ICT student projects: assessing students engaged in the community

Ms Kathleen Keogh
School of Information Technology and Mathematical Sciences
University of Ballarat
k.keogh@ballarat.edu.au

Ms Anne Venables
School of Computer Science and Mathematics
Victoria University

Abstract

To perform successfully in the workplace, employers demand that a new ICT graduate should be equipped with a strong set of technical skills, coupled with a flexible and creative approach to problem solving (Dawson, 2000; Hagan, 2004). To prepare students for their transition to industry, many tertiary institutions incorporate a team project into the final year of their IT and Computing courses (Clear, Young, Goldweber, Leidig, & Scott, 2001). Such projects include diverse technical areas including software development on specialised platforms, hardware design, networking problems and multimedia development.

The importance of a software team project in final year computing degrees is without argument (Newman, Daniels, & Faulkner, 2003; Lynch, Goold, & Blain, 2004); however, there is academic debate about how to best achieve 'industrial strength' authenticity for students. Should the project problem be one constructed by academic staff? Or should the problem be that of a real client? In this paper we argue the latter case by describing our experiences drawn from supervising student ICT projects done for real clients in the local community across three different institutions. An assessment model for student projects that successfully promotes the development of good graduate attributes and the development of professional processes is detailed. The model includes explicit scaffolding and support structures to aid students engaged in real-life (and loosely defined) software projects. We argue that assessment of project work is about valuing process as much as product; we also report upon lessons learned from implementing this model at different institutions.

Keywords

assessment, ICT student projects, real clients, community engagement

Introduction

Community and government expectations are that undergraduate degrees should prepare graduates for the workforce; to this end, Australian universities are required to report graduate attributes of exiting students in university accreditation audits (Toleman, Roberts, & Ryan, 2004). Employers demand that a new ICT[1] graduate should be equipped with strong technical skills and a flexible and creative approach to problem solving (Dawson, 2000; Snoke & Underwood, 2001; Hagan, 2004). To prepare students for their transition to industry, many tertiary institutions incorporate a team project into the final year of their IT and computing courses (Clear et al., 2001). Such projects encompass diverse technical areas including software development on specialised platforms, hardware design, networking problems and multimedia development.

In the next section, we argue that students should be given projects for real clients. We describe our experiences drawn from supervising student ICT projects for real clients in the local community across three different institutional settings. We then describe our assessment model for student projects. Our model successfully promotes the development of good graduate attributes and the development of professional processes. It includes explicit scaffolding and support structures to aid students engaged in real life (and therefore often loosely defined) software projects.

[1] Information and Communication Technology

We examine the benefits for students when they are given the opportunity to engage with a real client within a 'protected' educational setting, and discuss some of the issues for the academic staff who provide supervision and support. We argue that project work is as much about 'process' as 'product'. Finally, we report upon lessons learned across institutions and suggest that our approach could be adapted for ICT degree programs in various settings.

Community-based projects for real clients

Many institutions pursue work-based projects, whether authentic or in simulated learning settings (Bennett, Dunne, & Carre 1999; Cranmer, 2006) as a mechanism to improve their overall graduate outcomes because projects engage students in problem-based and experiential learning. Through projects, students become active participants in their own learning, constructing their own internal knowledge through experiences of the social and physical context of their work (Kolb & Kolb, 2005; Greening, Kay, Kingston, & Crawford, 1997; Newman et al., 2003). By exposing students to collaborative group work, software development projects allow students to assume responsibility and develop their collaboration and communication skills. Final year computing projects foster an environment where students can develop more generic attributes, such as time management, problem-solving skills and teamwork. Student projects have been associated with changes in student attitudes and an appreciation of industry ethics and professional perspectives (Lynch et al., 2004; Newman et al., 2003; McLay, Corich, & Millma, 2005).

There continues much discussion in the computing education literature about how best to provide 'industrial strength' authenticity for students undertaking software development projects. For staff organising projects, there are many factors to consider and options to try (Clear et al., 2001). Commitment to a specific delivery model or time and administrative constraints may sway lecturers to opt for a tightly constructed 'known' problem. Alternatively, others commit to servicing the needs of real world clients who have ICT problems that might be unconstrained and loosely defined (Fleming, 2005; Lynch et al., 2004; Keogh, Sterling, & Venables, 2007). Students are more motivated when faced with authentic tasks in a realistic setting where the client is external to the institution (Bothe, 2001; Krause, 2006).

Dawson (2000) strongly advocates — as we do — that learning through realistic experience for real clients with ill-structured problems is more beneficial for students than with well-defined and carefully planned projects. Our experience across three very different campuses — a regional Australian university, a small metropolitan university and a small teaching institute (teaching overseas students, partnered with an Australian university) — has been that community-based problems are very engaging and motivating for students. Some student project examples include: legacy accounting logistic software, a music recording interface for users with disabilities, a content management system, a club website/membership database, an e-commerce website, educational games, an online restaurant booking system, a virtual chemistry lab and a virtual desktop primary school network.

Students value the opportunity to work with real clients, even though this introduces a degree of unpredictability. In one survey[2], 19 out of 20 students answered *yes* to the statement 'It has been great dealing with a real client' despite nine of the same 20 students also replying *yes* when asked if they had experienced difficulties working with a real client. Running software development projects for community-based clients creates many teaching challenges. Careful selection needs to be made when deciding on projects and in identifying suitable clients for whom students can work.

Choice of project

Selecting projects for students to encounter can shape the student experience (for example, Fleming, 2005; Dawson, 2000) so care must be taken. Projects must be viable, have resources available and be of an appropriate size (Daniels et al., 2002). Ideally, it is best if they can be scaled up or down depending on the abilities, skills and experiences of the student team. The scope achievable throughout the project needs to be balanced with the amount of self-learning required by students to achieve a solution.

[2] End of Project 1 student survey, July 2006, University of Ballarat.

Developing the confidence to engage in independent learning of new technologies is a desirable graduate quality (Nunan, 1999). Previous to the project, students learning tasks are often limited to small, constrained assignments that are designed to reduce risk. The project offers an alternative broader experience. With staff guidance, the student team needs to scope the project and, in liaison with their client, prioritise requirements to implement in the final solution.

The main goal, from the students' perspective, is to succeed in delivery of a system that they have scoped, researched, designed and implemented for their client. They come to understand the role of analysis. Eliciting client requirements is often a difficult task that takes time and is often underestimated by students. Students learn the valuable lesson that effort and good planning are required to manage client relationships.

'Good' clients

It is important that the student project is not critical to the client (Garlan, Gluch, & Tomayko, 1997) so that students, whilst motivated to produce a product for a real client, are free to learn by making mistakes. Clients are made aware that our educational aims focus on the process rather than product, and these aims are achieved during the project experience (including making mistakes!). At worst, clients are guaranteed to receive good documentation describing their requirements and it is likely they will receive a working prototype that implements at least some of the core requirements. The most important qualities needed in the client are supportiveness and willingness not to interfere with the students' processes. Still, clients behave as typical clients and many clients can be initially vague or change their minds about requirements. Ideally, the requirements specification is signed-off half way through the project. If there are significant client difficulties, it is possible for a staff member to resolve these (if necessary, becoming a pseudo client) without jeopardising overall project success.

Assessment model

Studies in the US, New Zealand and Australia attest that a growing number of undergraduate students are motivated by pragmatic mechanisms by which to fulfil course requirements with the minimum of effort (Wilhelm & Comegys, 2004; Ditcher & Hunter, 2000; Venables, Tan, Devi Nagappan, & Ghous, 2006). The Centre for the Study of Higher Education notes:

> For most students, assessment requirements literally define the curriculum … Assessment is therefore a potent strategic tool for educators with which to spell out the learning that will be rewarded and to guide students into effective approaches to study.
>
> (CSHE, 2007)

As educators, we are interested in engaging our students and encouraging them through the experience of projects to construct their own knowledge and become self-regulating learners (Nicol & MacFarlane-Dick, 2006). We use formative assessment through feedback to promote student progress, to encourage appreciation of industry ethics and professional perspectives. We place a significant assessment emphasis on rewarding team communication and management processes with product outcomes.

Assessment structure

Our assessment structure comprises two main categories: process and outputs (product). Assessment guidelines explicitly detailing the criteria and percentages for each aspect of the project are released early in the course to ensure that students have a timeline and framework for what is expected of them. As a typical example of our practice, the assessment structure for a two semester project unit conducted at the University of Ballarat is shown in Table 1. Here, the first semester's assessment is split equally: 50% process, 50% output. The second semester assessment is weighted slightly more towards outputs to account for the significant effort students put toward the implementation (coding) of the solution. Analyses of other project units at Victoria University and at the International Institute of Business and Information Technology reflect similar emphasises on processes and output.

Table 1: Assessment tasks for a two semester project

Assessment task	Release date	Due date	%Weight
Semester 1			
Tender bid or project proposal	Week 1	Week 2	5%
Project Management Plan (PMP) version 1, project plan, WBS[3]	Week 1	Week 3	10%
Team processes, project management, maintenance of PMP	Week 1	Ongoing assessment	8 %
Requirements elicitation, client liaison	Week 1	Ongoing assessment	10%
Weekly timesheets, status reports, maintenance of project plan	Week 1	Ongoing assessment	7%
Teamwork, meeting attendance, supervisor meetings, initiative	Week 1	Ongoing assessment	8%
Requirements document (including analysis)	Week 1	Week 5 – draft 1, Week 8 – draft 2.	10%
Presentation: team management, organisation and project overview	Week 1	Week 9	15%
Assessment task	**Release date**	**Due date**	**%Weight**
Working prototype	Week 1	Week 10	10%
Design Document(version 1)	Week 3	Week 11	10%
Signed-off requirements, design decision-logs, meeting minutes, research report	Week 1	Week 12	7%
Semester 2			
Detailed design document (version 2)	Week 1	Week 3	8 %
Maintain documents, traceability matrix	Week 1	Ongoing assessment	5%
Project management, regular updates to project plan, status reports, timesheets, WBS	Week 1	Ongoing assessment	10%
Teamwork, meeting attendance, supervisor meetings, initiative	Week 1	Ongoing assessment	10%
Poster presentation on design	Week 1	Week 5	5%
Individual review of another team's documentation	Week 1	Week 8	5%
Test plan	Week 1	Week 9	9 %
Final software implementation, test reports	Week 1	Week 12	20%
User documents, client sign-off: delivery	Week 1	Week 12	8%
Poster presentation (marketing, demo final product)	Week 1	Week 12	10%
Final individual interview with staff	Week 1	Week 13	10%

Process includes quality assurance, management, attendance, communication and evidence of good practice: meeting minutes, timesheets, diary, electronic document repository and status reports. The assessment scheme motivates students to value process; initially perhaps for marks, but eventually we hope students appreciate the value of adopting good process.

Students initially submit a tender bid for their preferred project, outlining their skills appropriate to the project. A team project management plan (PMP) that states team rules, processes, roles responsibilities, standards and a project plan is required. Technical documentation and user documentation is assessed during the project. Each team maintains a public repository/website outlining their progress and storing tracking material such as meeting minutes and decision logs. In addition, the team repository, team minutes and project plan are inspected to audit whether or not the PMP processes have been followed as documented.

[3] WBS — work breakdown structure.

Marks are allocated for output, including technical documents, an implemented prototype and testing reports. Technical content is covered in prerequisite units; however, in project units there are supporting lectures and peer discussion workshops. In line with having unique, real world and varied projects, students are encouraged to decide upon the content of their specific project's documentation. We do not publish fixed templates for students to adhere to but request that students — as a team — decide upon the relevancy of sections as they relate to their own project. Students self-manage but are given milestones that must be met. Dates are set when material/presentation/documentation must be shown to their supervisor or the class. There is a loose structure that must be adhered to by all teams, but teams need to adapt the structure and documentation to suit their particular project.

Presentations are quite deliberately varied to provide opportunities for communication at different levels and to different audiences. We have marketing, user-focused and technical presentations using formal presentations and poster sessions.

Scaffolding support

Despite all of the documented benefits, software team projects are not without their problems. Software engineering capstone courses are 'pedagogically different from the standard courses within their program' (Lynch et al., 2004). Rather than focusing on individual learning of measurable tasks, the project requires collaborative teamwork (Newman et al., 2003). Undergraduates report valuing the experience and recognise the importance of teamwork and collaborative learning (Crebert, Bates, Bell, Patrick, & Cragnolini, 2004), however, they can find adapting to (relatively) large-scale projects somewhat disconcerting.

Each team needs to be supervised to ensure that they have enough direction and confidence to approach community-based projects without being daunted. It is important to provide a support structure that is aligned to the assessment tasks so that students are directed toward success, whilst at the same time providing them with a safety net for mistakes. Experience has shown us that it is not helpful if students are taught in a prescriptive manner. Rather, student learning is best achieved in the presence of good 'sign posts' and guides. These provide safeguards to ensuring projects are likely to survive and reach maturity, even when problems are encountered. Elsewhere, we have described how our flexible structure has been adapted and implemented at various institutions in regional and metropolitan Australian universities, including also off-shore and Australian teacher partners teaching to international students (Keogh et. al., 2007).

Feedback to students is very important to support formative assessment and self-regulated learning (Nicol & MacFarlane-Dick, 2006). We provide our students with access to comprehensive examples chosen from previous projects and encourage them to compare their work with these exemplars. Student feedback confirms that the examples are helpful. As one student commented[4]:

> It was such a huge help having last semesters work as a sample as it was a hard task for the PM[5] at the beginning to really know where to start with planning and organizing.

Students are empowered to compare and self-assess their work. Students submit documentation drafts for feedback prior to assessment, then at an agreed milestone date for initial assessment and again on completion of the project. We have found that students do learn through this hands-on experience and we argue that such learning might be limited if students did not have to construct and define their own document contents. One student commented[6]:

> No exam is necessary for this unit because I learnt more doing the hands on stuff rather then reading and memorizing.

Additional peer support involving interaction with past and current teams can be organised. We have achieved this by timetabling joint workshop discussions and also engaging students to review the work of other teams in a formal way.

[4] End of semester feedback provided on completion of Project 1, July 2007, University of Ballarat.

[5] PM: Project Manager.

[6] End of semester feedback provided on completion of Project 1, July 2007, University of Ballarat.

Institutional experiences, evaluation and feedback

On completion of each project course we have collected feedback from clients. Generally, the feedback has been extremely positive praising the professionalism of our students in their client dealings and also commenting upon the quality of the documentation.

Student feedback has been gathered by student surveys and during final interviews. When problems are reported, they are not technical ones. Rather, they occur when one or other of the team members didn't or couldn't contribute equally to the completion of the project. This becomes a significant issue that must be addressed in the assessment to ensure grades are fairly awarded. Fair assessment of group work is discussed widely in the literature and is not addressed here (see for example, Gatfield, 1999; Freeman & McKenzie, 2002).

Students experience satisfaction at completing high quality material for a real client. Generally, feedback from students includes comments such as 'lots of work involved, but great when we finish and see what we have done'. Those who fully engage with their project achieve excellent final grades and an industrial strength experience. A recent graduate commented[7]:

> The project units were awesome; they gave you a real life experience in all facets of project management and the phases involved. Our project was very relevant to future employment and the teamwork aspect enhanced our team working skills in readiness for industry employment.

A recent set of student surveys[8] asked students to rate their confidence in managing a successful project in their future workplace and to compare that confidence with what they had prior to enrolling in their project unit. Of the twenty students surveyed, eight reported the same level of confidence, twelve reported an increased level of confidence (seven of the twelve changed from 'not at all confident' to 'some confidence'). This supports the anecdotal feedback from students returning for graduation ceremonies, which confirms that students value their project experience as a good preparation for the future career work.

> I am definitely more confident in my ability as a Project Manager and it has confirmed that this is my chosen career. I think it will also look really good in my resume and I really hope the client was happy with our efforts[9].

Conclusion

Team projects with external clients provide an opportunity for students to gain industry experience, whilst being supported and guided by staff to minimise risks. Positive experiences are predicated on suitable selection of appropriate projects and 'good' clients, who are supportive of the educational aims of the project unit. Although students are highly motivated to achieve success in delivering a working system to their client's satisfaction, in reality students often do not have the management, planning or team coordination skills and they often make mistakes along the way. Professional wisdom suggests that students be allowed to make these mistakes and that they should be given the opportunity to learn from them. As well, students get the opportunity to practise management skills and experience group dynamics beyond theoretical exercises so that they can think independently and adapt to realistic problems that are likely to be encountered in the workplace.

We have outlined an assessment model designed to help students experience the process of software development whilst motivating them to complete the appropriate documentation. The model enforces appropriate project management practices and software engineering standards; it encourages best practice and helps ensure the final delivery of high quality software. With appropriate support, students can achieve great outcomes and within the safety of the educational institution — they can develop skills and experience to help ready them for graduate employment.

---

[7] University of Ballarat BIT(Prof Prac) Courses Review, 2007.

[8] End of project survey, 2006, University of Ballarat.

[9] Student feedback provided by email after completion of project, July 2007, University of Ballarat.

References

Bennett, N., Dunne, E., & Carre, C. (1999). Patterns of core and generic skill provision in higher education. *Higher Education, 37*, 71–93.

Bothe, K. (2001). Reverse engineering: The challenge of large scale real-world educational projects. *Proceedings of the 14th conference on software engineering education and training* (pp. 115–126).

Clear, T., Young, F., Goldweber, M., Leidig, P., & Scott, K. (2001). Resources for instructors of capstone courses in computing [Working group report ITiCSE]. *SIGCSE Bulletin, 33*(4),93–113.

Crebert, G., Bates, M, Bell, B., Patrick, C. J., & Cragnolini, V. (2004). Developing generic skills at university, during work placement and in employment: Graduates' perceptions. *Higher Education Research & Development, 23*(2), 1–21.

Cranmer, S. (2006). Enhancing graduate employability: Best intentions and mixed outcomes. *Studies in Higher Education, 31*(2), 169–184..

Centre for the Study of Higher Education (CSHE). (2007). *Core principles of effective assessment.* Centre for the Study of Higher Education, University of Melbourne. Retrieved August 13, 2007, from http://www.cshe.unimelb.edu.au/assessinglearning/docs/CorePrinciples.pdf

Daniels, M., Faulkner, X., & Newman, I. (2002, February25–27). Open ended group projects, motivating students and preparing them for the 'real world'. *Proceedings of 15th conference on software engineering education and training, 2002 (CSEE & T 2002)* (pp. 115–126). Covington, KY.

Dawson, R. (2000). Twenty dirty tricks to train software engineers. *Proceedings of the 22nd international conference on software engineering* (pp. 209–218). Limerick, Ireland.

Ditcher, A., & Hunter, S. (2000, July 2–5). *The instrumental student: An increasing problem?* Paper delivered at Flexible learning for a flexible society: HERDSA (The Higher Education Research and Development Society of Australasia) conference, Toowoomba, Australia.

Fleming, S. (2005). Talking past each other: Student and staff reflection in undergraduate software projects. *Issues in Informing Science and Information Technology, 2*, 93–102.

Freeman, M., & McKenzie, J. (2002). SPARK, a confidential web-based template for self and peer assessment of student teamwork: Benefits of evaluating across different subjects. *British Journal of Educational Technology, 33*(5), 551–569.

Garlan, Gluch, & Tomayko. (1997, November). *Agents of change: Educating software leaders (*pp. 59–65). IEEE Computer Society.

Gatfield, T. (1999). Examining student satisfaction with group projects and peer assessment. *Assessment & Evaluation in Higher Education, 24(*4), 365–377.

Greening, T., Kay, J., Kingston, J. H., &. Crawford, K. (1997). Results of a PBL trial in first-year computer science. In J. Hurst (Ed.), *ACM proceedings of the 2nd Australasian conference on computer science education* (pp. 201–206).

Hagan, D. (2004). Employer satisfaction with ICT graduates. In *Proceedings of the sixth Australasian computing education conference (ACE2004).*

Keogh, K., Sterling, L., & Venables, A. (in press). A scalable and portable structure for conducting successful year-long undergraduate software team projects with real clients. *Journal of Information Technology Education.*

Kolb, A., & Kolb, D. (2005). Learning styles and learning spaces: Enhancing experiential learning in higher education. *Academy of Management Learning and Education, 4(*2), 193–212.

Krause, K. (2006, July). *Dimensions of student engagement: New opportunities for learning and teaching.* Keynote address at the University of Ballarat Learning and Teaching Conference, Ballarat, Victoria.

Lynch, K., Goold, A., & Blain, J. (2004). Students' pedagogical preferences in the delivery of it capstone courses. *Issues in Informing Science and Information Technology, 1,* 431–442.

McLay, A., Corich, A., & Millma, M. (2005, July 1–13). Projecting students to employment. In *Proceedings of the 18th annual conference of the National Advisory Committee on Computing Qualification* (pp. 95–103), Tauranga, New Zealand.

Newman, I., Daniels, M., & Faulkner, X. (2003). *Open ended group projects: A "tool" for more effective teaching.* Paper delivered at the Fifth Australasian Conference on Computing Education, Adelaide, Australia.

Nicol, D. J., & MacFarlane-Dick, D. (2006). Formative assessment and self-regulated learning: A model and seven principles of good feedback practice. *Studies in Higher Education, 31*(2), 199–218.

Nunan, T. (1999, July 12–15). *Graduate qualities, employment and mass higher education.* Paper delivered at Cornerstones: Higher Education Research and Development Society of Australasia (HERDSA) Conference, Melbourne.

Snoke, R., & Underwood, A. (2001, June 27–29). An Australian IT industry view of generic attributes for is graduates. Paper delivered at the 9th European Conference on Information Systems, Lake Bled, Slovenia.

Toleman, M., Roberts, D., & Ryan, C. (2004). Retrofitting generic graduate attributes: A case-study of information systems undergraduate programs. *International Journal of Issues in Informing Science and Information Technology, 1,* 625–635.

Venables, A., Tan, G., Devi Nagappan, S., & Ghous, A. (2006, May 21–24). *Everything we wanted to know about our course, but were afraid to ask? Views from a students' perspective.* Paper delivered at the 2006 Information Resources Management Association International Conference, Washington D.C.

Wilhelm, W. B., & Comegys, C. (2004). Course selection decisions by students on campuses with and without published teaching evaluations [Electronic version]. *Practical Assessment, Research & Evaluation, 9*(16). Retrieved July 18, 2005, from http://PAREonline.net/getvn.asp?v=9&n=16

Copyright statement