changing, education systems are not adapting fast enough in revising pedagogical methods and course material to prepare students for the world in which they will enter as part of the workforce [9]. Gleason [14] urge higher education institutions to realise the value of project-based learning as valuable preparation of students towards readiness for the digital workplace. The author argues that teamwork and project-based learning is required to provide students with social and emotional skills imperative towards twenty-first-century success. Jamieson and Herdtner [18] affirm this and state that project-based learning allows students to "learn on the fly" as they solve project-based problems.

In the fields of computer science and engineering, project-based learning involves prototyping, which can be complex and expensive [7]. The introduction of Arduino as an open and freely available hardware platform in 2005 made it possible for prototyping to be less expensive and manageable.

The focus of this paper is on the use of Arduino as part of a project-based learning experience in a Computer Science class. The attitude of students towards using this platform as a teaching tool in Computer Science is investigated with specific reference to external factors that influenced the students' attitude towards using the Arduino hardware platform as a teaching tool. As such, the Technology Acceptance Model (TAM) was used as the foundation of the empirical part of the study. The TAM is a validated and widely used measuring instrument for evaluating the attitude towards technology being used [17].

This paper continues by including a brief overview of related work regarding the fourth industrial revolution, an introduction to Arduino and shields, a brief discussion about the technology acceptance model with external factors in Sect. 2. The research design of this study is highlighted in Sect. 3, followed by Sect. 4, the discussion of analysed results, while Sect. 5 concludes this paper.

2 Related Work

The world has experienced several industrial revolutions (IRs) since the 1980s [31]. Figure 1 shows that the first IR was initiated by mechanisation, and the second IR was sparked by the use of electrical energy. At the same time, the

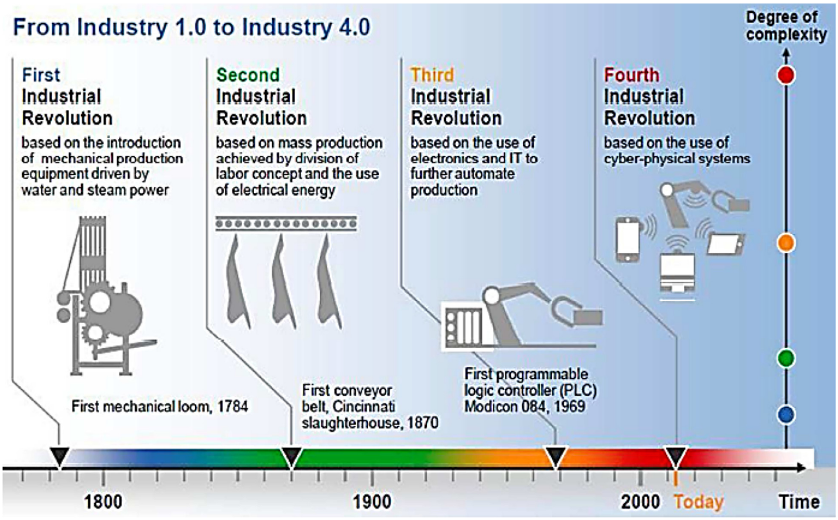


Fig. 1. Industrial revolutions through the centuries.

Table 1. 21st-century skills adapted from [5]

Categories	Descriptions
Ways of thinking	Creativity, critical thinking, problem-solving, decision-making and learning to learn
Ways of working	Communication and collaboration (teamwork)
Tools for working	Information and communication technology (ICT) and information literacy
Skills for living in the world	Citizenship (local and global), life and career, and personal and social responsibility

third IR came about with the use of electronics and automation [4]. Today we are part of the fourth IR, which is based on connectivity, big data and cyber-physical systems (CPSs) [14]. Cyber-physical systems are embedded computers and networks that manage and control physical processes while improving their intelligence [22].

Each IR had an impact on all levels of society, including the job market and education [31]. New jobs emerging within the fourth IR require a specific set of skills [12].

Table 1 shows the required 21st-century skills set, organised into four broad categories by an organisation known as Assessment & Teaching of 21^{st} Century Skills [5].

In their study on education for Industry 4.0, Benešová and Tupa [4] urge HEIs to take note of new IT jobs emerging such as information specialists, PLC programmers, Robot programmers, cybersecurity specialists and data analysts.

Furthermore, Industry 4.0 needs a workforce with an attitude of professionalism, experience in multidisciplinary teamwork, and outstanding communication skills [20].

Benešová and Tupa [4] state that curricula and disciplines within higher education need to change to meet the demand of Industry 4.0. A shift from universities being academic institutions that educate people to institutions that inspire innovation is required [1].

A multidisciplinary approach should be followed in Computer Science and Information Systems programs in order to equip students with skills to enable modern businesses to implement Industry 4.0 technologies [1].

Since the introduction of Arduino MDBs, several studies were conducted to investigate the use of this technology in the field of science and technology as a teaching tool [7,18]. Jamieson and Herdtner [18] used Arduino for the first time in 2011 as a prototyping board at the Department of Computer Science and Engineering at the University of Miami. Knowledge and skills gained from this project encouraged students to extend the range of prototyping boards to Raspberry Pi and BeagleBoneBlack boards. These easy to use platforms enhanced the educational experience of students in terms of self-motivation, confidence and system design abilities.

Buechley et al. [6] conducted various projects using the Arduino platform. Feedback from students varied from negative to extremely motivated. Similarly, Ling and Wah [23] investigated Arduino as a means of including hands-on computing activities towards a better understanding of Science, Technology, Engineering and Mathematics (STEM). For this purpose, students who enrolled for an introductory course in computing were exposed to hardware programming using the Arduino Uno micro development board. Results obtained from this project depicted that students enjoy working with Arduino and that interaction with an electronic device such as Arduino improves their programming skills. Furthermore, Arduino as teaching tool exposes students to future technologies and 21st-century skills such as critical thinking, collaboration and communication. Therefore, the researchers concluded that Arduino is highly recommended as a STEM teaching tool [23].

2.1 Arduino

Arduino is a microprocessor-based electronic prototyping platform, as shown in Fig. 2. The platform contains software features and hardware components [7]. The Arduino platform is used widely in the field of Electronics and Robotics as a prototyping platform for small scale projects and also as a teaching tool towards obtaining fourth IR technologies and skills [12]. Furthermore, Arduino is used effectively for collaboration and the integration of STEM fields of studies which includes programming, science, mathematics and technology.

Various Arduino boards came into existence with the most popular being the Arduino Uno that includes an ATmega328 microprocessor and a USB to serial chip for easy programming and communications [18]. Arduino MDBs can operate on various operating systems and are programmed with the Arduino

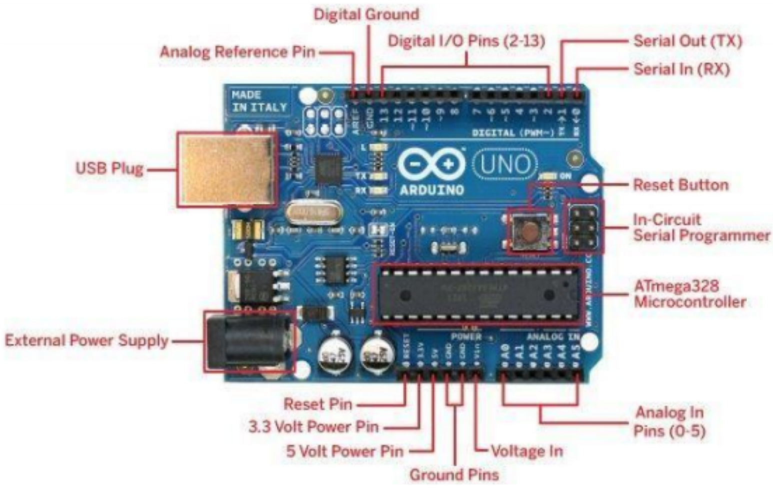


Fig. 2. The layout of an Arduino Uno MDB [2]

Integrated Development Environment (IDE). The IDE contains a selection of prebuilt libraries and example code used to learn how to code Arduino. These positive features have made Arduino be used extensively in education as an introduction to programming and electronics [7,23].

However, limited research is available on the acceptance of Arduino as a teaching tool in higher education programming from a student's perspective.

A shield is a board that contains the same form-factor as the standard Arduino. The shield can be plugged in on the top of an Arduino MDB, as shown in Fig. 3, to extend the capabilities of the robot [2]. A suitable shield for a project depends is selected based on the abilities required to perform the task at hand, such as 2D or 3D printing [19] or GPS capabilities [32]. For this study, the JellyBeanBot shield shown in Fig. 3 was used since the task posed to students included moving along a track while avoiding obstacles.

2.2 Technology Acceptance Model

The technology acceptance model (TAM) was inspired by the work of Fishbein and Ajzen [13] in the field of theory of reasoned action (TRA) [10,17,21]. The theory states that the intention of an individual to behave in a particular way is influenced by attitude and social pressure towards the intended behaviour [21]. The TAM was developed by Fred Davis, who established a conceptual framework for users' motivation to use a system based on attitude and intention to use [10].

Internal Factors. Attitude can be described as "one's positive or negative feelings about performing a behaviour (for example, using technology)" [34]. Two distinguishing factors or constructs that influence a user's attitude were identified as part of the TAM, namely perceived ease of use (PEOU) and perceived

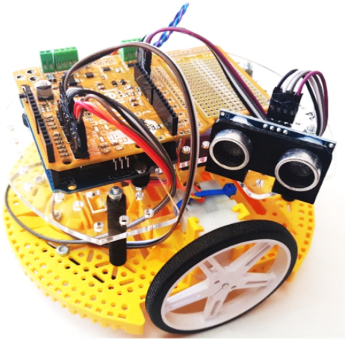


Fig. 3. Arduino Uno MDB with JellyBeanBot shield

usefulness (PU). The original TAM, shown in Fig. 4, shows the relationships between these two factors and attitude as components that influence user motivation to use technology. Users' attitudes towards using technology determine, in turn, the actual use the technology. The users' attitude towards the use of technology is influenced by the users' perceived usefulness and ease of use. Davis's model also includes external variables, X1, X2 and X3, which refer to factors that influence the perception of the usefulness of the system and ease of use [11].

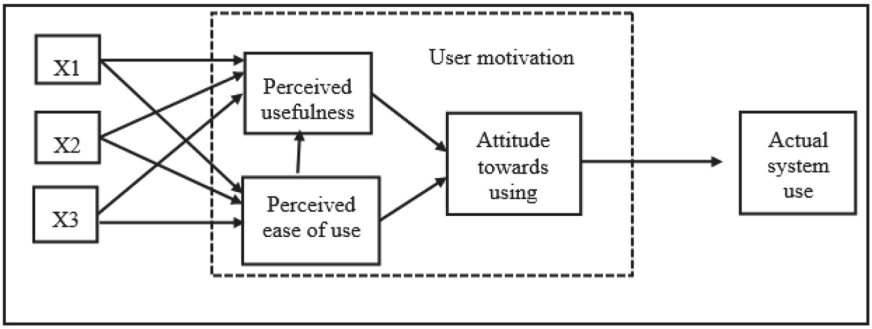


Fig. 4. Original TAM [13]

Since the introduction of the original TAM, several versions of the TAM were developed, focusing on the influence of a variety of external factors on the attitude of the user towards the acceptance of various technologies [17,35,37]. The theoretical reasoning on the construction of TAM2 is based on a match between the importance of technology in the workplace and how useful the technology is to get the work done [36]. TAM2 implements two categories - social influence and cognitive instrumental processes - as external processes that influence behavioural intention with individual differences, system characteristics, social influence, and facilitating conditions as constructs within each category [35]. Venkatesh and Bala [35] extended TAM2 by extending the list of constructs while focussing on social influence (i.e., compliance, identification, and internalisation) and several traits and emotions (i.e., computer self-efficacy, computer playfulness, computer anxiety and enjoyment).

This study aimed to determine whether students found the Arduino MDBs to be useful learning instruments in the Computer Science class. For this purpose, external factors were adapted from both TAM2 and TAM3 towards evaluating social influence (subjective norms), cognitive processes (focused immersion) due to the context of programming and problem-solving skills and emotions (experienced enjoyment).

External Factors. Subjective norm (SN) is defined as a "person's perception that most people who are important to him think he should or should not perform the behaviour" [13]. Literature reveals that social aspects play a significant role in intended behaviour in the acceptance of technology [15,16]. People who are a novice in using technology will be influenced to a larger extent to behave in a specific way by people close to them, than people who have experience in using the technology [36].

Focused immersion is associated with a state of flow, which refers to people who become completely absorbed in what they are doing [24]. Basawapatna et al. [3] found that learning computational thinking skills takes place more effectively when students are totally engaged in the task at hand. Students can perform challenging tasks within a zone of proximal flow. In a flow state of mind, students are focussed and immersed in what they are doing [27]. Tan et al. [33] state that problem-solving requires intrinsic motivation and deep thinking skills. Intrinsic motivation to stay involved and enjoyment are important results of a flow experience. The circumstances that initiate a flow state of mind include just-manageable tasks and continue with feedback on actions performed, which results in adjusting action based on the feedback. A series of graded challenges accomplished causes the person to experience progression, enjoyment and motivation to continue to try to achieve the next task. In a state of flow, intense concentration is exercised, and tasks normally regarded as too difficult to accomplish are often completed [33].

3 Research Design

3.1 Background Information

The purpose of the project was to introduce students to fourth IR technologies and allow them to explore hardware programming and robot technology hands-on. Furthermore, the use of technology in class was investigated to establish a student's acceptance of technology as a teaching tool in class.

The project involved a group of final year Computer Science students enrolled for an Expert System course, which compulsory as part of their Computer Science degree modules. The contact time was a three and a half hours session per week for ten weeks. Since the students did not know electronics and hardware programming, they were introduced to basic concepts during the first five weeks of the course. An experimentation Arduino shield per group of 6 students was used for this purpose. Students could choose their team members each week for these primary tasks. Each introductory session included: a brief information session on the topic for the week as well as hands-on implementation of the knowledge. This was done in the form of a simple task that the students had to do in a group.

During week 5, students were divided into 40 groups of 3–4 students per group. The lecturer allocated members to the groups based on the students' performance during the previous introductory tasks. The compilation of the new group was done to ensure that students who did not perform well previously be grouped with at least one student who performed well to share knowledge and skills with the other group members. A robot (shown in Fig. 3) was provided to each group during class time once a week, which they had to program. The students were introduced to the playing field and received one task per week to add to the list of tasks that the robot had to perform for final evaluation during week 10. The method of instruction was informal, with no slides or other formal study material. Students had to use the Internet to obtain information and had to brainstorm as groups to compile solutions to problems they encountered. The lecturer only provided hints and selective guidance to steer the students in the right direction if required.

On completion of the project during week ten students were requested to complete a questionnaire derived from the validated TAM questionnaire and slightly adapted for this study. The questionnaire was adapted to evaluate factors that influence the students' attitude towards the use of Arduino as a teaching tool in class.

3.2 Sampling Method

The target population for this study included Computer Science students registered at public South African higher education institutions (HEIs). The sampling frame includes 26 HEIs. From this sampling frame, one campus of an HEI located in the Gauteng province was selected using judgement sampling. In continuation, a convenience sample of 120 Computer Science students was drawn from the original sampling frame.

3.3 Measurement Instrument and Data Collection

The data collection for this study was done by making use of a self-administered questionnaire - this questionnaire comprised two sections. Section A gathered the participants' demographical information. In contrast, Section B was adapted from existing scales by Fishbein and Ajzen [13], Saadé and Bahli [29] and

Table 2. Reliability coefficients, descriptive statistics and correlation matrix

Constructs	Cronbach alpha	Mean	Standard deviation	A	BI	PU	PEOU	SN	FI
A	.870	5.26	.904						
BI	.824	5.26	.846	.830**					
PU	.907	5.14	.928	.843**	.785**				
PEOU	.879	4.78	1.011	.697**	.663**	.707**			
SN	*	4.90	1.106	.655**	.653**	.748**	.577**		
FI	*	4.41	1.287	.622**	.546**	.528**	.498**	.430**	
EE	.781	5.03	1.065	.741**	.702**	.686**	.585**	.611**	.510**

**Correlation is significant at the 0.01 level (2-tailed)
*Minimum of 3 items required to calculate
A - Attitude/BI - Behavioural intention/PU - Perceived usefulness/
PEOU - Perceived ease of use/SN - Subjective norms/
FI - Focused immersion/EE - Experienced enjoyment

Venkatesh and Davis [36] and gathered data about the subjective norm, focused immersion, experienced enjoyment, perceived usefulness, perceived ease of use, attitude and behavioural intention of the participants towards the use of Arduino MDBs in a Computer Science class. A six-point Likert scale, ranging from strongly disagree (1) to strongly agree (6), was used to measure these scaled items.

4 Results and Discussion

The captured data was analysed using the Statistical Package for Social Sciences (IBM SPSS) Version 25. The methods used to analyse the data comprised Pearson's product-moment correlation analysis, reliability and validity measures, descriptive statistics as well as regression analysis.

From the 120 students who participated in the study, using Arduino Uno MDBs in an exit level module, during class sessions, 100 completed questionnaires were returned, calculating a response rate of 83%. All of the specified age groups were represented in the sample. The majority of the sample was 21 years of age (23.0%), closely followed by those 20 years of age (20.0%). The sample consisted of more males (60%) than females (40%), with the majority of participants were in their third year of study (58%).

Descriptive statistics comprising measures of location and variability were computed, as well as Pearson's product-moment correlation coefficients for each pair of constructs. Correlation analysis was conducted to assess nomological validity. Table 2 presents the means, standard deviations and correlation coefficients.

As shown in Table 2, the Cronbach alpha values computed for each of the constructs exceeded the recommended level of 0.7 [28] and ranging from 0.781 to 0.907, indicating acceptable internal consistency reliability as all the values, which was done to assess the internal consistency reliability of the scale. Concerning Computer Science, students' acceptance of Arduino MDBs as a teaching tool

Table 3. Regression analysis of PU and PEOU as a predictor of A

Model	Unstandardised coefficients		Standardised coefficients		
	B	Std. error	Beta	t	Sig.
A (Constant)	.851	.276		3.086	.003
PU	.672	.074	.690	9.120	.000
PEOU	.199	.071	.212	2.808	.006

R = .856, Adjusted R-squared = .732 (73.2%),
Std. error of the estimate = .47266, p = .000
A - Attitude/PU - Perceived usefulness/
PEOU - Perceived ease of use

Table 4. Regression analysis of PEOU as a predictor of PU

Model	Unstandardised coefficients		Standardised coefficients		
	B	Std. error	Beta	t	Sig.
PU (Constant)	1.805	.331		5.450	.000
PEOU	.693	.068	.720	10.261	.000

R = .720, Adjusted R-squared = .518 (51.8%),
Std. error of the estimate = .64775, p = .000
PU - Perceived usefulness/PEOU - Perceived ease of use

in class, means above 4.4 were computed on the dimensions of attitude (mean = 5.26), behavioural intention (mean = 5.26), perceived usefulness (mean = 5.14), perceived ease of use (mean = 4.78), subjective norms (mean = 4.90), focused immersion (mean = 4.41) and experienced enjoyment (mean = 5.03). Correlation analysis using Pearson's Product-Moment coefficient was conducted. The computed correlation coefficients presented in Table 3 indicate that the relationships between each of the pairs of constructs are positive and statistically significant ($p \leq 0.01$), suggesting nomological validity.

The following section illustrates the results of the regression analysis conducted. Multiple linear regression was calculated to predict attitude based on perceived usefulness and perceived ease of use. As presented in Table 3, a significant regression equation was found ($p < .000$), with an R^2 of 0.725.

Perceived usefulness and perceived ease of use show a strong relationship towards attitude with R = 0.855, which infers that perceived usefulness and perceived ease of use have a significant influence on the students' attitude towards Arduino MDBs, explaining 72.5% variance.

In Table 4, the standardised coefficients (β) indicate that perceived usefulness (β = 0.700) has a larger impact than perceived ease of use (β = 0.202) on students' attitude towards the use of Arduino MDBs as a teaching tool. The regression analysis of perceived ease of use as a predictor of perceived usefulness delivered a significant regression equation ($p < .000$), with an R^2 value of 0.495, explaining 49.5% of variance. Perceived ease of use has a significant influence on the students' perceived usefulness (β = 0.707). As depicted in Table 5, a significant regression equation was found ($p < .000$) in the analysis testing attitude

Table 5. Regression analysis of A as a predictor of BI

Model	Unstandardised coefficients		Standardised coefficients		
	B	Std. error	Beta	t	Sig.
BI (Constant)	1.176	.281		4.177	.000
A	.776	.053	.830	14.720	.000

R = .830, Adjusted R-squared = .685 (68.5%),
Std. error of the estimate = .47432, p = .000
A - Attitude/BI - Behavioural intention

Table 6. Regression analysis of SN, FI and EE as a predictor of PEOU

Model	Unstandardised coefficients		Standardised coefficients		
	B	Std. error	Beta	t	Sig.
PEOU (Constant)	1.275	.402		3.171	.002
SN	.283	.088	.309	3.199	.002
FI	.173	.070	.220	2.474	.015
EE	.269	.096	.283	2.787	.006

R = .678, Adjusted R-squared = .437 (43.7%),
Std. error of the estimate = .758688, p = .000
A - Attitude/BI - Behavioural intention/PU - Perceived usefulness/
PEOU - Perceived ease of use/SN - Subjective norms/
FI - Focused immersion/EE - Experienced enjoyment

as a predictor of behavioural intention. The model explains 68.5% (see adjusted R-squared) of the variance in this relationship.

The following tables pertain to the regression analysis of the external factors that influenced Computer Science students' acceptance towards Arduino MDBs. According to the results presented in Table 6, the regression analysis of subjective norms, focused immersion and experienced enjoyment as a predictor of perceived ease of use delivered a significant regression equation ($p < .000$), with an R^2 value of 0.437, explaining 43.7% of variance. Subjective norms have the most significant influence ($\beta = 0.309$) followed by experienced enjoyment ($\beta = 0.283$) and focused immersion ($\beta = 0.220$) on the students' perceived ease of use. According to the results presented in Table 6, the regression analysis of subjective norms, focused immersion and experienced enjoyment as a predictor of perceived ease of use delivered a significant regression equation ($p < .000$), with an R^2 value of 0.437, explaining 43.7% of variance. Subjective norms have the most significant influence ($\beta = 0.309$) followed by experienced enjoyment ($\beta = 0.283$) and focused immersion ($\beta = 0.220$) on the students' perceived ease of use.

As presented in Table 7, the regression analysis of subjective norms, focused immersion and experienced enjoyment as a predictor of perceived usefulness delivered a significant regression equation ($p < .000$), with an R^2 value of 0.651, explaining 65.1% of variance. Subjective norms again have the most significant influence ($\beta = 0.495$) followed by experienced enjoyment ($\beta = 0.301$) and focused immersion ($\beta = 0.162$) on the students' perceived usefulness.

Table 7. Regression analysis of SN, FI and EE as a predictor of PU

Model	Unstandardised coefficients		Standardised coefficients		
	B	Std. error	Beta	t	Sig.
PU (Constant)	1.271	.291		4.374	.000
EE	.415	.064	.495	6.496	.000
FI	.117	.051	.162	2.310	.022
SN	.262	.070	.301	3.762	.000

R = .813, Adjusted R-squared = .651 (65.1%),
Std. error of the estimate = .5481923, p = .000
A - Attitude/BI - Behavioural intention/PU - Perceived usefulness/
PEOU - Perceived ease of use/SN - Subjective norms/
FI - Focused immersion/EE - Experienced enjoyment

Table 8. Regression analysis of SN, FI and EE as a predictor of A

Model	Unstandardised coefficients		Standardised coefficients		
	B	Std. error	Beta	t	Sig.
A (Constant)	1.44	.278		5.258	.000
SN	.220	.061	.268	3.585	.001
FI	.202	.048	.288	4.172	.000
EE	.365	.067	.430	5.466	.000

R = .820, Adjusted R-squared = .663 (66.3%),
Std. error of the estimate = .5251971, p = .000
A - Attitude/BI - Behavioural intention/PU - Perceived usefulness/
PEOU - Perceived ease of use/SN - Subjective norms/
FI - Focused immersion/EE - Experienced enjoyment

According to the results depicted in Table 8, the regression analysis of subjective norms, focused immersion and experienced enjoyment as a predictor of attitude delivered a significant regression equation ($p < .000$), with an R^2 value of 0.663, explaining 66.3% of variance. In this case, experienced enjoyment has the most significant influence ($\beta = 0.430$) on students' attitude, followed by focused immersion ($\beta = 0.288$). Subjective norms have the least significant influence ($\beta = 0.268$) on students' attitude towards the use of Arduino MDBs as a teaching tool in class, contradicting the previous two regression analyses (perceived usefulness and perceived ease of use).

The dependent relationships between external variables and internal variables for the acceptance of using Arduino MDBs in a Computer Science class, as depicted in the previous tables are shown in Fig. 5.

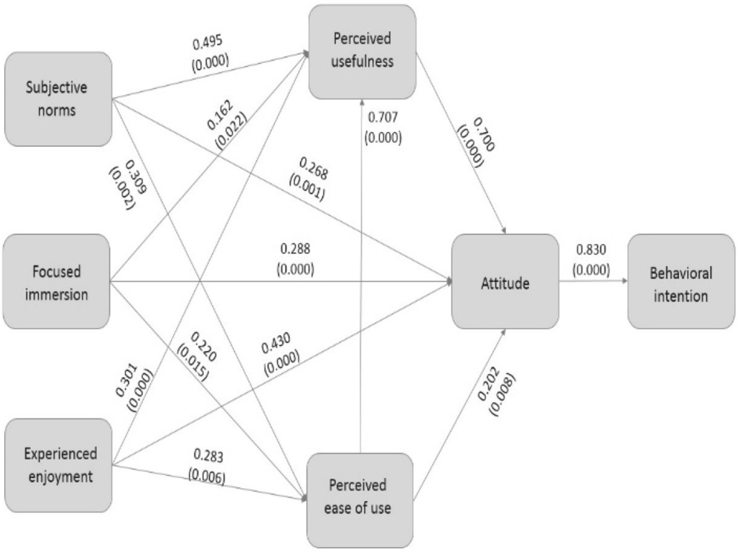


Fig. 5. Factors influencing computer students' attitude towards the use of Arduino MDBs in the computer science class

5 Conclusion

This paper contributes to research on the acceptance of technology in a class by using and extending the TAM to identify the motivational factors influencing the acceptance behaviour of students towards using Arduino MDBs in the computer science class.

As several studies suggest, external factors also have a significant influence on the attitude towards the acceptance of technology [11,26]. In this paper subjective norms, focused immersion and experienced enjoyment were identified as possible external factors that influenced the students' acceptance towards Arduino MDBs. External variables, together with internal variables of TAM, were used to determine students' acceptance towards the use of this technology in class. From the results presented, it is evident that subjective norms significantly influences students' perceived usefulness and perceived ease of use towards using Arduino MDBs in class.

It is in line with literature about the student cohort, as some studies found students to be susceptible to social influence or the view of the people close to them in using technologies in different fields of application such as online gaming [16], mobile banking [25] and social networking [15]. Additionally, people who are less experienced in using technology will be influenced to a larger extent by people close to them [36].

Furthermore, the results show that students' attitude towards using this technology in class is influenced significantly by their experienced enjoyment. This is similar to a previous study [8] that found experienced enjoyment to have a significant influence on attitude, which in turn has a significant influence on the intention to use. Insights gained from this study may assist researchers and educators in understanding the factors that influence a student's attitude towards using technology as a teaching tool in class. Similar to previous studies that indicate that the Arduino MDBs are popular platforms to use in education on Electronics, Automation and Robotics [7,18,23], the findings of this study confirm computer science students' positive attitude towards using Arduino MDBs in class. Therefore, educators should incorporate technology such as Arduino in computer science classes as the results of the study show that students regard the implementation of Arduino in class as useful which contributes towards a positive attitude towards technology in class. As we are entering the fourth industrial revolution, computer science students should be encouraged to use technology in class. Students should be provided with the opportunity to gain experience in using technology to complete projects requiring creative thinking and problem-solving skills.

As depicted in the model, ease of use has a less significant influence on the attitude of computer science students towards the acceptance of Arduino MDBs in class. This could be related to the fact that computer science students are problem solvers by nature and do not regard exploring how gadgets of technological nature work as an obstacle but rather as a challenge. However the causes and possible benefits of ease of use having a less significant influence on the attitude of computer science students towards technology in class could be investigated as part of future research on Industry 4.0 oriented education in Computer Science at higher education institutions.

References

1. Akşit, M.: The role of computer science and software technology in organizing universities for Industry 4.0 and beyond. In: 2018 Federated Conference on Computer Science and Information Systems (FedCSIS), pp. 5–11. IEEE (2018)
2. Anoop, B., Jithin, M., Mukunda Dasa, K.: Ultra stable flying robot (2015)
3. Basawapatna, A.R., Repenning, A., Koh, K.H., Nickerson, H.: The zones of proximal flow: guiding students through a space of computational thinking skills and challenges. In: Proceedings of the Ninth Annual International ACM Conference on International Computing Education Research, pp. 67–74 (2013)
4. Benešová, A., Tupa, J.: Requirements for education and qualification of people in industry 4.0. Procedia Manuf. **11**, 2195–2202 (2017)
5. Binkley, M., et al.: Defining twenty-first century skills. In: Griffin, P., McGaw, B., Care, E. (eds.) Assessment and Teaching of 21st Century Skills, pp. 16–26. Springer, Dordrecht (2012). https://doi.org/10.1007/978-94-007-2324-5_2
6. Buechley, L., Eisenberg, M., Catchen, J., Crockett, A.: The LilyPad Arduino: using computational textiles to investigate engagement, aesthetics, and diversity in computer science education. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 423–432 (2008)
7. Candelas, F., et al.: Experiences on using Arduino for laboratory experiments of automatic control and robotics. IFAC-PapersOnLine **48**(29), 105–110 (2015)

8. Cheikh-Ammar, M., Barki, H.: It desirability as a means to exploring is post-acceptance usage of everyday technologies: conceptualization and comparative assessment. In: DIGIT 2014 Proceedings (2014)
9. Chu, S.K.W., Reynolds, R.B., Tavares, N.J., Notari, M., Lee, C.W.Y.: Twenty-first century skills and global education roadmaps. In: 21st Century Skills Development Through Inquiry-Based Learning, pp. 17–32. Springer, Singapore (2017). https://doi.org/10.1007/978-981-10-2481-8_2
10. Davis, F.D.: Perceived usefulness, perceived ease of use, and user acceptance of information technology. MIS Q. **13**, 319–340 (1989)
11. Davis, F.D., Venkatesh, V.: A critical assessment of potential measurement biases in the technology acceptance model: three experiments. Int. J. Hum.-Comput. Stud. **45**(1), 19–45 (1996)
12. Eichinger, P., Hofig, B., Richter, C.: Education 4.0 for mechatronics-agile and smart. In: 2017 International Conference on Research and Education in Mechatronics (REM), pp. 1–7. IEEE (2017)
13. Fishbein, M., Ajzen, I.: Belief, Attitude, Intention, and Behavior: An Introduction to Theory and Research. Philosophy and Rhetoric, vol. 10., Wesley, Reading (1977)
14. Gleason, N.W.: Higher Education in the Era of the Fourth Industrial Revolution. Springer, Singapore (2018)
15. Hsu, C.L., Lin, J.C.C.: Acceptance of blog usage: the roles of technology acceptance, social influence and knowledge sharing motivation. Inf. Manag. **45**(1), 65–74 (2008)
16. Hsu, C.L., Lu, H.P.: Why do people play on-line games? An extended TAM with social influences and flow experience. Inf. Manag. **41**(7), 853–868 (2004)
17. Ibrahim, R., Jaafar, A.: User acceptance of educational games: a revised unified theory of acceptance and use of technology (UTAUT). Int. J. Educ. Pedagog. Sci. **5**(5), 557–563 (2011)
18. Jamieson, P., Herdtner, J.: More missing the boat-Arduino, Raspberry Pi, and small prototyping boards and engineering education needs them. In: 2015 IEEE Frontiers in Education Conference (FIE), pp. 1–6. IEEE (2015)
19. Kamble, P., Khoje, S., Lele, J.: Implementation of paper PCB using conductive ink 2D plotter. In: 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA), pp. 1–6. IEEE (2018)
20. Kozák, Š., Ružickỳ, E., Štefanovič, J., Schindler, F.: Research and education for Industry 4.0: Present development. In: 2018 Cybernetics & Informatics (K&I), pp. 1–8. IEEE (2018)
21. Lai, C., Wang, Q., Lei, J.: What factors predict undergraduate students' use of technology for learning? A case from Hong Kong. Comput. Educ. **59**(2), 569–579 (2012)
22. Lee, J., Bagheri, B., Kao, H.A.: A cyber-physical systems architecture for Industry 4.0-based manufacturing systems. Manuf. Lett. **3**, 18–23 (2015)
23. Ling, O.S., Wah, J.L.P.: UCT's foundation students' perception towards Arduino as a teaching and learning tool in STEM education. e-Bangi **16**(3) (2019)
24. Lu, Y., Zhou, T., Wang, B.: Exploring Chinese users' acceptance of instant messaging using the theory of planned behavior, the technology acceptance model, and the flow theory. Comput. Hum. Behav. **25**(1), 29–39 (2009)
25. Luarn, P., Lin, H.H.: Toward an understanding of the behavioral intention to use mobile banking. Comput. Hum. Behav. **21**(6), 873–891 (2005)
26. Mathieson, K.: Predicting user intentions: comparing the technology acceptance model with the theory of planned behavior. Inf. Syst. Res. **2**(3), 173–191 (1991)

27. Nakamura, J., Csikszentmihalyi, M.: The concept of flow. In: Handbook of Positive Psychology., Oxford University Press, Oxford (2002)
28. Pallant, J.: SPSS Survival Manual. McGraw-Hill Education, London (2013)
29. Saadé, R.G., Bahli, B.: The impact of cognitive absorption on perceived usefulness and perceived ease of use in on-line learning: an extension of the technology acceptance model. Inf. Manag. **42**, 317–327 (2005)
30. Scharf, F., Winkler, T., Hahn, C., Wolters, C., Herczeg, M.: Tangicons 3.0: an educational non-competitive collaborative game. In: Proceedings of the 11th International Conference on Interaction Design and Children, IDC 2012, pp. 144–151. Association for Computing Machinery, New York (2012). https://doi.org/10.1145/2307096.2307113
31. Schwab, K.: The fourth industrial revolution. World Economic Forum (2017)
32. Syafnidawaty, S., Susanto, F., Gumilar, P.: Prototype pemantau bus menggunakan GPS tracking geolocation Berbasis ADUINO UNO. J. CERITA **4**, 18–23 (2018). https://doi.org/10.33050/cerita.v4i1.551
33. Tan, O.S., Teo, C.T., Chye, S.: Problems and creativity. Problem-based Learning and Creativity, pp. 1–14 (2009)
34. Teo, T.: Factors influencing teachers' intention to use technology: model development and test. Comput. Educ. **57**, 2432–2440 (2011)
35. Venkatesh, V., Bala, H.: Technology acceptance Model 3 and a research agenda on interventions. Decis. Sci. **39**, 273–315 (2008)
36. Venkatesh, V., Davis, F.D.: A theoretical extension of the technology acceptance model: Four longitudinal field studies. Manag. Sci. **46,** 186–204 (2000)
37. Venkatesh, V., Morris, M.G., Davis, G.B., Davis, F.D.: User acceptance of information technology: toward a unified view. MIS Q. **27**, 425–478 (2003)

Author Index