The Undergraduate Capstone Computing Project Experience

Donald McFall
University of Ulster, Northern Ireland

Abstract

This paper reports on the analysis of the assessment of 613 final year computing projects at the University of Ulster (Ulster). The analysis provides insight into the effectiveness of the teaching and assessment of computing projects and identifies shortcomings in the teaching and assessment processes. Potential causes and solutions are outlined. The paper also draws conclusions on topical project-related issues based on data drawn from peer review sessions that provide insight into the actual student experience and perception of project-related education.

1. Introduction

In the UK it is common practice for the final part of an honours degree in computing to include the assessment of a substantial individual project. This is a requirement of any computing degree programme accredited by the BCS -The Chartered Institute for IT [1]. The project typically involves the creation of a product that is some combination of software, service or process that is a result of a development process and that addresses a problem of interest in the subject area of computing systems. A main characteristic is that the project demonstrates the execution of a development process which, in almost all cases, results in the production of software. The students are expected to apply technical and professional skills which have been acquired from their course of study and consequently the project, at Ulster, is referred to as a 'capstone project'. Alignment with the expectations of the BCS has resulted in the specification for the project at Ulster being broadly similar to that in equivalent degree programmes accredited elsewhere by the BCS.

There are several computing degree programmes offered at Ulster across three schools. This paper focused on the projects of four computing programmes, three full-time and one part-time, offered at the Faculty of Computing and Engineering at Ulster between 2004 and 2012.

These projects were worth 30 credit points, implying an expected 300 hours of work for each student, and were executed over the first and second semesters of the final year of each student's programme. Potential supervisors and students proffered project proposals which were verified as compliant with University and BCS requirements. In some cases a student was expected to evolve a proposal with a member of staff. A mechanism for allocating projects proposed by staff that promoted equality of opportunity was employed. In the case of two students being equally in favour of one proposed project, the previous academic performance of the student was used to determine who would obtain the project. Supervisors were either associated with the selected proposal or were allocated randomly. Supervisors provided guidance in the management and technical execution of the project, provided meaningful and timely feedback, encouraged students to produce deliverables on time and to an appropriate level and participated fully in the assessment and moderation process.

Data from the assessment of 613 projects over an eight-year period has provided insight into the effectiveness of the teaching and assessment process.

Practice within the project module included peer review sessions where students were given an opportunity to reflect on topical project-related issues guided by the module tutor. The written feedback from these sessions included reflection on methodology usage, process understanding, effort hours, common practice and supervision. Empirical evidence on these topics was provided by forty-five peer groups representing the views of two hundred and twenty five students within the timeframe of the study.

2. The Problem

Anecdotal evidence suggested that the ability and experience of the supervisor had a distinct impact on the performance of the student in the execution of a project. Whilst it is the objective of an assessment and moderation process to identify and remove such anomalies it remained a possibility that inappropriate

supervision detrimentally affected student performance. Perhaps this is not surprising because a degree of variation exists in all teaching and assessment processes, but what was the actual variation and was it acceptable?

Supervisors and students often bonded in a working relationship over the duration of the project. There was anecdotal evidence that some supervisors were inclined to assess the work of their students leniently but to what extent was this true?

Both staff and students were permitted to propose projects. It was often the case that potential supervisors proposed projects based on their research interests and this was widely marketed as attractive and desirable because students were exposed to state-of-the-art concepts and ideas. Students often proposed projects based on their own hobbies and interests and these were molded into a formal proposal. All proposals were vetted and approved in a verification mechanism. But which, if any, of these methods was best for the student in terms of ultimate performance in the project?

Ulster offered the same capstone computing project, assessing the same learning outcomes using the same or very similar marking schemes, to students from different computing degrees. This was efficient in resources but the students employed slightly different technical skill-sets in the execution of the work. Did an overarching capstone project prove more suitable for one cohort of students than for others?

Anecdotal evidence suggested that matters of methodology and process usage were of little concern to students in the execution of their final year projects. Was this true and if so, why was a fundamental professional practice so misconstrued?

Notionally three hundred hours of effort was required for a student to complete the capstone project. But what actual effort was expended by the average student and was the effort expectation of three hundred hours appropriate?

These questions are familiar to many academics involved in the teaching and assessment of capstone projects [4] and whilst contextualized here by Ulster, they are common causes of concern across many universities.

3. The Method

The assessment profile, post moderation, for 613 capstone projects was analysed. The projects were drawn from four computing courses offered by the University of Ulster, Faculty of Computing and Engineering, over a period ranging from 2004 to 2012.

The capstone project at Ulster had four deliverables. An initial report was produced by the

student in the early stages of the project lifecycle. This initial report was in fact a brief project plan and was assessed by the student's supervisor. A presentation, pertaining to a technical description of the project and progress made, was delivered by the student at the midway point and was assessed by two members of staff. The supervisor was not normally involved in the assessment of the presentation. An interim report was produced by the student at the three-quarter mark and was assessed by the supervisor. This interim report was a project monitoring and control report and was a companion to the project plan. Lastly, a final report was produced by the student at the end of the project cycle and this was assessed by three members of staff including the student's supervisor.

The supervisors had a variety of backgrounds. For the purpose of this work they were categorised in two ways. Firstly, the supervisors were categorised according to their academic credentials and experience. The categories were Computer Scientist, Computing Associate, Engineer, and Engineering Associate. Those who were categorised as 'Engineer' were from a variety of engineering contexts but excluding software engineering. Those who were categorised as 'Computing Associate' were from a variety of backgrounds related to Computer Science; for example, multimedia experts. Those who were categorized as 'Engineering Associate' were from backgrounds somewhat related to engineering; for example, science specialists. Secondly, the supervisors were categorised as either research active or non-active.

The analysis of the dataset, including the assessment and peer review data, allows us to draw some conclusions regarding the basic questions under consideration.

4. The Results of the Assessment Analysis

Anecdotal evidence and indeed basic psychology suggests that supervisors have a bias towards their supervisee. Kahneman [2] implied that confirmation bias exists where people often seek data that are likely to be compatible with the beliefs they currently hold. Most supervisor-supervisee associations are positive, leading to the likelihood that where a supervisor has control over the performance of an assessment, the mark awarded will be positively biased.

Across all students there was little difference in the average marks awarded for the initial and interim reports: 67.9 and 67.1 respectively with a standard deviation of 10.66 and 12.49. Using the initial report then as a benchmark and comparing this with the average mark awarded for the presentation of 59.4, there was a considerable disparity between the two

figures of greater than 8 points. Similarly, the average mark of the final report of 59.1 when compared to the average mark of the interim report of 67.1 indicated a disparity of 8 points. The supervisor had sole responsibility for assessment of the initial and interim reports. In the case of the presentation, the supervisor had no role in assessment. A team of three, including the supervisor, was involved in the assessment of the final report. Figure 1a illustrates for each of the fifty-one supervisors the average of the marks they awarded supervisees for the initial report and separately the average marks earned by the supervisees in the presentation.

Figure 1a. Initial report and presentation marks

Figure 1b illustrates for each of the fifty-one supervisors, the average of the marks they awarded supervisees for the interim report and separately the average marks earned by the supervisees in the final report. The disparity is evident in each case.

Figure 1b. Interim report and final report marks

On a closer look at individual supervisors, it was the case that six of the fifty-one supervisors had a standard deviation on the initial report of less than 5. In the case of one supervisor, a standard deviation of 1.8 in the initial report marks and a high average of 76 (the highest in the population) indicated empowerment by the supervisor of advantage to the supervisees.

The better performance by supervisees in the initial and interim reports may have a number of explanations.

Project Management, which lies at the heart of the initial and interim reports, may have been better understood by students than the technical subject of the final report. Although this may have been true in some

isolated cases, there was no evidence to support this proposition. Indeed, a cursory examination of the average mark of students in project management modules yielded an outcome closer to a 60% average, indicating a performance much closer to the presentation and final report average marks than to those of the initial and interim reports. Students then had no special understanding of project management topics.

The initial report may have been less challenging to complete. It was certainly less demanding in terms of its scope compared to the final report and consequently the capacity for differentiation was less.

The supervisors may have been less qualified to understand the nature of the initial and interim reports and consequently the assessment performed on them was less rigorous. Few of the supervisors have expertise in this area.

With a 10% weighting for the initial and interim reports the grade inflation caused by this anomaly was small. However, in the case of some supervisors who awarded high marks, the inflationary effect was of course proportionally greater; however, this evidence is only suggestive of bias: the bias had a small inflationary effect on the overall marks awarded.

The more important question relates to whether some supervisors enabled their supervisees to perform better in the assessment process independent of any bias. In other words, if we remove the biased initial and interim reports, is there evidence that the marks attained by the students of some supervisors were markedly better or worse than those of others, mirroring the ability of the supervisor?

Figure 2 shows the average final report marks produced by fifty-one supervisors, representing agreed final report marks from a panel of three assessors, blind marking the final report component. There was an absence of bias ensured by the assessment involving three markers, and collection of the data over time ensured that the differing ability of the supervisees was somewhat normalised.

There was considerable diversity in the average performance of supervisees over the range of supervisors. The bottom five supervisors had an assignable average final report mark of 49.8% for the group of five whereas the top five supervisors had an assignable average final report mark of 68.8 for the group of five. The population average was 59.1%. The range for the average final report marks was 28.3.

Only one supervisor in the bottom five was a Computer Scientist, while three were designated as 'Engineering Associate' and one was 'Computing Associate'. No 'Engineering Associate' appeared in the top five. The same five supervisors featured in the bottom seven when the average of the presentation
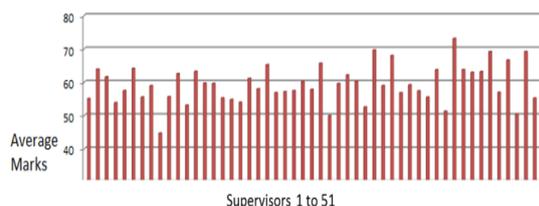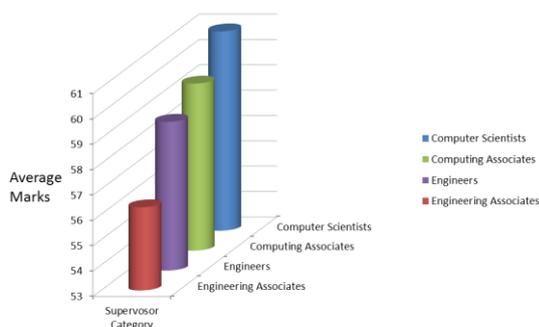
Figure 2. Supervisee average final report marks for supervisors

marks was analysed.

A comparison of the average final report marks produced by supervisor background is shown in Figure 3. Classification of the supervisor profile was based on human judgment utilising experience and knowledge of the individuals involved. Alternative opinions produced slightly different categorisations whereby a number of supervisors were classified differently, but this did not change the order of information presented in Figure 3.

One explanation for the clear distinction in performance here was the credentials and experience of the supervisor. The computing capstone project demanded that the supervisor provided technical guidance based on a software development lifecycle. Those supervisors with a computing science background had software process-related knowledge and skills more appropriate to the supervision of a computing science project. This is perhaps not surprising but nevertheless it had implications for the performance of students.

**Figure 3. Average final report marks for supervisor category**

The answer to the question whether some supervisors consistently enabled supervisees to perform better in the assessment process independent of any bias, is clearly 'yes': some supervisors were simply better at supervision than others, and the experiential profile of the supervisor had an impact. With 613 projects conducted by 51 supervisors and

providing an average sample size of only 12, at the level of individual supervisors there was insufficient data to draw any firm conclusions regarding individuals and so Figure. 1 should be viewed simply as a dashboard for the monitoring of performance and provision of early alerts. However, when the data was aggregated up to the level of supervisor category (Figure 3) the sample size became sufficient and the evidence became much more compelling.

An alternative way to view the distribution of marks is based on whether the student undertook a staff proposed project or not. Staff-proposed projects were likely to encompass the research interests of the proposer. There was less available data on this topic. However, in 125 cases data is available. Where students had undertaken their own proposed projects the average final report mark for this group was 61.4. The average final report mark for the group of students who undertook a staff-proposed project was 58.1. Obviously there may be a greater degree of motivation demonstrated by a student who undertakes his or her own project and this might be sufficient to explain such disparity. Many projects proposed by potential supervisors were based on the research interests of the proposer. At undergraduate level, it is the execution of an engineering process that should be at the heart of a proposed project. The BCS guidelilnes [1] make clear that projects should reflect the aims and learning outcomes which characterize a programme. Rowles et al [14] suggested that the primary focus of capstone project assessment should be on application of previously acquired knowledge. Since an engineering process and a research process are two distinct entities, the former being an important computing paradigm, might it be the case that staff proposals based on research interests are less appropriate at undergraduate level in computing and lead students into unchartered and less well understood directions? They might in fact distract a student's attention away from the important development process. It is certainly not the case that undertaking a staff-proposed project offers a student any particular advantage.

There is always a group of students who fail to propose their own projects and are unsuccessful in obtaining a staff-proposed project. In these cases, a student and supervisor evolved a hybrid proposal. Evidence of the performance of these students in the execution of hybrid projects was limited to a population of 29 students but the average performance in the final report for this group was significantly low at 52.6. Perhaps this is not surprising since the allocation of project proposals was based on the student performance in second year and favoured better performing students. Since year two performance is highly correlated with final report

performance (see Figure 4) this result might be expected. It is difficult to isolate the impact of the failure to obtain a proposal, thereby having to undertake a hybrid project, separately from the effect of low year two performance but there is perhaps a moral prerogative which should prevent this type of allocation; accessibility of proposals should be the same irrespective of past performance. Peer review evidence suggested that 63% of students were less than satisfied with the allocation process.
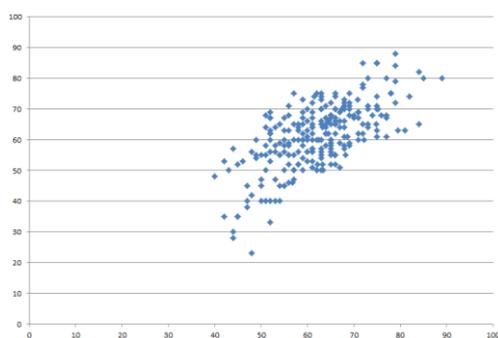
Figure 4. Correlation between year 2 average marks (y) and final report average marks (x) (Correlation = 0.68)

5. The Results of the Peer Review Analysis

The use of methodologies was assessed in the peer review data. One question asked during peer review pertained to the adherence to a methodology irrespective of the type of method adopted.

Students were asked to assess the value of their adopted methodology in terms of its utility within the project on a scale from 5 (highly valuable) to 1 (not valuable) in the fashion of a bipolar dimension of appraisal. The bar chart of Figure 5 shows the results from 100 respondents.

The majority of students clearly did not obtain value from the methodology they adopted. 57 students out of 100 (57%) rated the value in terms of utility as low and only 15 students from 100 (15%) rated the value of a methodology as high. This contrasts with the findings of Joy [3] where 57% claimed to use the project methodology a lot.

The disparity might be explained by the lack of a proper definition of the semantics of 'using a lot', where the label might mean many different things. The findings at Ulster were more precise in that the issue of proper usage lay at the heart of the question. The Ulster data suggested that there was an immature approach to the adoption of methods within final year projects.

Anecdotal evidence and experience would imply that methodology was retrofitted by students for the final report and after the product was produced. This explains, at least in part, the time pressure experienced by students close to submission dates. Students have often demonstrated, during peer review, a superficial understanding of the application of methodology in the project. They identified a particular lifecycle quite easily but were often ignorant of the goals and activities of the various stages of a lifecycle, lacking the appropriate software process knowledge; application of a methodology then became problematical.
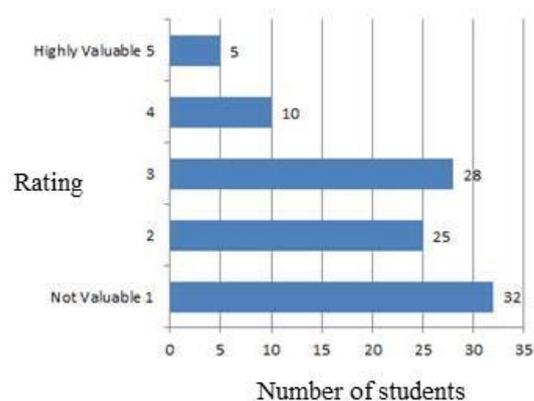
Figure 5. Value rating for methodology (n=100)

There is confusion among project students in general regarding the difference between a lifecycle, a methodology and a process. The literature does not help much with myriad plausible definitions. It can be said that a life-cycle model is a more general term for a category of methodologies and a software development process a more specific term to refer to a specific process chosen by a specific organization. Students were asked to identify the lifecycle methodology they made most use of in their project and the potential responses were unconstrained. The responses are shown in Figure 6.

Interestingly, when students were asked about the problems they encountered in using lifecycle methodologies they often commented that they did not really use a methodology (a factor noted in the previous discussion of Figure 5) but this was not reflected in the responses of Figure 6 to any great extent. Further discussion in peer group activity uncovered that the major issue was one of ignorance of the actual tasks to be performed within the stages of the lifecycle methodology. The popularity of the Waterfall and its variants was its relative simplicity. The loose structure of the Waterfall approach enabled students to contrive tasks to be performed at each stage and this was further driven by dubious advice offered

by supervisors. The identification of CMMI® [19] as a methodology by some students belies the fact that the CMMI® is a body of knowledge and model of processes and not a process itself. Interestingly the students who utilized CMMI® all produced final reports in the first class category. The advantage offered by CMMI® was the clear identification of the goals and practices of the engineering processes most useful for software development; filling in the gap in the students' understanding of lifecycle tasks. Since the Waterfall paradigm is non-descriptive regarding practice, those students who adopted a simple Waterfall paradigm were consequently denied the practice knowledge necessary to implement a method.
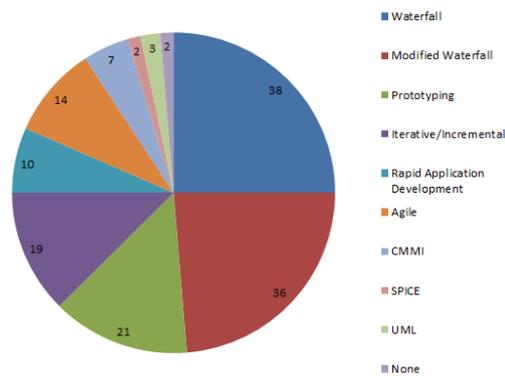

Figure 6. Type of methodology utilised (n=152)

Agile processes, including Kanban and Scrum, have been growing in popularity. These processes encourage the sacrifice of large work items in favour of smaller work items which can pass through a work flow much more quickly thereby delivering value to the customer in a shorter timeframe. They forgo the need to perform detailed estimation up front in favour of better understood team velocity later in the project; generally the focus is away from documentation and more on progress. Examination of 10 of the 14 projects where an Agile approach was adopted revealed the same problem as that experienced by students who utilized the Waterfall Lifecycle, lack of practice understanding. These projects utilized the concepts of 'lean' to identify small work items and the workflow cycle through which the items would progress but neglected the task details which are required as entry-exit criteria for each cycle stage. The Agile Manifesto [20], among other desires, claims to value individuals and interactions over processes and tools. This can be viewed by project students quite literally. In reality the Agile Manifesto makes it clear that there is value in process and clearly the practice level knowledge

required by project students lies in a proper appreciation of the process and the tasks which should be defined for each column in any visual representation within Scrum or Kanban. These Agile approaches then favour visibility and work flow over process and task understanding but it is the latter that is the appropriate focus in the educational context of a Capstone Computing Project; it is the latter which espouses the former. Since the timeframe of the study Agile has continued to grow in popularity among students who consequently fail to grasp the importance of process in the learning context.

Capstone project guidelines on supervision at Ulster required that a supervisor was normally contactable weekly during the period of the project and that a student contact the supervisor frequently throughout the duration of the project. Data from the peer review question set was extracted whilst students were still completing projects. Students were asked to rate satisfaction with the supervisory experience on a scale from 5 (highly satisfied) to 1 (not satisfied) and to identify the number of hours expended in face-to-face contact with their supervisor. Figure 7 shows the contribution made by 103 students for each response on contact hours and Figure 8 shows the contribution of each student response on supervision satisfaction.
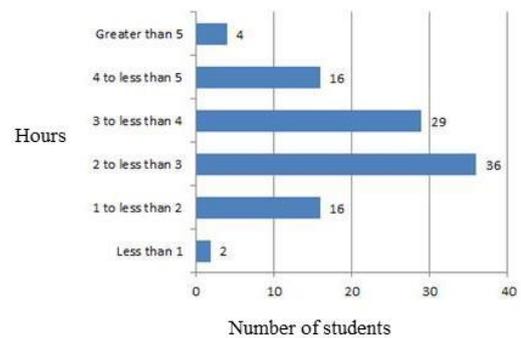

Figure 7. Hours contact time (n=103)

It is difficult to gauge what reasonable face-to-face contact time between a supervisor and supervisee should be. Anecdotal evidence suggests that good students often need less time with their supervisors and are self-motivated and maybe even content with contact through other means such as email or social media: however, all of the five students reporting contact time of greater than five hours had final report marks in the first class range. Weaker students also tend to spend less time at such meetings choosing to avoid what can be challenging expectations. Eleven of the 18 students who reported spending 2 or less hours in face-to-face contact with their supervisor went on to

obtain marks in the third class range for their final reports. None of this group of 18 students obtained a final report mark in the first class range. Notable here is the relationship between performance and attendance revealed by Romer [18] who discovered that good attendance was rewarded by better performance among students.
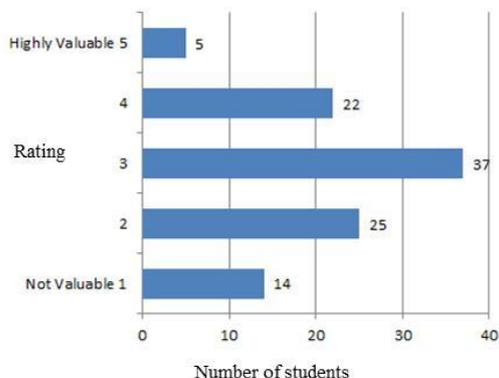

Figure 8. Value rating for supervisor (n=103)

It has been recognized [12] that "a project-oriented capstone course is frequently loosely supervised." The data (Figure 8) suggest that 14 students were not satisfied with the supervision they received and 10 of these students went on to receive a final report mark in the third class category. Five students were very satisfied with the supervision they received and 4 of these students went on to receive a final report mark in the first class category.

Undoubtedly, weak students attend fewer supervisory meetings and also report dissatisfaction with supervision. Good students attend more supervisory meetings and also report satisfaction with supervision. It is difficult to determine cause and effect here, so no firm conclusion is drawn from the findings.

Joy [3] suggested that students expend more time on average (369 hours) than the overall effort expected (300 hours) by his institution. After normalisation by removal of outliers, the effort hours cited by Joy is 307 hours.

The main body of our peer review sessions ended at the midway point in the execution of the project so data available here is restricted to a smaller number of students (population 48) who reported the effort hours subsequent to completion of their work and prior to assessment. The average effort hours reported is 282 hours. The result shows that the assumption of a 300-hour effort for the 30-point capstone project is broadly accurate.

A select number of 17 students recorded data in a log to estimate the effort expended on various predefined lifecycle activities as a proportion of the total hours expended. These 17 students formed a subgroup of the peer review students whose activities were closely monitored by their supervisor. Figure 9 represents the distribution of the recorded hours over the project activities.

The average of the total effort for the 17 students was 306.94 hours. The most significant proportion of time was spent on implementation (average 102 hours) and write-up (average 65.7 hours). The total hours for these 17 students align significantly with the 307 total effort hours identified by Joy [3]. The small case study here represents data that was carefully logged whilst Joy's data is drawn from student estimates in questionnaires. The detailed log kept by the 17 students in this case study forms an accurate tally and distribution of the effort hours expended on the various process activities in the project.
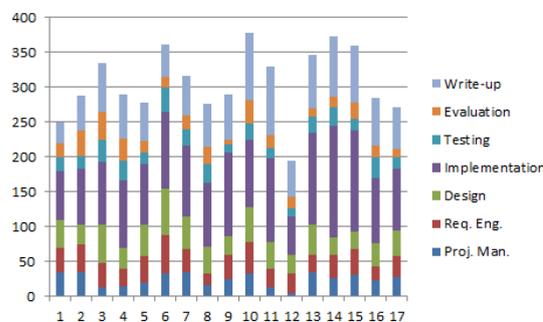

Figure 9. Distribution of hours over project activities

During peer review students were asked to critique the assessment schemes used in the assessment of the final report. The most surprising outcome of the critique was that many students wished for effort to be recognized and included in the assessment. This finding concurs with Tippin [15] who suggests that students judge professors as unfair when perceived effort invested in the completion of an assignment does not compensate for actual poor performance.

Finally, there appeared to be a differential in the performance of students from the three full-time courses. Figure 10 shows that students from course A perform less well on average. This may be due to differing entry criteria applied to applicants for each course; however, the skill-set imbued on students in course A is somewhat different from that imbued on students in the other courses B and C even though the capstone project assessment criteria were very similar.
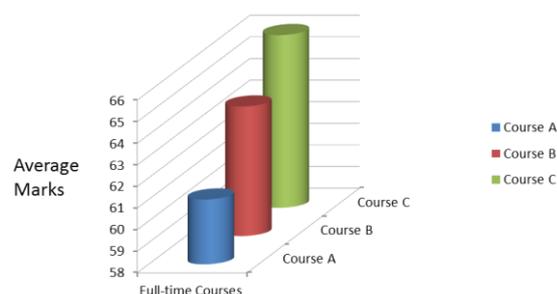
Figure 10. Average final report marks for course

6. Solutions

The results were sufficient to have raised concerns in a number of areas associated with the teaching and assessment of final year capstone projects.

Variation in marking is often higher for capstone project components than for other work elsewhere [11]. The capstone project is different from a normal taught module. The latter is taught by a member of staff whose credentials are aligned with the requirements of the module. The former can be supervised by a variety of staff, not all of whom approach the task with the same experience or knowledge set. This had an impact on the nature of the supervision offered to students. It would be appropriate in this case to have a programme of professional development tailored to help staff in the supervisory role. Marshall [5] suggests that academics supervising students undertaking final year projects need support and advice for undertaking a role which is significantly different from other forms of teaching. Furthermore, to cater for the growth in student numbers, Healy et al [4] has advised that differing supervisory relationships be built into our courses.

Although resources would hinder the prospect, it might be beneficial to restrict the number of staff undertaking the supervisory role to those best qualified. Alternatively a team-based approach might be adopted whereby specialised staff operates drop-in centres on topics such as programming, requirements development, testing, and so on. Students would then be able to avail of this expertise; where a project coordinator provides advice on general project housekeeping, this might ultimately remove the need for an assigned supervisor. Students as partners in learning [10] offer a potential for students to influence the assessment of projects so that the assessment criteria are selectively aligned with the particular skills of the supervisor and supervisee. This would mandate a supervisor-supervisee assessment contract permitting the tailoring of a standard assessment tool. Although a form of asset control might be needed to construct and maintain the basic assets, the potential for more mature associations would be greatly enhanced.

It appears that 'supervisor' assessed components offered an opportunity for lenient marking, with both initial and interim reports attracting higher marks than other components of project assessment. This was not detected in module evaluation, which only dealt with in-year profiles. It would seem reasonable to remove the supervisor from the role of independently marking the work of the supervisee or at least introduce a more robust moderation process. The Swedish National Agency for Higher Education [6], for example, advises that an external quality panel of experts is used in the assessment. The simple initial and interim reports might also be reengineered to offer a greater challenge to students.

It is clear that failure to understand the practice level knowledge that underpins the lifecycle is a major hindrance to the appropriate use of methodologies. This is a failure in the teaching of computing and should be straightforward to reconcile. For many final year projects across various disciplines and universities students do not appear to have been adequately prepared [7]. Emphasis on software process is needed during course design so that students can better appreciate its role during project execution. Some taught modules should be prerequisites in support of this aim, as opposed to a traditional autonomous approach which makes it difficult for students to see the relevance of the software process to the project. Another approach may be to allow elements of the project mind-set to be developed in early years through the use of self-reflection on project-related processes, an essential part of preparation [8]. A specialized component of assessment might be adopted to provide formative feedback on the supervisee's knowledge and understanding of the software process [13]. Some institutions in the U.S.A have incorporated the Major Field Test, a national standardized examination, to test the students' knowledge of fundamental project-related processes prior to commencement of the capstone project [12].

The effort hours expended on the project is broadly in line with our expectations. The large number of hours spent on implementation and write-up offer an opportunity to direct faculty support to these activities through the adoption of drop-in centres to support both programming and technical writing. Ulster students believed that up to 10% of the marks might be awarded for effort. This of course is strewn with problems regarding the accurate estimation of effort and its significance. However, Ulster specifically identifies an expected level of effort as 10 hours per credit point. At 30 credit points the capstone project is expected to take on average 300 hours of work. Given that student

attendance at classes and meetings was cited as a significant problem by Muir [16], there is recognition among students that at least attendance, associated with effort, should be rewarded. Muir suggests *"there is a tendency for students who attend classes more regularly to gain better marks."* Muir's study reveals *"there is a real sense of injustice amongst some students that poor attenders should still be able to obtain their degree, even if they do less well than those who attend regularly."* Marburger [17] discovered that, when absenteeism was penalized, attendance was higher. Despite the obvious problems associated with assessing effort perhaps assessment of capstone projects should include a percentage of marks for regular attendance at supervisory meetings and participation in peer group activity; positive reward rather than negative penalty. In-house studies at Ulster have shown that a regular attendance pattern of 64% in one module was improved to 83% with just the small addition of 3% to coursework marks for attending 80% or more of the classes. Although weak students have a poor record of attendance at supervisory meetings it seems clear that the demands of a final year capstone project place such a burden on students that it is the instrument which is most likely to expose any underlying weakness. Furthermore, frameworks which facilitate a structured conversation between students and their supervisors are available [9] and these reduce misunderstandings between a supervisor and supervisee on the nature of the final year project and can be used to set attendance requirements. If Muir is correct, improving the attendance of students at project-related meetings, classes and other activities will improve project performance.

The approach of allocating project proposals must ensure that every student has an equal opportunity to obtain the proposal of their choice irrespective of previous performance. Systems of distribution which use previous performance isolate the weaker students who then proceed to develop hybrid proposals without having any idea initially what will be the focus of their project. This wastes initial time while the student and supervisor try to expedite a hybrid proposal. In any event, the fees paid by students for higher education should mandate such an approach. Students are more motivated when they propose their own projects so this should be encouraged. There is no advantage gained by a student in pursuing a staff-proposed project, so students should be facilitated in understanding how to create their own proposals, in the months preceding the start of the final year. There are various ways in which this might be achieved but professional development modules, a feature of modern computing degrees in the earlier years, might be the most appropriate placeholder for such activity.

The one-size-fits-all approach adopted by many universities in fitting one capstone project assessment for a range of courses is inadvisable. When skillsets vary, so too should the nature of the final project assessment.

7. Conclusion

This work has many limitations. A useful data set of over 600 assessments allows some degree of normalisation but the student count for individual supervisors is an insufficient sample. This is mitigated by aggregating up to the level of supervisor category but it would be useful to expand the data so that conclusions could be drawn with regard to individual supervisors.

It would be useful also to replicate the results in another similar data set in another institution where academics from a variety of backgrounds supervise final year Capstone Computing projects.

The peer review data is obtained from relatively informal and voluntary contributions made by over two hundred students over a significant time period. It is not clear that the profile of these students is representative. This said it is still apparent that the main findings are strongly supported by evidence.

University faculties offering computing science degree courses would be advised to allow each computing degree course to sustain its very own capstone computing project together with its own unique learning outcomes. Faculties should encourage students to create and execute their own project proposals. Faculties should monitor attendance and the outcome of this should be incorporated into the assessment of the capstone computing project. Faculties should ensure that supervisors are well trained to handle the supervisory role or alternatively assessment of capstone computing projects should be confined to a select group of individuals possessing the necessary skills. Software process education is crucial for the successful execution of the project. This should be better taught so that students appreciate the nuances of lifecycle paradigms. Software process should be better integrated into the curriculum as a supporting device for project execution so that the value and relationship to the capstone project is clearly visible. The role of supervisors in the assessment of capstone computing projects should be better monitored and controlled.

In order to improve the experience of capstone computing project students, Ulster has implemented many of the recommendations made here during its cyclical revision of courses in 2013. A future review will investigate the appropriateness and success of these solutions and enable validation of these

recommendations.

8. Acknowledgment

The valuable insight into undergraduate final year projects provided by Healey et al [4] and the use of associated references is acknowledged.

"Final-year projects and dissertations (FYPD) undertaken by students at the end of their Bachelor degree courses are a topic of current interest in many countries. It is timely to reassert the importance of FYPD and to rethink their role in the curriculum as the context of higher education changes." (4).

9. References

[1] BCS, Guidelines on Course Accreditation (September 2010, updated for use from Autumn 2012).

[2] D. Kahneman, Thinking, Fast and Slow, Penguin (0100) ASIN: B00DO94MO4, 2012.

[3] M Joy, (2009), "Undergraduate Computing Projects – an investigation into the student experience," in ITALICS Volume 8 Issue 1 February 2009 ISSN: 1473-7507, 2009.

[4] Mick Healey, Laura Lannin, Arran Stibbe and James Derounian, *Developing and enhancing undergraduate final-year projects and dissertations*, https://www.heacademy.ac.uk/sites/default/files/resources/NTFS_Project_Gloucestershire_2010_final_0.pdf, 2013 (Access Date 1st Nov 2014).

[5] S Marshall, "Supervising projects and dissertations." in Fry, H., Ketteridge, S. and Marshall, S. (Eds.) *Handbook for teaching and learning in higher education*. (pp.150-165). Abingdon: Routledge. 2009.

[6] Swedish National Agency for Higher Education, "The Swedish National Agency for Higher Education's quality evaluation system 2011–2014. Stockholm:SNAHE", http://www.hsv.se/download/18.328ff76512e968468bc80004249/1103R-quality-evaluation-system-2011-2014.pdf , 2011 (Access Date 3rd Nov 2014).

[7] M. Todd, P. Bannister, S. Clegg, "Independent inquiry and the undergraduate dissertation: Perceptions and experiences of final-year social science students," in *Assessment & Evaluation in Higher Education*. 29 (3), pp. 335-355. 2004.

[8] P. Greenbank, C. Penketh, C, "Student autonomy and reflections on researching and writing the undergraduate dissertation," *Journal of Further and Higher Education*. 33 (4), pp. 463-472. 2009.

[9] M Luck, M. "Undergraduate research projects: A flexible wireframe for success." Poster: *International Society for the Scholarship of Teaching and Learning Annual Conference*, Hamilton, Canada: 24-27 October. 2012.

[10] Mick Healey, Abbi Flint and Kathy Harrington, "Engagement through partnership: students as partners in learning and teaching in higher education," *HEA*, https://www.heacademy.ac.uk/sites/default/files/resources/NTFS_Project_Gloucestershire_2010_final_0.pdf (Access Date 21st Oct 2014).

[11] Webster, F., Pepper, D., Jenkins, A. "Assessing the undergraduate dissertation," *Assessment & Evaluation in Higher Education*. 25 (1), pp. 71-80. 2001.

[12] J. Mingxian, "Redesign of computer science capstone course by integrating the major field test (MFT)." *Journal of Computing Sciences in Colleges*, Volume 24, Issue 1. Pp 239-246. 2008.

[13] Heylings, D.J.A. and Tariq, V.N. "Reflection and feedback on learning: a strategy for undergraduate research project work." *Assessment & Evaluation in Higher Education.* 26 (2), pp. 153-165. 2001.

[14] Rowles, C., Koch, D., Hundley, S. and Hamilton, S. "Towards a model for capstone experiences: Mountaintops, magnets, and mandates." *Progress, Trends and Practices in Higher Education*. 16 (1), pp. 13-15 2004.

[15] Tippin, G. K., Lafreniere, K. D., Page, S. "Student Perception of Academic Grading: Personality, Academic Orientation and Effort." *Active Learning in Higher education,* 13(1), pp. 51-61, 2012.

[16] Muir, J. "Student Attendance: Is it Important, and What Do students Think?" *CEBE Transactions*, Vol 6, Issue 2, pp. 50-69, 2009

[17] Marburger, D. "Does Mandatory Attendance Improve Student Performance?", *Journal of Economic Education,* 37 (2), pp148-155, 2006.

[18] Romer, D. "Do Students Go To Class? Should They? *Journal of Economic Perspectives*, 7 (3), pp167-174, 1993.

[19] Mary Beth Chrissis, Mike Conrad, Sandy Shrum, "*CMMI for Development Guidelines for Process Integration and Product Development*", Addison-Wesley Professional; 3 edition, 2011.

[20] M.A.S.C.Manamendra, K.N.Manathunga, K.H.D. Perera and N.Kodagoda, "*Improvements for agile manifesto that make agile applicable for undergraduate research projects*", The 8th International Conference on Computer Science & Education, Colombo, Sri Lanka (ICCSE 2013) April 26-28, 2013.