

Chapter 14

Task Versus Process: A Taxonomy for Group Projects



Clive Rosen

Abstract Group projects appear to be a mandatory requirement of undergraduate degree programmes in software engineering and other, similar courses. But what are these projects for, and what do we expect students to learn from them? Often the group project exists within the curriculum simply because it is required by various authorities and without a clear pedagogic rationale. The language used in module specifications usually refers to employability skills and collaborative working, but often disguises a wide range of different delivery approaches. Assessment regimes may prioritise outputs over the process of producing those outputs. Even the word “process” is ambiguous. Do we mean the systems development process or the group process? If the former, why do we need a group? If the latter, how can this be assessed to the satisfaction of students and external examiners? The “Task Versus Process” taxonomy firstly suggests a model for categorising the aims and objectives of the group project, secondly, evinces an appropriate assessment strategy and finally suggests some approaches that can be taken if staff teams wish to address the “Personal Development” quadrant of the taxonomy. The chapter contextualise the student group project within broader aspects of curriculum design and then attempts to clarify the multifarious and often contradictory objectives proposed for student group projects. Fully appreciating what it is we, as instructors hope to achieve in group projects takes us beyond resigning ourselves to the compulsory inclusion of the project. Projects offer the opportunity to help our students achieve their full potential and to actually deliver the learning outcomes.

Keywords Learning and teaching strategy · Student group projects · Group project assessment

C. Rosen (✉)
Passerelle Systems Limited, Newcastle Under Lyme, UK
e-mail: passerelle.ltd@outlook.com

© Springer Nature Switzerland AG 2023
J. Carter et al. (eds.), *Higher Education Computer Science*,
https://doi.org/10.1007/978-3-031-29386-3_14

187

14.1 Introduction

What is the purpose of student group projects in software engineering courses? Their ubiquity suggests that they must be a very important component in such courses. However, when one explores the range of learning outcomes associated with group project modules, one finds a wide range of potentially conflicting objectives and often assessment approaches that are not wholly compatible with the learning outcomes. This chapter explores the reasons why this might occur and proposes a taxonomy that associates the actual objectives of the group project with the assessment approach taken. The purpose of the group project is situated within the more general model of the learning process (Fig. 14.1) in order to plot clear connective patterns from programme learning outcomes, through module learning outcomes to appropriate assessment methodologies. It is hoped that this approach will enlighten discourse on programme design in IT and computing related courses.

Group project module learning outcomes often cite such employability related requirements as the ability to work in a team, decision making or effective communication. These nebulous terms are often treated as an inexorable consequence of the group project process, avoiding the fact that learning about oneself requires self-reflection and challenging preconceived personal constructs. Students need to be presented with opportunities to learn and also be given support for self-reflection. Computing students in particular are often reluctant to engage with this sort of process. The group project module provides the opportunity to integrate employability skills into the curriculum. This chapter offers some suggestions of how this might be done. However, if it were that easy, everybody would be doing it. Resources play a part, but there is also a challenge for staff. We cannot expect students to reflect

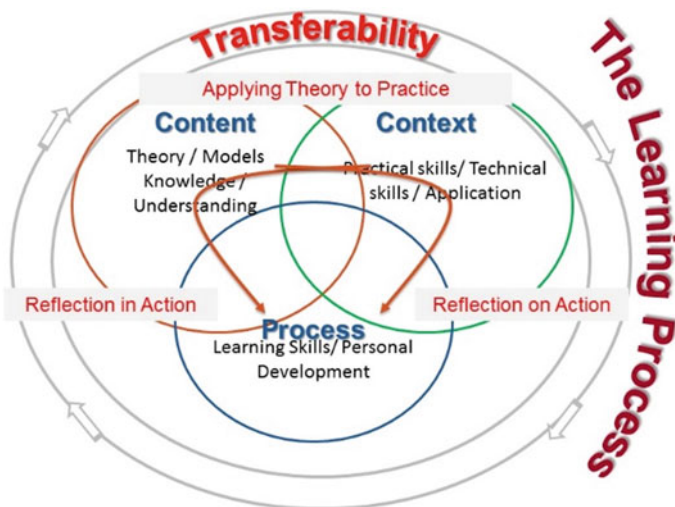


Fig. 14.1 The learning process in higher education (Rosen and Schofield 2011; Rosen 2015)

on themselves without having gained some self-awareness and some ability for self-reflection ourselves. Many would not see this as part of their lecturing role. It is hoped that some of the suggestions made in this chapter will help to overcome this reluctance on the part of both students and staff.

14.2 The Purpose of Student Group Projects

The demands made on student group projects are often extensive. Projects are often expected to deliver a wide range of outcomes including “real world” experience of industrial weight project work, employability skills, team-working and communication skills. Group projects are also viewed as capstone modules, integrating knowledge from other parts of the curriculum (such as database development, web site development and programming). This range of expectations can lead to students becoming confused regarding the purpose of the module and what is expected of them. It can also result in a situation where the mechanisms used for assessment either fail to test the actual learning outcomes, fail to challenge students to produce their best work, or fail to provide sufficient opportunity for students to exercise appropriate critical self-awareness.

Before considering the role of the student group project, it might be helpful to stand back a little and consider how higher education relates to governmental and industrial expectations of software engineering students and how the group project sits within this context and the software engineering curriculum. The diagram above helps to illustrate this positioning.

This model identifies three intersecting circles representing models of student learning which can support the development of a learning and teaching strategy at both the curriculum or module level. The three inner circles represent “content”, “context” and “process”. These circles intersect in a “Venn” style diagram suggesting the lack of precise boundaries between the three forms of learning.

“Content” is comprised of the theories pertinent to an academic domain, abstract knowledge associated with it and accumulated experience concerning the subject area. It is abstracted from any given situation. Content can normally be found in text books, journal papers etc. and is often delivered in HE settings through formal classes. Content is the domain that academia traditionally focused on delivering and, more particularly, developing, through on-going research. So an example of content in IT would be normalisation.

“Context” is the application of a particular theory to a given context and requires practical/implementation skills and the ability to select an appropriate solution from the body of knowledge (content). Vocational subjects generally give more emphasis in this sphere than the humanities, as, being able to deliver the product or service is an essential element of success. Context is situated in real world problems and encompasses knowledge and experience associated with practical problem solving. When employers complain about the lack of readiness of graduates to work, it is often issues associated with the context sphere that they complain about.

The association of context with competency contrasts with the association of content with understanding. An example of the difference between the two within the computing domain would be the capability of a student to program in a particular language (context) compared to an understanding of the principles of programming languages (content). In reality, both are needed. Students struggle to understand programming principles without knowing how to solve problems in a given language, yet they must understand the principles to be able to learn the range of languages they may be confronted with when they leave university. Most courses contain elements of both content and context. Examples and case studies are often used in teaching for just this purpose. Without context/application, abstract knowledge is interesting, but essentially unproductive.

Whilst both understanding and technical competence are considered to be essential requirements of the higher education system, neither are sufficient either singularly nor together. An implicit demand in recent years, particularly from industry (but arguably often undermined by governmental policies) has been the ability of graduates to transfer knowledge from one context to another. This requires the third sphere, “process”.

Today’s students need greater capability; in their ability to learn independently; to use their knowledge and understanding to solve unfamiliar problems; to understand the processes and standards involved in learning; to recognise which tools and techniques are applicable to a particular context as well as other social and learning skills. Students need the problem solving skills of logical reasoning, deduction, research and critical evaluation. These capabilities do not automatically emerge from content and context, but need to be nurtured and encouraged in equal measure through reflective practice. Schön’s (1991) concepts of “reflection in action” and “reflection on action” can be applied here to support the learning process. “Reflection in action” is the idea that (in Schön’s phrase) the “Reflective Practitioner” (in this case the student) does not simply do work, but thinks about it whilst they are engaged in it to ensure that any particular course of action is the best available at the time. “Reflection on action” is the idea that once a particular project is complete, a professional should not give a huge sigh of relief and move on to the next project, but should review the project to see what can be learnt from it. Were there any mistakes made? Could the output have been more effective? Were any inefficiencies introduced into the process? (The parallels between this and the Shewhart Cycle (1986) and Deming (2000) quality improvement process should not be overlooked.) Traditionally, one might argue, that higher education has used research as a vehicle to enculturate students into these activities, but the group project represents an ideal opportunity to encourage students to become reflective, particularly if ongoing progress reviews are included in the module implementation. However, reflection, particularly reflection on one’s own performance, involves the ability to be self-critical which necessitates a level of self-awareness and self-confidence; qualities that are often limited amongst computing students.

In practice content, context and process are not independent of each other. Teaching staff switch between them in rapid order, drawing on case studies to expound

theory, asking students what they think might be going wrong (theorising) in a practical application and asking them to consider alternative approaches to the problem they are trying to solve. However, this model does offer an idealised conceptualisation of what we might be trying to achieve. This objective for Higher Education may be aspirational and idealistic, but it is expressed in Benchmark statements for all the computing subject areas. And, the group project provides curriculum designers with the opportunity to demonstrate the development of these skills in the programme. This, it might be suggested, is often one of the sources of conflict within the group project module. For how can we help develop self-reflection in students and how do we assess it? One might argue that this is particularly difficult with computing students who are often resistant to the notion that self-awareness, personal reflection and personal development are necessary to becoming software or systems engineers.

Student group projects rarely include new knowledge and skills. They are more likely to focus on applying the skills that students have already acquired in other modules to a particular situation (or “context” in the diagram above) and often in a more complex environment. But the learning outcomes for the group project module will often include “the ability to work in a team”, “real world understanding”, “creative thinking” and so on. These skills are “Process” skills as defined by the model above. However, students will usually be expected to be able to demonstrate that they have actually understood what they have done. They might be expected to explain why they did something one way and not the other. In other words, to use critical analysis to show that they know why they did something. This activity is situated in the “Content” part of the model above. So, when student group projects are considered in the light of the above model, they sit very firmly at the nexus of the diagram. It is unsurprising therefore that lecturers may feel pulled in several different directions at once. For any particular group project module, should the emphasis be on the real world experience, reasoned decision making, team working of something else? The consideration of “Task Versus Process” may help to articulate the answers to some of these questions by informing the module design process and helping to make decisions more explicit.

14.3 Task Versus Process

The “Task Versus Process Grid” provides a means of considering the primary purpose of the group project in a particular programme. The grid offers two parametric dimensions. On the vertical axis we have the question of, “for this module, on this programme, do we consider students’ capability to complete the task more or less important than their understanding of how to undertake the task (i.e. the process)?” However, process can have two different meanings here. The first meaning suggests that the process we are considering is the development process. A second interpretation of process is the team process. Are we more concerned about the way the system is produced or an understanding of the group dynamics and hence the ultimate capability of the student to work in a real industrial team? The second axis helps to clarify

this difference. If the former, we would be more concerned with the outputs from the students' activity. If the latter we would pay more attention to the students' ability to think about the way the group has operated, and how the intra-group process has affected the conduct of the task.

The four quadrants therefore provide us with a taxonomy to consider where we wish to place the emphasis for the module. If we are concerned with a student's technical capability, this would place us in the lower left hand quadrant of the model. We would want the student to demonstrate their ability to produce some outputs (probably in the form of a database, website, program or app) and we would concentrate on the knowledge associated with "doing" the task. Alternatively, we may be concerned less with the actually getting the task done than with how the students went about the task. Did they establish the requirements? Did they produce a product design? How rigorous was their testing? These are questions regarding the governance of the project and place us firmly in the "software engineering" quadrant.

If the purpose of the project is to promote employability skills, we would be concerned with how students worked together to produce their outputs. The artefacts themselves would be less important than the students' ability to reflect on what worked, what did not, and how they communicated with each other to arrive at appropriate and timely decisions. What did the students learn from their experiences?

The final quadrant, the "personal development" quadrant is perhaps the most ephemeral and also the least well-articulated quadrant. It may also be the quadrant most aspired to when curriculum designers construct courses, and perhaps the hardest to achieve. Questions relating to students' personal development include how did the group process support or prevent you from contributing to your maximum ability? Were you able to contribute to decision making? Were you able to reflect in and on action (Schön 1991)? These questions can be the most difficult in which to engage students, and the most difficult to assess. Teaching staff are probably least qualified to support student learning in this area and, perhaps therefore least willing to focus on this aspect of the group project.

Figure 14.3 summarises this discussion.

14.4 Assessment

The argument above would be incomplete without a discussion of how to assess team projects within each quadrant as, if each focusses on particular forms of output, each should use those outputs as the means of assessment. Figure 14.4 outlines the means of assessment that would be best associated with each quadrant.

It can be seen that the range and variety of potential assessments is extensive. Learning outcomes defined in module specifications are often ambiguous and lack precision, leaving it to the module leader to design their own assessments. The breadth of scope however means that it is not possible to assess every outcome which is why having more clearly defined learning outcomes becomes important (notwithstanding the point that if the module team are not clear what they are trying to achieve, the

students are likely to be confused). Some may argue with the details of the taxonomy presented here, but this analysis clearly demonstrates the difficulties associated with assessing group projects. Whilst the “Software Engineering”, “Technical Capability” and “Employability Skills” quadrants all have relatively self-evident outputs that can be assessed, the problem is deciding on what to focus.

The Personal Development quadrant is more problematic for many reasons. It has already been suggested that computing students often fail to appreciate the importance of the more esoteric skills resident in this quadrant. Often staff feel reluctant to stray into an area they might feel unqualified to assess. The subjectivity associated with assessing personal development is also problematic in a discipline aspiring to achieve engineering status. Furthermore, many staff have developed an antipathy to assessing personal development from the experience of trying to assess “key skills” in HND, “core skills” in apprenticeship schemes and “employability skills” as part of the employability agenda. One lesson many have learnt from these experiences is that if employability skills are not fully integrated into the curriculum in general, and the group project module in particular, students react poorly to their add-on, ad hoc injection into a programme. This raises the question of how can seamless integration be achieved in such a practical subject area such as computing. The next section of this chapter suggests some approaches that can be used as part of the group project. Other chapters in this volume look at other approaches.

14.5 Implementation

This section will concentrate on the introduction and assessment of the skills associated with the upper right, “Personal Development” quadrant of the Task Versus Process Grid (Figs. 14.2, 14.3 and 14.4). Staff may feel more confident implementing some suggestions than others. Logistics and resources also play a part as do the peculiar academic environment surrounding all group projects. With this in mind, staff may choose to introduce changes over a period of time if they feel any of these ideas are appropriate to the pedagogic context in which they are working.

14.5.1 *Live Client Projects*

A point of dispute amongst group project module leaders is whether or not to trawl for real client projects, allow students to construct their own projects or for the module team to design projects the students can do as a group. The choice may well depend on the particular circumstances in which the project is being conducted and the objectives of the project. The Task Versus Process taxonomy above can help with this decision. However, often the choice boils down to more pragmatic considerations. Do the module team have sufficient time to find and vet potential projects? Can we trust students to deliver for real clients? What mechanisms are available for installation on

Fig. 14.2 Task versus process grid

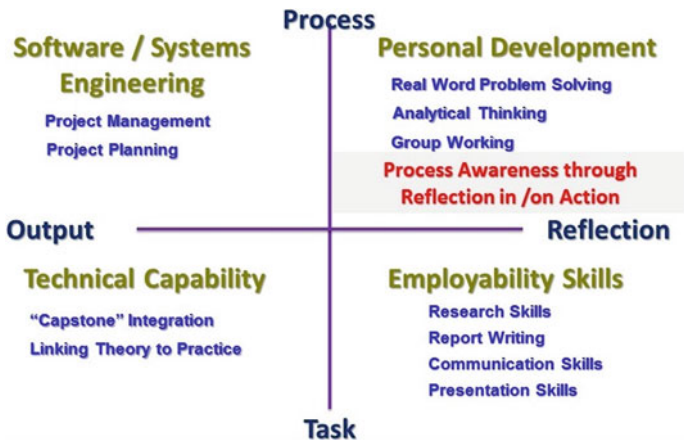
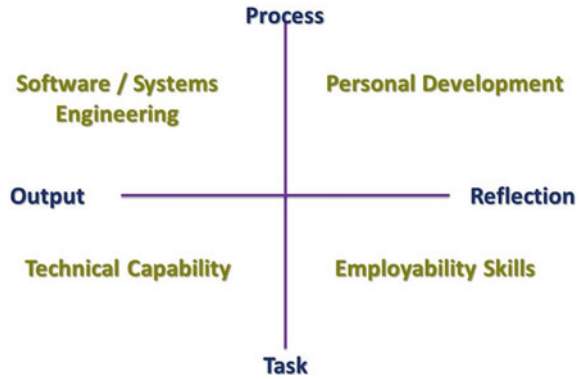


Fig. 14.3 Activities most associated with each quadrant of the task versus process grid

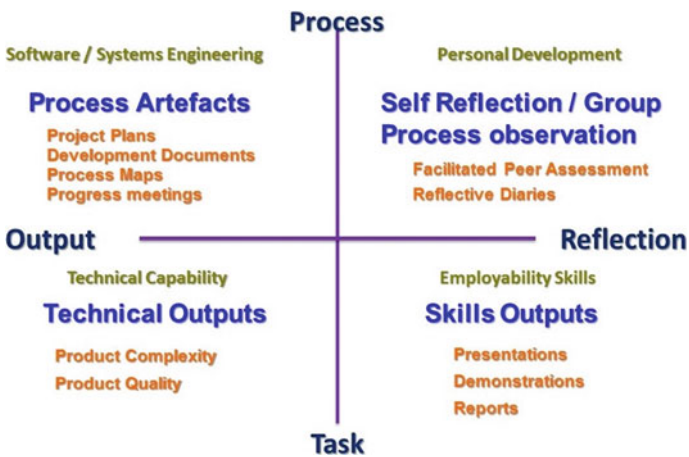


Fig. 14.4 Assessment associated with each quadrant in the task versus process grid

client systems and ongoing support? Real projects tend to be much messier and more complex than contrived projects. This means that students need to be given much more hands on support during the project; particularly with client management. But this is also the strength of using real projects and real clients. Students discover that part of systems development is client management; working with clients to identify their actual requirements, agreeing with them what is practical and what can be delivered given the time and resource constraints. Clients do not conform to the stereotypes implied in software engineering texts. Working with real clients exposes students to these realities and provides many powerful learning opportunities.

Furthermore, students engaged on real projects experience one of the fundamental differences between employment and student life which often goes unacknowledged; that work is undertaken for an employer, not for oneself (excepting self-employment). Studying is a very self-centred occupation and feeds into many students' egocentricity. Employment implies doing work for other people (even when self-employed) and this involves a radically different mind-set. Real, live projects can help with this transition in a relatively safe environment, at least one where the module staff can provide a safety net if necessary.

One further factor when considering whether live projects should be used, especially when external (to the institution) clients are adopted, is the role of the student as ambassador to the college or university. This can be a very positive experience for students, and the institution, if students embrace the responsibility, but can cause difficulties for the programme staff if things go wrong. Teaching staff must decide for themselves the balance of risk and benefit. Client preparation is essential and staff cannot absolve themselves from continuous client management, particularly when working with external clients. There is undoubtedly less work for the teaching team when contrived projects are undertaken, but they are potentially less rewarding for both students and staff. Group projects present this opportunity. There is rarely a shortage of potential clients. Only the module team can assess the cost/benefit ratio in their environment.

14.5.2 Student Application to Chosen Project

One approach that can help to integrate employability skills with the group project is to require students to make a formal application to a particular project. This works particularly well when external clients are offering projects. The assessment would require students to produce a CV, write a letter of application to the project provider and attend an interview for the project. (Students would only get their first choice project if they pass the interview.) Careers staff and project providers can be recruited to help with the interview process so the experience mirrors more closely a genuine recruitment exercise. An assessment centre can also be included in the process if time and resources permit.

One essential element of this activity is immediate feedback to students from the interview. It is remarkable the learning that can occur even in this relatively

contrived situation when a student has underprepared for the interview and then received feedback from an interview panel on their performance. A large majority of students will never have previously experienced a formal interview and the reality and immediacy of such a meaningful interview is a very powerful learning experience. It is important for the interview panel to set the right tone. Hostile interrogation could generate a student's defence mechanisms which would not be helpful, a potential danger when a student fails to take the process seriously or fails to prepare for the interview. Having the external client on the interview panel brings a number of benefits. Not only does it make the experience more real, it also increases the client engagement in the project process. At the same time, being interviewed by a real client promotes greater student commitment to the group project and therefore improves the prospects of their full engagement.

A commonly recognised problem with the assessment of student group project is the evaluation of the student free-rider, i.e. a student who contributes little to the project, but expects to be awarded equal marks to those that have worked on the project. Facilitated Peer Assessment, see below, helps to alleviate this problem, but interviewing students for the project in the first place, also helps student engagement by increasing their initial investment in the project.

14.5.3 Skills Inventory

The skills inventory presented here is a development from a skills matrix that was presented to the HEA BMAF Placements workshop in 2006. The inventory has evolved to its current version since that date. Six principles have emerged during this evolution:

1. To provide a supportive structure for student reflection.
2. To minimise the psychic threat to students.
3. To encourage students to identify specific examples to work on.
4. To focus on learning from personal experience.
5. To make the process of self-reflection intrinsic to the exercise.
6. To use the skills employers say they want as the vehicle for self-evaluation.

As can be seen in Table 14.1, each row represents one of the skills employers consider to be important. The student is asked to identify a specific example of when they have demonstrated each skill. If they have more than one example, they can insert additional rows under the same heading. The "activity description" column requires the students to articulate the specific circumstances. The "when" and "where" columns are included to ensure a particular time and place, and not a generalised activity. In the "what did you do" column, the word to emphasise is "you". Students often experience these activities as part of a group, and it is important that they learn to differentiate themselves as an individual rather than simply a member of a group so that they can attribute the outcome to their personal contribution. Students are often reluctant to appreciate what they do well, and this exercise helps

them identify strengths as well as learning experiences. The seventh column, the “learning” column is perhaps the most important. It is these entries where the ideas of self-reflection and self-evaluation are expressed. Having the experience alone is insufficient. Professional development requires personal reflection on what has been learnt from the experience. To experienced professionals this may become integral to their professional practice, but to students and new entrants into the profession, the concept of continuous self-evaluation and reflection are not self-evident and often students are never told that it is required. This exercise makes explicit an activity experts often take for granted.

The final column is optional. It is there to help students develop an action plan and to decide on which skills should be priorities for future development.

The inventory can be used as a standalone exercise, but has greater relevance if students are preparing to apply for placement or post graduate opportunities. UK employers commonly adopt a competency based approach to interviewing applicants (sometimes known as CAR, Context, Action, Result). This takes the form of asking questions such as ‘describe a situation when ... what did you do ... what was the outcome?’ The skills inventory helps prepare students for such questions and therefore improves their chances of performing well in interviews.

Table 14.1 Skills inventory

Skill	Activity description	When	Where	What did you do?	Outcome	Learning	Priority
Leadership/initiative taking							
Influencing/negotiating							
Team work							
Effective communication							
Self-motivation							
Decision making							
Planning/organisation							
Working under pressure							
Personal development							
Commercial awareness							
Presentation/report writing							

14.5.4 Facilitated Peer Assessment

Facilitated Peer Assessment (FPA) was first described in Rosen (1996) as a means of assessing the individual contribution to group projects, but it is also a means of supporting student reflection in that students are required to assess their own performance in relation to other members of the team and to give feedback to other team members. As such, it provides a vehicle for exploring self-reflection both as part of the project and as part of the assessment process.

FPA has the form of a post project review in which students in a face to face group meeting with a module tutor evaluate the success or otherwise of the project. As a group, they must also reach a consensus on the allocation of individual members' contributions to the project. The role of the staff member is to facilitate the discussion rather than to influence it (other than to encourage the students to reach a conclusion).

FPA has several advantages over other forms of peer assessment which usually consist of students completing undisclosed assessment forms about each other. The lack of transparency with this traditional approach undermines the value of peer assessment. Students can attribute any contribution level to their fellow team members, but it is unclear what criteria or perceived equity of treatment they are applying. Group projects can provide excellent opportunities to explore unconscious bias in teams, but undisclosed assessments of each other fail to challenge these biases in the assessment method. As a result, it is often down to the module team to moderate the students' assessments. It can be argued that moderation of marks is the responsibility of the module team, but if the module team are going to moderate the peer assessment, students may question the reliability, and possibly the validity of the process. Furthermore, academics' moderation of the peer assessment process undermines the students' sense of responsibility for the process as they cannot take full ownership of the process.

FPA avoids these problems. Firstly it is open and transparent. Students know what is being said about them by other team members, both positives and negatives. FPA therefore has the capability of being a very affirming activity as well as a functional one. Students retain responsibility for the process as the role of the staff member is facilitation, not moderation. FPA also provides the opportunity for students to reflect on the assessment process and the legitimacy of various criteria. They must decide whether the team's project manager has contributed more or less than its chief designer. What is the nature of contribution? This has the effect of requiring students to reflect on their own performance, as does the receipt of feedback from other team members. FPA does challenge the assertiveness of less articulate students, but it is the role of the facilitator to support students to express themselves (one of the key requirements for employability).

Student ownership of the FPA process has an important side effect. Group project module leaders will be familiar with the concept of the "free riding" student, the one that hopes to benefit from the work other students do without contributing much themselves. Often the only (meaningful) sanction the staff team have is to sack such students from the team. This is very much a nuclear option as it results in the dismissed

student failing the module and staff have to have very clear evidence that a particular student has not, and will not, contribute. In group dynamics terms, a student's lack of engagement in the project can be very complex. Gender, race, culture and language can all play a role in peer acceptance or rejection within the team, and an apparently non-contributing student may feel that they are being excluded by the group rather than the student not wanting to contribute. Students have responsibility and authority for FPA so they have greater responsibility for the successful functioning of the group. FPA gives the students themselves a sanction against non-contributing members; one that can be referred to during the project to bring pressure to bear on a student and one that avoids the nuclear option. The result is that contributing students feel more in control, and often respond positively to the responsibility they have for project progress. The nuclear option is still available, but if it does need to be invoked, there is usually a clearer evidence trail to support it.

FPA is problematic as it does depend on the confidence of academic staff to manage the FPA process. Smouldering conflict that has been present within the team can, and sometimes does emerge during the FPA. Staff may feel that managing such conflict is not one of their responsibilities, but if this tension has resulted in the team not achieving its full potential, staff should at least be aware of it, and it must surely be beneficial for students to explore how such tensions affect the progress of a development project. After all they are likely to face these sorts of situations during their working life and if they have reflected on it within the relatively safe and supported group project environment, they will be better prepared for it in the world of work.

Managing conflict in group projects is challenging for staff. It is probably the aspect of student group projects academics find most off putting, and probably one of the reasons why many staff, when charged with the responsibility of running group projects, avoid the "Personal Development" quadrant of the "Task Versus Process" grid. Yet the ability for graduates to be able to work successfully with other team members often marks them out as potential employees, and in a profession where such capability is a rare commodity (Teague 1998; Capretz 2003), suggests that the effort may produce its own rewards.

14.5.5 Team Membership Selection Process

The team membership selection process is somewhat orthogonal to the taxonomy of group projects as any team selection method could be used whichever quadrant the project is focussed on. However it is nevertheless a bone of contention and teaching staff need to be cognisant of the influence the selection process can have on the group dynamics. In group theory, group formation is one of the most significant events in the life of a group, and the method of its formation reverberates through the group process in one form or another.

The debate over the team selection process revolves around whether or not students should be allowed to select their own groups or whether team members should be

selected by staff members. The arguments over which is best have been well rehearsed and are not going to be regurgitated here. However, two options are available that may be of interest that somewhat sidestep this problem and provide an opportunity to encourage student reflection.

The first is to make use of an early tutorial session to discuss with the students the method of selection to be used. This involves brainstorming the various ways in which groups can be selected. Students usually identify between 20 and 40 different possibilities ranging from the obvious, that they (the students) should select their own groups or the teacher announces the teams, to geographic proximity to where students live and the physical height of the students. The class then go on to discuss the pros and cons of each method, eliminating some for practical or ethical reasons and looking at the more legitimate approaches critically. This discussion often raises some ethical and moral questions such as ‘what happens to the student no one wants in their group?’, ‘what about the new student who nobody knows?’ or ‘what happens when friends fall out?’. After this discussion, the students then decide which method to adopt. It has to be said that students usually adopt a self-selection method, though not always. However this approach does place the responsibility of the selection method with the students. They are therefore unable to claim that if the project goes sour it was the teacher’s fault. Students do seem to like this discursive approach and it is nearly always enlightening, both for students and academic staff.

The second approach that has been alluded to above is the ‘project first’ approach. This is where the projects are defined first and students then choose the project they want to work on. This approach need not be combined with the interviewing process described above or include external clients, but it does mean that fairly clear specifications of what the project involves need to be provided to the students. Again this approach places the responsibility for the team selection with the students, albeit indirectly in this case. The project first approach subtly changes the initial dynamic of the module emphasising the importance of the project task over being a bit of fun with mates at the end of the year.

14.6 Conclusions

This chapter aims to develop a pedagogic taxonomy for determining the focus of student group projects in the context of the objectives of any given software engineering course. It identifies four quadrants that might become the focus of a group project module on that programme, the “Software Engineering” quadrant, the “Technical Capability” quadrant, the “Employability Skills” quadrant and the “Personal Development” quadrant. For each quadrant, it identifies the type of activities associated with the quadrant and the assessment artefacts that might be used to assess student capability. In providing this mapping, it is hoped that coherence and consistency between assessment methodology and learning outcomes can be maintained, and that, as a result, staff may be better able to articulate their own aims for the module and prepare students for the assessments they might expect. Congruence between

learning outcomes and assessment methodology seems to be taken for granted in other modules within a software engineering curriculum. Because there can be such a wide variety of both overt and covert objectives for group project modules, it seems important to explicitly resolve any ambiguities that might exist regarding what the group project module in a given programme is really for. It is hoped that this taxonomy will help in this regard.

The “Personal Development” quadrant in particular can be quite challenging. It takes academic computing staff into unfamiliar territory and areas which it might be argued are not their responsibility. However, group projects are often cited as being key components of the employability agenda, and UK universities and colleges have signed up to this agenda, however reluctantly in some cases. In reality, it is not easy to differentiate elements of personal development from employability in general. Team working for example always involves managing personal differences and sometimes even conflict. If we are genuinely to address such aspects of employability, we probably need to provide students with the opportunities to reflect on, and learn from appropriate experience. The group project module can be such an opportunity. Some suggestions have been proposed in this chapter which can be employed to help to stimulate student personal development. Many may still wish to steer clear of any involvement in this aspect of the current higher education agenda. Whichever approach is adopted, it is hoped that the taxonomy presented here provides a framework for decision making.

References

- Capretz LF (2003) Personality types in software engineering. *Int J Hum Comput Stud* 58(2):207–214
- Deming WE (2000) *Out of the crisis*. MIT Press, Cambridge, Mass
- Rosen C (1996) Individual assessment of group projects in software engineering—a facilitated peer assessment approach. In: 9th annual conference on software engineering education. Daytona Beach Florida: IEEE Computer Soc Press
- Rosen CCH (2015) A philosophical comparison of Chinese and European models of computer science education. (A Discussion Paper) 11th CHINA—Europe International Symposium on Software Engineering Education Zwickau Germany April 2015
- Rosen CCH, Schofield R (2011) Reliability and validity in work-based learning, work based learning futures V, Derby
- Schön D (1991) *The reflective practitioner: how professionals think in action*. Aldershot, Arena
- Shewhart WA (1986) *Statistical method from the viewpoint of quality control*. Dover, New York
- Teague J (1998) Personality type, career preference and implications for computer science recruitment and teaching ACSE98 Brisbane Australia