Issues in Informing Science and Information Technology

Web ACE – A Study in Reciprocal Informing

Iwona Miliszewska and Grace Tan
Victoria University, Melbourne City, Australia

Iwona.Miliszewska@vu.edu.au Grace.Tan@vu.edu.au

Abstract

A Computer Science degree includes a compulsory final year Project subject. The Project involves the design and implementation of a real-life computer application for a client, and gives students an opportunity to work in a setting emulating a real-life information technology environment. Students undertake the Project subject together with its co-requisite subject in English Language and Communication. The English subject focuses on consolidating written and oral communication skills such as compilation of technical reports, and delivery of oral presentations – skills directly relevant to the Project. This paper reports on a unique Project experience where a group of students developed a Web based system for 'special' clients – their English lecturers. The paper discusses the rationale for the project, its details and benefits. It highlights the relationship between the Project students and the English lecturers and the dual roles of informers and clients, that each of the parties played in the process.

Keywords: education, problem-based learning, reciprocal learning, software development, communication skills, teamwork.

Introduction

The Computer Science degree at Victoria University is a three-year full time program, and the majority of students enter the course as school-leavers. A significant part of the degree is a final year subject – the Project. The Project, regarded as a 'dress rehearsal' for computing students about to graduate and face the needs of the commercial and industrial world, places special emphasis on consolidation of problem solving skills, and the enhancement of associated communication skills. This emphasis is particularly timely as a national report on Employer Satisfaction with Graduate Skills (AC Nielsen Research Services, 2000) found that Computer Science graduates lacked in both areas. Relatively low ratings were given for both written and oral business communication skills, initiative, leadership qualities, personal presentation, and problem solving skills. The Project provides students with an opportunity to develop and improve these essential skills before they enter the workforce and, as such, it is likely to influence their future employability.

Project students, under the supervision of a staff member, tackle a software development problem for a company or organisation. Typical projects include design and development of database applications, Web based applications, or software modules for computer packages; integration and enhancement of existing software modules; and system analysis and simulation studies of production and inventory systems. Students are divided into groups of three to four members; each group is allocated a project, a sponsor – the client, and an academic supervisor. Each group works on their project, while liaising with the client and consulting with the supervisor, submits reports, and gives presentations. Their work is further helped and scrutinised by the English lecturers who, while not familiar with the computing side of the projects, are language and communication experts. They play a vital role in helping, the often reluctant, computing students realise the importance of good communication skills: the effect of an articulate interview, the impact of well structured and clear reports, and the power of well prepared presentations.

Typically, during the course of the Project subject, students would only learn language and communication skills from their English lecturers and, in turn, the lecturers would only teach those skills, i.e. the roles of the participating parties would be well defined and one-sided. Referring to the Informing Science paradigm, students, as learners/clients, would be on the receiving side of information, and lecturers would play the role of teachers/providers. This division of roles and a single platform of interaction – language and communication – left little, if any, room for reciprocal learning between students and lecturers. Could this reciprocity of roles be realised in the Project environment?

This paper reports on a unique Project experience, which provided an answer to the above challenge. In this instance, students developed a Web based content management system for their English lecturers. The paper identifies the unique opportunities presented by this Project experience, and refers them to related research findings. It describes the development process from its conception to final deployment; the introduction of a second platform of interaction – information technology – that enabled reciprocal teaching and learning; and, the benefits that all participating parties gained from the experience.

The Project

The aim of the Project is to provide students with an opportunity to work on a real-life software development task; to appreciate the needs of the business client for whom they are expected to build the software system; to apply software engineering and database design methodologies to the design and implementation of a complete system; to confront issues developers face on a daily basis, such as liaison with clients, working in a team, and documenting the system; to gain experience in translating their knowledge into practice; and, to obtain feedback concerning their progress from intensive reviews of their work.

Underlying Conceptual Frameworks

Firstly, the Project promotes Problem Based Learning (PBL). Many variations of PBL have been documented, but all of them include situations where an initial problem serves as a catalyst for subsequent learning (Fogarty, 1997; Kingsland, 1996). The learning that occurs in working with the problem enables students to develop new knowledge, as well as further consolidate their existing skills and knowledge (Schiller, Ostwald & Chen, 1994). Students have a greater responsibility for their own learning and receive less guidance from the teacher, factors distinguishing PBL from subject-based learning (Dolmans, 1992). The role of the teacher is different too: it is one of consultant rather than instructor. PBL usually includes a collaborative component; students often work in groups where collective decisions are made about task distribution, and where group members investigate different aspects of the problem that together contribute to the total solution. PBL is often project based, with the project work varying in complexity and scope from a project exercise, through a project component and project orientation (Morgan, 1984).

Secondly, in addition to PBL in a collaborative environment, the Project advocates the pursuit of worthwhile tasks, that is, real projects commissioned by real clients. The emphasis on ‘authentic’ tasks, PBL, and collaboration exemplify the three components of another framework for learning – Engagement Theory. *The major premise of engagement theory is that students must be engaged* *in their course work in order for effective learning to occur. The theory posits three primary means to accomplish engagement: (1) an emphasis on collaborative efforts, (2) project-based assignments, and (3) non-academic focus. It is suggested that these three methods result in learning that is creative, meaningful, and authentic* (Kearsley & Schneiderman, 1999). Engagement theory is based on the idea of creating successful collaborative teams that work on tasks that are meaningful to someone outside the classroom. Its core principles are summarized as “Relate”, which emphasizes characteristics such as communication and social skills that are involved in team effort; “Create”, which regards learning as a creative, purposeful activity; and “Donate”, which encourages learners to position their learning in terms of wider community involvement.

Thirdly, the Project is the final and all-encompassing subject in the degree program. It is a culminating experience in which students synthesize and consolidate knowledge acquired throughout their studies. Fairchild & Taylor (2000) define such experience as a Capstone. In addition to integration, experiential learning, and real-world problem solving, capstone emphasizes teamwork, decision-making, critical thinking, and interpersonal communication. These are non-technical skills considered essential in many types of work, and computing work is no exception; graduates are expected to demonstrate technical skills, as well as project management skills and awareness of business practices (Fairchild & Taylor, 2000; Magney, 1996; Novitzki, 1998; AC Nielsen Research Services, 2000).

Lastly, and ideally, the Project should also be an instance of reciprocal learning. Project students should learn from the Project experience and, at the same time, communicate their computing knowledge to others, particularly, their fellow project group members and project sponsors. Research promotes reciprocal roles of teachers and learners, and stresses their interdependence (McLoughlin & Oliver, 1999). It also advocates a shared responsibility for creating and exploring knowledge (Scardamalia & Bereiter, 1991; 1994).

Application of all the above frameworks helps ensure the educational quality of the Project, as it supports the universal seven principles for good practice in undergraduate education: encourages contact between students and lecturers; develops reciprocity and cooperation among students; encourages active learning; gives prompt feedback; emphasises time on task; communicates high expectations; and, respects diverse talents and ways of learning (Chickering & Ehrmann, 1996; Chickering & Gamson, 1987; Chickering & Gamson, 1991).

Project Model

Guided by the frameworks presented above, Project was included as a mandatory subject in the Computer Science degree to enable students to consolidate the knowledge and skills gained throughout the degree. As a consequence, an elaborate system of supervision exists to support Project students. Students form groups of three to four members and nominate a group leader; each group is allocated an industry project (task) and an academic supervisor. The projects are sourced, by the Project coordinator, from local business and industry, and typically are applications, of interest to the clients (project sponsors). Although the problems are situated in the ‘real-world’, the pressure on the students for successful completion is minimized. Thus, students are able to concentrate on the problem in hand. They can proceed at a pace that enables them to gain maximum benefit from the experience, and complete work that is of high standard.

Once the projects have been allocated, students contact their supervisor to arrange the initial meeting with the client/sponsor. Meetings with supervisors take place every week. Each group keeps a logbook used to scribble ideas, designs, and bits of codes, and to record the number of hours spent on the project. During the course of the Project, each goup submits a number of documents including the logbook and project reports; gives oral presentations to fellow students and project supervisors at various stages throughout the year; and, finally, demonstrates its software system. Project groups present all the facets of their systems and reply to any questions regarding functionality and implementation. Grades are awarded on reports, oral presentations, and software demonstration.

English Language and Communication

Students undertaking the Project enrol in a co-requisite subject in English Language and Communication. This subject builds on language and research skills acquired earlier, and has two goals: to provide students with advanced written and oral communication skills necessary for successful completion of their academic studies, particularly the Project; and, to provide preliminary preparation for future employment.

With respect to academic studies, the subject emphasises the development of skills such as listening and note taking, reading and summarising, locating and assessing information sources, writing reports, and making oral presentations. Students apply these skills to the Project and learn how to plan, draft, and edit reports, and how to give oral presentations of their work to an audience. Emphasis is given to the conventions of the English language – grammar, syntax and vocabulary – and to clear writing in a range of contexts. The involvement of the English Language and Communications lecturers in the Project subject provides students with an opportunity to obtain professional feedback on the content and style of project documents prior to formal submissions, and on oral presentations during trial runs.

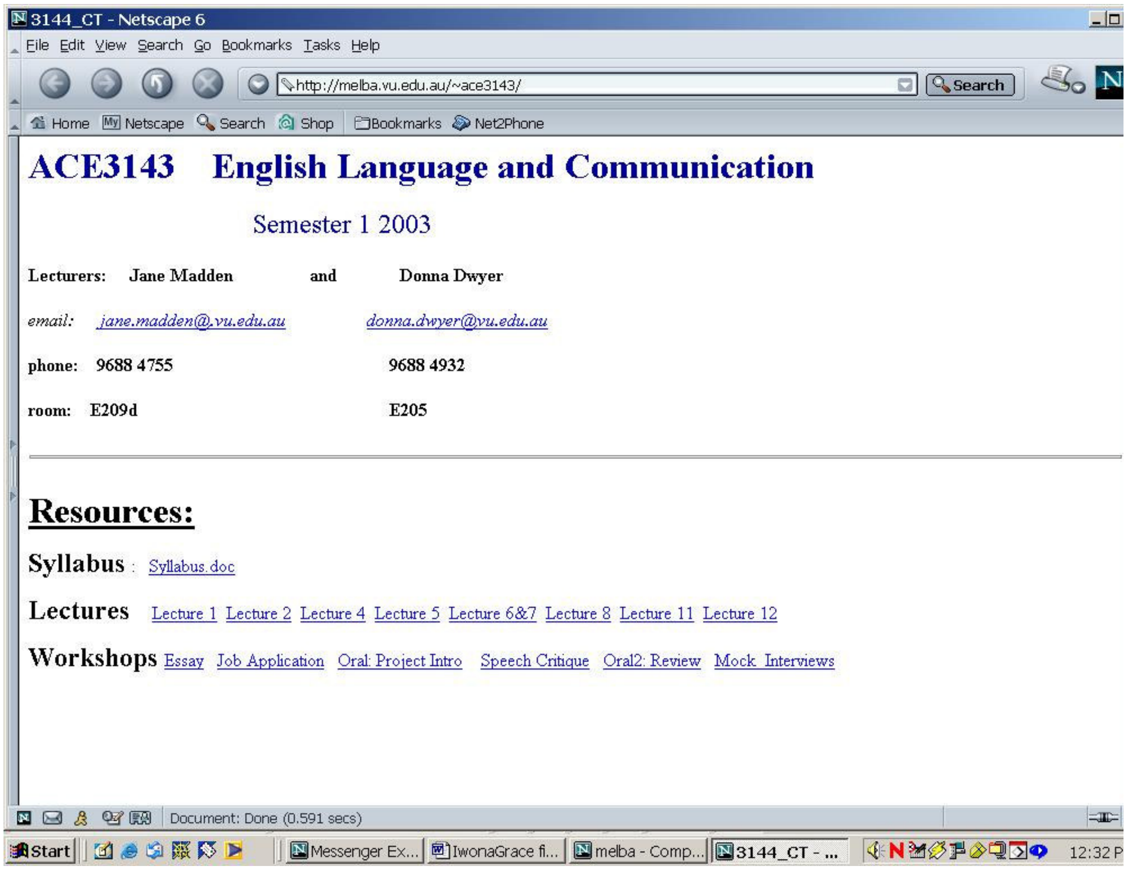
In regard to future employment, the English subject introduces students to work-related skills including job applications, resumes, interview techniques, interpersonal skills, small group communication, and writing and speaking professionally. Students continue their work on Project, while presenting the results of their work in written reports and oral presentations. Emphasis is given to the professional standard of their efforts.

Web ACE

Information on the English Language and Communication subject and course material was provided to students via the Internet on the subject's Web site. The Web site, depicted in Figure 1, was very basic and poorly structured; its layout seemed awkward – links to subject material were organised horizontally across the page – hindering navigation; it was impossible to distinguish between new and old information; and, there were no links to related subjects, nor any links to useful resources. Moreover, the Web site was maintained not by the English lecturers themselves, but by a third party. As a consequence, most of the time, the displayed information was outdated; even news on 'current' events reached the Web site after the events had occurred. These delays were unhelpful and annoying to the users. Indeed, to Computer Science students, the Web site was an example of 'how not to design and maintain a Web site', as it violated all the criteria of effective Web sites (Rathswohl, 2002).

Motivation

A peculiar scenario has developed: on the one hand, professionals in the field of language and communication taught computing students new and valuable skills utilizing a clumsy information technology (IT) tool in the form of the subject Web site; and, on the other hand, budding computing experts, while appreciative of the new language and communication knowledge imparted to them, grew quite frustrated with the antiquated subject Web site and felt that they, too, could 'teach' their English lecturers a lesson, in IT.


3144_CT - Netscape 6
File Edit View Search Go Bookmarks Tasks Help
http://melba.vu.edu.au/~ace3143/
Search
Home Netscape Search Shop Bookmarks Net2Phone
Document: Done (0.591 secs)
Start
Messenger Ex...
IwonaGrace fi...
melba - Comp...
3144_CT - ...
12:32 P

ACE3143 English Language and Communication

Semester 1 2003

Lecturers: Jane Madden and Donna Dwyer

email: jane.madden@vu.edu.au donna.dwyer@vu.edu.au

phone: 9688 4755 9688 4932

room: E209d E205

Resources:

Syllabus : Syllabus.doc

Lectures Lecture 1 Lecture 2 Lecture 4 Lecture 5 Lecture 6&7 Lecture 8 Lecture 11 Lecture 12

Workshops Essay Job Application Oral: Project Intro Speech Critique Oral2: Review Mock Interviews

Figure 1: The old Web site for the English Language and Communication subject.

A group of three students felt so strongly about the matter that instead of selecting a project from the available list, they proposed a project of their own – the development of a new Web based system for the English subject. They believed that, as 'insiders' – both in their capacity as users of the existing subject Web site and IT 'professionals' at the same time – they were in a unique position to design and implement a new system that would be functionally sound, user friendly, and easy to maintain by the English lecturers themselves. Having obtained approval from the Project Coordinator, the students approached their English lecturers with the offer. The lecturers, somewhat surprised with the students' initiative, agreed. This was the beginning of Web ACE (the name for the new system was derived from the subject code – ACE3144).

Analysis and Design

It was decided that the new system should satisfy two main objectives. Firstly, to enable the English lecturers to perform collective editing of subject material in the back-end of the system, and to facilitate subsequent placement of the material on the students' Web site without the involvement of a third party. Secondly, to provide students with current subject material, instant feedback on assignments, links to useful resources, and the ability to interact with English staff and fellow students online. It was therefore apparent from the start that the existing static Web site needed to be replaced by a dynamically updateable content management system with a back-end managed by the English lecturers, and a front-end available to students.

The lecturers, thrilled as they were with the prospect of timely updates to the subject material posted on the Web site, met the idea with some apprehension. They were afraid that, with the third party removed, it would be their responsibility to write software code every time they wished to make the required updates to the Web site. Students dispelled their worries in an instance, explaining the ‘obvious’ to the ‘technically challenged’ English lecturers in a language that they could easily understand for example, that updating the content of the Web site would be as simple as sending an email. Interestingly, when asked about identifying requirements of the new system, the lecturers could not think beyond the scope and capabilities of the existing one. Here, again, the students came to the rescue and presented a list of possible requirements. The overwhelming reaction of the lecturers was an utter disbelief at the functionality and variety of presented options, as they repeatedly asked ‘Can you really do that?’ The system requirements were finalised during subsequent meetings and included facilities to: edit (collectively), upload and update subject material, news, and results; create and edit links to resources; manage staff information; and, communicate with students. A ‘frequently asked questions’ (FAQ) list, help function, and an appropriate security mechanism were also included. Lecturers were also consulted regarding screen layouts and colour scheme for the system.

The other group of clients, the students enrolled in the English subject, also participated in the requirements collection process. They were asked to complete a survey detailing the proposed system; they stated their preferences regarding various facets of the system ranging from functions, through layouts, types of links and font sizes; they were also invited to identify any additional features that they wished included in the system. In addition, the existing Web site was evaluated using Jakob Nielsen’s (2000) Useability Principles to identify its deficiencies and create suitable responses.

Having established and consolidated the system requirements, the students, in consultation with their supervisor, decided that the best approach to develop the Web ACE system would be to use Boehm’s spiral model of the software process; it involved four continuous stages of planning, analysing, designing and implementing, followed by evaluation of each iteration of the software process (Sommerville, 1996). It was decided that the Unified Modeling Language (UML) would be used to specify, construct and document the architecture of the system (Priestley, 2000). Different ‘use case’ diagrams and scenarios were used to illustrate the sequence of events for the various system functions for instance, logging in to the system, viewing students’ results, uploading subject material by staff, and downloading of material by students. Other non-functional requirements such as user interfaces and useability, system performance with respect to downloading time, hardware options, maintainability, portability, and security were also defined and documented in the Software Requirements Specification (SRS) document (Sommerville, 1996).

The design phase, following the development of SRS, consisted of three major activities: preliminary design of mock-up graphical user interfaces; design of system architecture; and, the detailed design of the entire system. Then, a prototype of the new system was implemented and presented to the clients for evaluation. The process was repeated, following the Boehm’s spiral model, and each time the system was enhanced and refined.

Implementation

The front-end of Web ACE – the part of the system accessible to the students – was primarily coded in ASP, with HTML and VBScript linked with a Microsoft Access database. Coding has been completed in segments, with each system function, such as ‘News’, broken down into smaller functional modules for example ‘Add News’, ‘Edit News’, Delete News, etc. The code of these smaller modules was subsequently re-used in similar functional elements of other main functions such as ‘Staff Information’, ‘Help’ and ‘Links’ for example. The front-end Web site was implemented only after the back-end of system – the part of the system used and managed by the English lecturers – had been completed. This ordering of implementation steps was necessary, as the front-end relied heavily on data that could only be created through the back-end. Once both the front-end and back-end were up and running, testing of both subsystems could be conducted, and necessary alterations easily made, as the core code had been already established. The Web ACE prototype was repeatedly evaluated by the project team, the English lecturers, and the project supervisor. Ideas raised and recommendations made in these sessions were used to fix any existing errors, improve inadequate areas, and accommodate new user requirements. It is worth noting that the English lecturers were almost apologetic every time they wished that yet another enhancement be included in the system. Suddenly, they gained a whole new appreciation of the impact that such changes would have not only on the design and implementation of the software system, but also on system documentation and reports – the very assessment tasks required in the English subject.

Web ACE used a metaphor of a filing cabinet to help users find information, and navigate through the system in a logical manner. Helpful icons, resembling the tasks at hand, were used to create associations between the system and the real world. A site map was included to provide better control and ease navigation of the site. Aesthetic, minimalist, and functional design was applied to the Web interfaces according to criteria advocated by Rathswohl (2002), such as impression on first entry, inclusion of key fact in the homepage, consistency in the use of colour, usefulness of external links, etc. Error handling routines, including helpful error messages and alerts, were also incorporated in the system. For example, error messages were produced in case of incorrect data entry, or upload of oversized files; and alerts were issued to require confirmation for deletion and modification of files, or make users fill in mandatory text fields.

Marketing

With Web ACE fully implemented, its creators decided that their software development effort would not be complete without proper marketing of the finished product. They were not satisfied with the prospect of presenting their system only during the final demonstration session – a requirement for all Project students. They decided that, like in the 'real' IT world, a marketing workshop, coupled with an acceptance test, was in order. The aim of the workshop was to showcase the system, provide a thorough step-by-step demonstration, and a hands-on tutorial session on Web ACE. Students designed, printed, and hand-delivered invitations to the workshop. The fourteen invitees included the English lecturers, the project supervisor, the Project co-ordinator, and a number of senior academic staff. The workshop was conducted in a suitably equipped laboratory with students projecting the demonstration from a PC onto a big screen. Workshop participants, armed with user manuals and logged into the system as super-users, replicated various demonstrated steps on their computers.

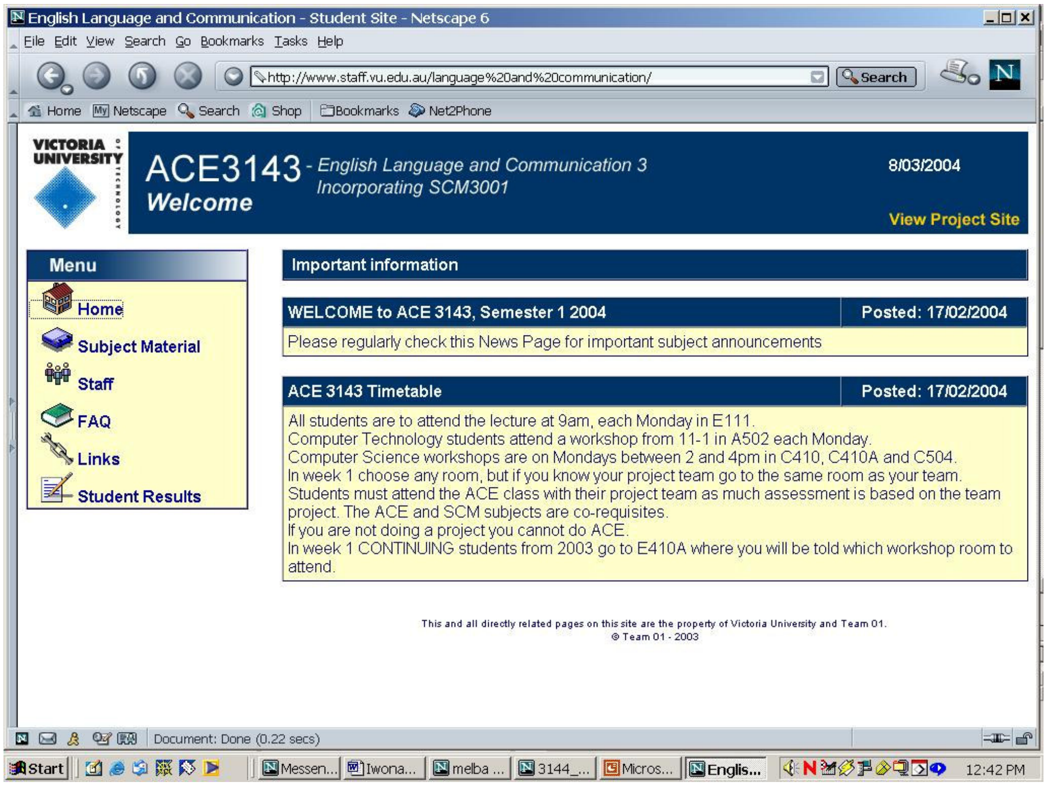
Web ACE creators took turns in presenting various aspects of the system. It was apparent that a great deal of thought and effort had gone to the preparation of the demonstration. It was well structured, well rehearsed, and well timed, with smooth changes between presenters. Demonstrating comprehensive knowledge of the system, presenters skilfully singled out its most impressive 'selling' aspects, and emphasised its advantages. After the presentation, they answered questions and asked for comments and recommendations; they also conducted a survey to obtain final feedback on the system to enable them to put finishing touches on the product before its delivery and deployment.

Evaluation

Web ACE has been regarded highly by its clients throughout its development; this view was further confirmed by the results of the workshop survey completed by fourteen participants. The survey was divided in two parts. The first part evaluated various facets of the system ranging from accessibility and efficiency aspects through helpfulness of on-screen instructions, and quality of screen layouts; a 5-point Likert scale was used to measure the responses, with 5 indicating *strongly agree* and 1 indicating *strongly disagree*. The second part of the survey asked for comments regarding most useful and best liked system features, as well as suggestions regarding possible improvements and future enhancements.

The level of satisfaction with Web ACE, its Students' View depicted in Figure 2, was high, with an average score of 4.88. The features deemed most useful and best liked included the 'News' and 'Links' modules; the workgroup environment enabling collective editing of course material; the ease of adding new course content to students' Web site; uniformity of interfaces and accessibility of various menu options; and, speed of uploading and downloading files. Suggestions of possible improvements and enhancements included minor rearrangements of menu options, for example upgrading 'Results' to a main option and removing it as a subsection of 'Subjects'; renaming 'Help' to FAQ to reflect better the true nature of the system feature; marking mandatory fields with a star (*); and, greying out inaccessible read-only information. The best summary of the Web ACE effort was a single comment on one of the surveys: 'Well done!'

The value of Web ACE and the skills of its developers were given one more vote of confidence – they were turned into an example of good practice to be used with future generations of Project students. This was accomplished in three ways; firstly, Web ACE was immediately deployed, and has been operational ever since; secondly, the documentation generated by the development team in the course of the Project was included as reference material in the English subject; and thirdly, the students' performance in the 'marketing' workshop was videotaped for use as demonstration material in the subject.


English Language and Communication - Student Site - Netscape 6
File Edit View Search Go Bookmarks Tasks Help
http://www.staff.vu.edu.au/language%20and%20communication/
Search
Home My Netscape Search Shop Bookmarks Net2Phone
VICTORIA UNIVERSITY
Menu
Home
Subject Material
Staff
FAQ
Links
Student Results
View Project Site
Document: Done (0.22 secs)
Start
Messen... Iwona... melba ... 3144_... Micros... Englis...
12:42 PM

ACE3143 - *English Language and Communication 3*
Incorporating SCM3001
Welcome

8/03/2004

Important information

WELCOME to ACE 3143, Semester 1 2004 **Posted: 17/02/2004**

Please regularly check this News Page for important subject announcements

ACE 3143 Timetable **Posted: 17/02/2004**

All students are to attend the lecture at 9am, each Monday in E111.
Computer Technology students attend a workshop from 11-1 in A502 each Monday.
Computer Science workshops are on Mondays between 2 and 4pm in C410, C410A and C504.
In week 1 choose any room, but if you know your project team go to the same room as your team.
Students must attend the ACE class with their project team as much assessment is based on the team project. The ACE and SCM subjects are co-requisites.
If you are not doing a project you cannot do ACE.
In week 1 CONTINUING students from 2003 go to E410A where you will be told which workshop room to attend.

This and all directly related pages on this site are the property of Victoria University and Team 01.
© Team 01 - 2003

Figure 2: Web ACE – Students' View.

Conclusions

This paper described the inception and implementation of Web ACE, a Web based content management system developed by Computer Science students, in their final year Project subject, for an unusual group of clients – academics from the department of English Language, Communication and Cultural Studies. Those clients were also involved with the Project students in a different capacity – as their lecturers providing service in the English Language and Communication subject. This unique scenario created two platforms for client/server, or learner/teacher, engagement whereby each of the parties involved played the role of clients/learners on one platform, and the role of service providers on the other. On the English language and communication platform, the English teachers played the role of service providers, and the students were the learners. However, on the computing platform the roles were reversed, the students provided service, and the English teachers learned. The fact that the teachers were not the only source of knowledge in the universe of the Project was very important. It created an implicit understanding that teaching and learning was a shared responsibility, and that teachers were also open to learning. As a result, a more horizontal process of interaction developed between teachers and students.

The Web ACE experience yielded a number of benefits: student developers gained confidence in their real-life computing skills and appreciation of good communication skills; English teachers gained respect for the students' computing skills, and saw the positive impact of extra motivation and well directed encouragement on the students' communication skills; and, most importantly, future Project students will benefit not only from using Web ACE, but also from working with English lecturers who have acquired a better understanding of the difficulties faced by students developing real-life computer systems. It appears that Web ACE has come up trumps!

References

AC Nielsen Research Services (2000). Employer Satisfaction with Graduate Skills. Research Report 99/7. Department of Education, Training and Youth Affairs (DETYA).

Chickering, A.W., & Ehrmann, S.C. (1996). Implementing the seven principles: Technology as lever. *AAHE Bulletin, 49* (2).

Chickering, A.W., & Gamson, Z. (1987). Seven principles of good practice in undergraduate education. *AAHE Bulletin*, *39* (7)*,* 3-7.

Chickering, A.W., & Gamson, Z.F. (1991). *Applying the seven principles for good practice in undergraduate education*. San Francisco: Jossey-Bass

Dolmans, D. (1992). The relationship between student-generated learning issues and self-study in problem-based learning. *Instructional Science*, *22* (4), 251-267.

Fairchild, G.F., & Taylor, T.G. (2000). Business simulations and issue debates to facilitate synthesis in agribusiness capstone courses. Retrieved September 2002, from http://bear.cba.ufl.edu/centers/ciber/workingpapers/capstone.pdf

Fogarty, R. (1997). *Problem-based learning and other curriculum models for the multiple intelligences classroom*. ERIC Document No. ED405143.

Kearsley, G., & Shneiderman, B. (1999). Engagement theory: A framework for technology-based teaching and learning. Retrieved July 2002, from http://home.sprynet.com/~gkearsley/engage.htm

Kingsland, A. (1996). Time expenditure, workload, and student satisfaction in problem-based learning. In L. Wilkerson & W. H. Gijselaers (Eds.), *Bringing problem-based learning to higher education: Theory and practice* (pp.73-81). San Francisco: Jossey-Bass.

Magney, J. (1996). Preparing technicians for the management of technology. *ATEA Journal*, *23* (4), 8-11.

McLoughlin, C. & Oliver, R. (1999). Pedagogic roles and dynamics in telematics environments. In M. Selinger & J. Pearson (Eds.), *Telematics in education: Trends and issues* (pp. 32-50). Oxford: Pergamon.

Morgan, A. (1984). Project-based learning. In E. Henderson & M. Nathenson (Eds.), *Independent learning in higher education.* Englewood Cliffs, NJ: Educational Technology Publications.

Nielsen, J. (2000). *Designing Web usability: The practice of simplicity*. Indianapolis: New Riders Publishing.

Novitzki, J.E. (1998). The MIS capstone: Development of an integrating group applied project course. In *Proceedings of the International Academy for Information Management (IAIM) 13th Annual Conference* (pp.100-109). Helsinki, Finland.

Priestley, M. (2000). *Practical object oriented design with UML*. London: McGraw Hill.

Rathswohl, E. J. (2002). A technique for teaching website effectiveness in undergraduate I.S. classes. In *Proceedings of the International IS2002 Informing Science and IT Education Conference* (pp. 1311-1316). Cork, Ireland.

Scardamalia, M., & Bereiter, C. (1991). Higher levels of agency for children in knowledge building: A challenge for the design of new knowledge media. *Journal of Learning Sciences*, *1* (1), 37-68.

Scardamalia, M., & Bereiter, C. (1994). Computer support for knowledge building communities. *Journal of Learning Sciences*, *3* (3), 265-283.

Schiller, J., Ostwald, M., & Chen, S. (1994). Implementing a problem-based, distance education undergraduate course in construction management. *Distance Education*, *15* (2), 300-317.

Sommerville, I. (1996). *Software engineering* (5th ed.). Wokingham, England: Addison Wesley Publishing Company.

Biography

Iwona Miliszewska is a senior lecturer in Computer Science at Victoria University. She has participated in research projects involving distance education, effective teaching methods, lifelong learning and women in computer science, and has published in these areas.

Grace Tan is a lecturer in Computer Science at Victoria University; her research interests include innovative teaching methods, development of graduate attributes, and issues related to female students in computing courses.