The World Wide Web: a vehicle to develop interactive learning and teaching applications

Shicheng Tian

The author

Shicheng Tian is a Lecturer at the School of Computing and Management Sciences, Sheffield Hallam University, Sheffield, UK.

Keywords

Internet, World Wide Web, Computer-based training, Education

Abstract

The paper is about how to use the World Wide Web (Web) for education. After a brief review of this area, the main issues regarding the use of Web for education are discussed. Based on this discussion, a design, a preliminary implementation of a prototype, the experience gained from this work and future work are described and a few conclusions drawn. Experience from the trial of the system shows that the Web is a cost-effective technology to facilitate the development of educational applications across the Internet.

Electronic access

The research register for this journal is available at
http://www.mcbup.com/research_registers

The current issue and full text archive of this journal is available at
http://www.emerald-library.com/ft

Internet Research: Electronic Networking Applications and Policy
Volume 11 · Number 1 · 2001 · pp. 74–83
© MCB University Press · ISSN 1066-2243

Introduction

The Web provides a distributed multimedia hypertext system that can be used in teaching, research and administration. Using the Web, we can: retrieve multimedia documents from around the world, publish documents globally, run programs on remote servers and download and run programs on local machines. With several million host computers linked to the Internet, and estimates of many times that number of actual users, there is a huge number of potential Web page authors, who are adding new information to the vast information reservoir that already exists.

Many institutions have developed campus-wide information systems using the Web. Students can access resources from around the world to assist them in their learning and take advantage of various distance learning courses which are now available on the Web. Conferences, including reports, video, slide images, etc., can be published on the Web, allowing greater and more rapid dissemination. Clearly, the Web is an important tool for many applications, including education, and is expected to be so for some years to come.

In this paper, the author's using the Web for facilitating teaching and learning are discussed. Apart from supporting normal Web-site features (e.g. publishing lecture slides, providing reference information, etc.), this prototype Web site also supports the so-called, within-the-current-context features (e.g. after a lecture is selected, while keeping the main browser window display the slides of this lecture, in other small, pop-up windows, a student can make comments on the selected lecture, read the comments previously made on this lecture, view the self-asked questions for this lecture, etc.). Besides, students can communicate freely with each other, or with the tutors, using the built-in communication facilities: feedback and e-mail. In order to be aware of who is using the system, an authentication procedure has been designed to ask students to submit their access "ids" and passwords when logging on to the system. After getting into the system, students can change their passwords at any time. In the following, discussions are given on some innovative computer-based education (CBE) initiatives

Internet Research: Electronic Networking Applications and Policy
Volume 11 · Number 1 · 2001 · 74–83

using the Internet and the Web, the main issues related to the use of the Web for education, a preliminary design and an implementation of a Web-oriented educational prototype, the experience gained, future work and a few conclusions.

Related work

The Internet can be used to facilitate teaching and learning. In these kind of applications, for example, students could submit their assignments over the network to a server machine, where acknowledgements of receiving the work would be sent back to the students, and the assignments would be assessed by the computer, according to predefined standards.

The BOSS system (Joy and Luck, 1998) is one such kind of application. With the support of the system, it is possible to have student-programming assignments submitted, compiled and tested automatically. Although fully automatic marking is not supported yet by the system, it does support the process of marking, and enables marking tasks to be divided among several individuals while maintaining rigour and consistency. The developer of the system finds out that not only can this system alleviate the pressure of increasing workload due to having more and more students, it can improve the learning experience available to students (in providing them with facilities for immediate and effective feedback, for example), and also enable other administrative tasks to be automated, as part of a coherent approach to a full course management.

Some Web pages are purely written in HTML, others, however, begin to incorporate some form of interactivity. Normally, to make Web pages interactive, script languages are used to implement small routines that are inserted into these pages. These routines carry out specific functionality, such as checking if the data that is entered to a text field is valid, etc. In order for the Web browsers to successfully present the Web pages that contain those routines, corresponding script language interpreters need to be incorporated into these browsers. Usually, these interpreters are called plug-ins.

Several script languages and Internet-oriented languages have been used to embed interactivity into Web pages (Hall, 1998), including Java and JavaScript. Among them, Java is the most powerful tool to make Web pages interactive. This is because it is a proper object-oriented language, platform free and has more functionality than its competitors. Actually, programming the Web with Java remains the hottest topic in recent years, in both the academic world and the computer industry. Java applets embedded into Web pages can be run on all kinds of computer systems, so long as machine-dependent Java virtual machines have been installed.

The Computer Assisted Teaching and Learning (CASTLE) project (Pownall and Mobbs, 1998) is a good example of using the emerging Web technology for CBE. This system contains a multi-choice question (MCQ) bank, a Common Gateway Interface (CGI) marking program and a number of utility tools. The system provides teachers and students with an easy to use tool set for the delivery of online assessments. Using the provided tool set requires no prior knowledge of any CGI scripting and HTML. The only requirement to create an online test is the ability to use a Web browser. In this way, this system is platform independent. The assessment documents are in standard HTML. Therefore, the tests may contain any multimedia files, which can be delivered over the Web, from audio, images, or animated GIF files to links to other Web resources. According to the developer of this system, Java is going to be used to replace the current CGI approach, in order to take full advantage of the newly emerging, Web-oriented, object-oriented language, to further improve the functionality of the system.

Apart from using traditional approaches (e.g. CDs, videodiscs, single-machine packages), there is a trend now in using the Internet and especially the emerging Web technology to facilitate CBE applications. In the next section, the main issues related to using the Web for facilitating teaching and learning are discussed. These issues will lead to a design and a preliminary implementation of a Web-oriented CBE prototype.

Internet Research: Electronic Networking Applications and Policy
Volume 11 · Number 1 · 2001 · 74–83

Six basic issues of using the Web for education

Basically, CBE contains five components: teachers possessing the knowledge and being able to teach, students willing to learn, the knowledge itself, the means to evaluate the learning outcomes of the learners, and the enabling technology. The first four components are identified in Pownall and Mobbs (1998). A careful review of a face-to-face conventional educational context, however, will lead to adding an additional component to the list: the communications between teachers and students during the whole process of the teaching and learning activity. Indeed, teachers do have many communications with students, regardless of whether on course work, projects, assignments, etc. It would be hard to imagine achieving any fruitful academic result without these vital communications. Hence, this factor ought to be considered when we try to use the computer technology to facilitate CBE. Therefore, the following six components should be considered for designing CBE systems: teachers, students, knowledge, evaluations, communications and the enabling technology.

Evaluations means conducting assessments to check if the learning outcomes have been achieved. Broadly speaking, assessments can be broken down into two kinds: formative assessments and summative assessments. The former refers to testings that can help students with their learning; it is a way of pinpointing weak points across a whole class or within individual students and a good way to aid student learning, while the latter refers to the end-of-course assessments whose procedures are more rigorous and the student is given a grade which contributes to his or her final degree on a particular course.

The most popular way of using network-oriented computer technology for education is to conduct formative assessments, e.g. MCQ tests. In this case, predefined MCQ tests are presented to students from local machines or over computer networks and the answers from the students are sent back to the computers, where assessing processes are carried out automatically, based on a predefined marking scheme. Although technology is predominantly used in formative assessments, effort from the CBE community is being made to use it for summative assessments as well. Some difficulties are to be overcome before we can fully take advantage of the technology, especially computer technology. For computer-supported summative assessments, these difficulties include insufficient security support, and non-robust network connectivity (Kennedy, 1998).

Figure 1 shows the relationships between the six components mentioned above.

A preliminary design and an implementation of a prototype

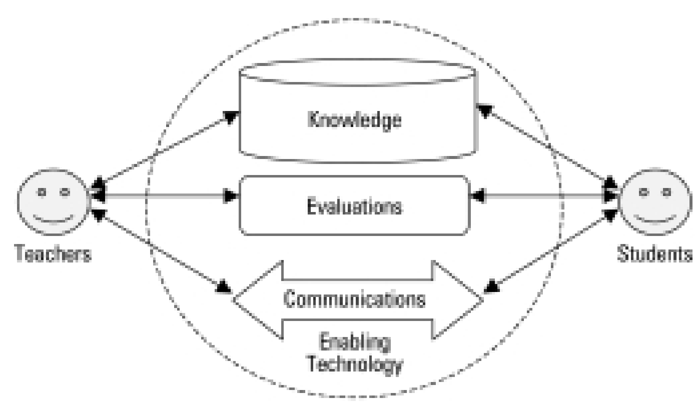
Based on the model derived from the analyses in the previous section, a prototype design and a preliminary implementation are discussed in this section. Experience gained from using this prototype for teaching and learning is given in the next section.

The design

Knowledge

This is related to individual teaching tasks. For the author, it includes the following subjects: Network Systems, Java Programming Languages, JavaScript, Visual Basic and Distributed Object Management. The knowledge is stored in HTML files that can be viewed by students with a Web browser. These files can be constructed with Web authoring tools (e.g. the Microsoft Front Page) or any other text editors (e.g. the popular Notepad for PCs).

Figure 1 The CBE model



Internet Research: Electronic Networking Applications and Policy
Volume 11 · Number 1 · 2001 · 74–83

Evaluations
MCQ tests are designed to check if students achieve the required learning outcomes. Each test is constructed as an HTML page that contains a form within which a set of MCQs is designed. A student answers each question by selecting one of the several options. During an online session, he or she can make as many attempts as possible to answer one question, and immediate feedback is possible, if this is desired. After submitting a test page, the answers are assessed automatically by the prototype system, according to a predefined marking scheme. The answers and their marks are saved by the system for later reference and record-keeping purposes.

Communications
It is necessary to provide channels for teachers and students to communicate with each other, to facilitate their co-operation during a computerised teaching and learning process. Examining face-to-face teaching and learning processes shows that there are several kinds of communications going on between teachers and students. They include one-to-one communications, one-to-many communications and group communications.

Available communication means or dedicated innovative information software should be exploited or developed to provide support to the various communication needs identified above. These may include using the available electronic mail and computer conferencing facilities, and using or reengineering some groupware systems that are communication-centred and publicly available. From the author's experience, for example, only a limited amount of effort needs to be made to incorporate a discussing forum into the prototype, through reusing a suitable public-domain groupware.

Enabling technology
Enabling technology is closely related to all of the three components discussed above. HTML is used for storing the knowledge; HTML and assessment software is used for evaluating the expected learning outcomes; e-mail, computer conferencing and groupware technology are used for communications. Apart from utilising the above technology resources, an overall communication framework needs to be designed. This includes using the Internet and the Web as a vehicle to pass and store information and course-related knowledge, and designing a middleware program. This program is responsible for co-ordination between teachers and students, automatically assessing student tests, enabling and controlling the e-mail, computer conferencing facilities and the embedded groupware system.

Teachers and students
They are the users who use the system. First, they need to be convinced that the system is effective, easy to use and helpful for their teaching and learning activities, compared with conventional approaches. Then, they should be given trials through which they learn how to use the system. It is important to encourage them to contribute comments on the system. Their feedback should be evaluated carefully, to modify the system and tailor its functionality, as well as the user interface, to the need of the users. User feedback should be collected regularly during the process of implementing the system and after it has been implemented, in order to continually modify the system to suit users' requirements.

The preliminary implementation

The prototype consists of two parts: the client and the server. For the client side, a Web browser is used by a student to access the knowledge stored at the server side. On the server side, a Web server is installed that hosts a CGI middleware program written in visual basic (VB), with Microsoft Access database accessing capabilities. The data from the users is used to update a user database at the server side. The following are the main components of this prototype.

Client side user interface
The interface consists of four parts (Figure 2); the first part provides course-work related facilities, the second part provides teaching material updating facilities, the third part provides communication facilities and the fourth part provides house-keeping facilities. Only lecturer-users are able to access all these facilities; in contrast, student-users will be denied access to the updating facilities – in fact, such facilities disappear from the interface when student-users log on to the system. A menu bar has been designed and implemented. It consists

Internet Research: Electronic Networking Applications and Policy
Volume 11 · Number 1 · 2001 · 74–83

Figure 2 User interface – the main menu

of a number of menus. Each menu of the menu bar represents an individual facility.

The first part of the interface contains three menus or facilities: unit, lecture and tutorial. Teaching materials are organised under different unit titles. Selecting the unit menu will make a small window appear. This small window has a menu bar that allows a user to create, delete or rename a unit title, or select a particular unit in order to view its teaching materials. Only the select facility is available when a student-user logs on to the system. The lecture and tutorial menus allow a user to view lecture slides and tutorial notes from the currently selected unit. The user can also make a comment on the currently selected lecture or tutorial. After doing so, a tiny microscope icon will be displayed adjacent to the selected lecture/tutorial hyperlink. Multiple comments can be made on each individual lecture/tutorial by individual users. Clicking on the top of the small microscope icon adjacent to one lecture/ tutorial hyperlink allows a user to view the comments and other useful information: e.g. the author(s) of the comment(s), submit times, etc. Adjacent to the hyperlink of each lecture, under the lecture menu window, is another tiny question-mark icon. Clicking on top of this icon will make a small window appear, which shows the self-asked questions from this lecture. Around ten such questions have been designed for each lecture, for students to attempt to facilitate obtaining a better understanding of the just-taught lecture.

Unlike student-users, lecturer-users can use the second part of the interface (add, rename and delete) to update teaching materials. This includes adding, renaming or deleting lecture slides or tutorial notes (Figure 3).

The third part of the interface consists of two menus: feedback and e-mail. Using these two menus, students can send their generic comments on their course-work to the lecturers and send e-mail messages to their classmates, to discuss things of interest around the taught subjects. Comments and e-mail messages are accompanied with senders' names, to facilitate communications among lecturers and students.

The final part of the interface contains: home, password, information and logout. Clicking on home will always return the user to the front page of the system. The password menu is used to allow a user to change his/her password, provided the user has submitted the same password which had been used during the authentication procedure at the beginning of this session. The information menu provides information about units, assignments, references, etc. Using the logout menu, a user can log out from the system explicitly. Otherwise, he/she could be logged out automatically after a predefined period of time. Currently, a 30-minute time-out has been set up for trial purposes.

Server side functionality

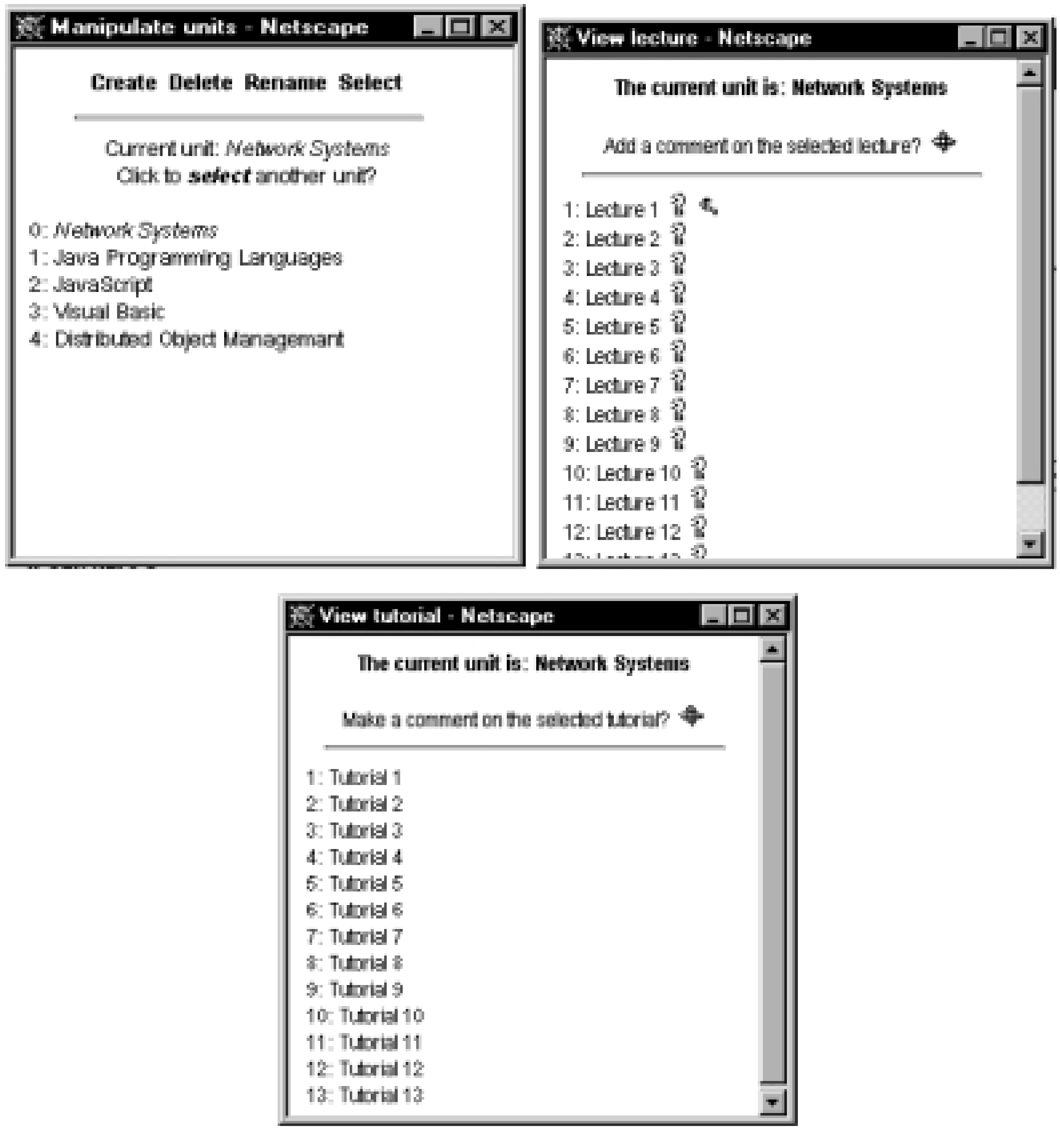
Three parts are composed with the server side: a HTTP Web server, the required CGI program and a backend Microsoft Access user database.

The Web server, currently in use is WebSite-1.0 from the publisher O'Reilly. The CGI program is implemented with VB 6.0 that supports database-accessing primitives. In addition, VB 6.0 is easier for students to both learn and use in comparison to other programming languages, C++, for example. Microsoft Access database management system has been used to host all of the course related data and information, including the teaching materials, user feedback, messages, comments, references, etc. Updating the database is accomplished through issuing structured query language (SQL) queries from within the CGI program. A user can query, modify, or delete available records from the database.

Figure 4 shows the architecture of this prototype, which can be accessed over the Web

Internet Research: Electronic Networking Applications and Policy
Volume 11 · Number 1 · 2001 · 74–83

Figure 3 User interface – the sub-menus



with the following URL: http://shicheng.cms.shu.ac.uk/course, logging on as guest and using guest as the password.

The experience

Our experience in utilising this learning and teaching environment is discussed below, around the six basic issues concerning CBE and the feedback of questionnaires from students, regarding using this prototype system to facilitate their learning process. The six issues on CBE have been described previously.

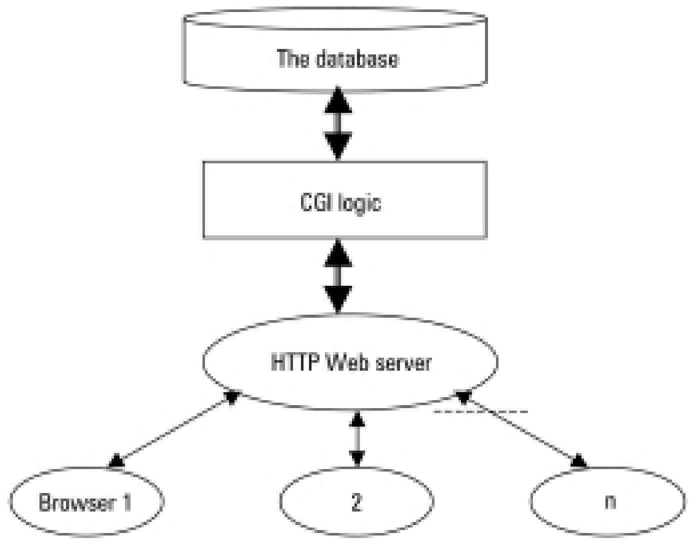
Enabling technology

From our view, the enabling technology now is mature enough to provide essential support for the basic needs of education over the network, especially over the Web. We can use off-the-shelf educational packages or use the latest technologies to develop preferable systems ourselves, to facilitate teaching and learning. The phrase – technology looking for applications – could be used to better describe the situation in this area (i.e. the development of technology is ahead of the demand of using the technology to implement innovative educational systems).

Communications

Two facilities – feedback and e-mail – have been designed and implemented in addressing this important issue. To our surprise, however, few students have so far utilised

Figure 4 The architecture of the Web-oriented CBE prototype



them since the beginning of the trial during the first semester of the 1999-2000 academic year. Our early investigation reveals that there could be two reasons for this happening; students are used to using the university default e-mail system (FirstClass) for communications; and unless absolutely necessary, students always access this Web prototype purely for accessing the teaching materials and other course-work related information. The cold response to the communication facilities suggests that integrating the communication component into a CBE system may depend on the individual application scenario. In the case of a distance learning situation, for example, such integration could be necessary and welcomed. We are planning to try this prototype system with a group of distance learning students, who are currently registered with the department, and find out the response from them, especially on the integrated communication facilities.

Evaluations

Experience in conducting MCQ tests over the Web using another similar prototype causes our concerns as to in what way to conduct the tests so that the final objective – strengthening students' learning outcome – would be reached. Although the advantage of doing so is apparent (e.g. the ease of attempting such tests over the Web, the prompt replies from the server, efficient computerised marking processes, etc.), problems do exist which need further investigations, namely credibility, fairness, reliability, etc. For the problem of credibility we found it very hard to have students attempt Web-oriented MCQ tests individually; cheating may lead to a high similarity in some of the submitted answers, either answered correctly or wrongly. The problem of fairness is related to the problem of credibility; a strong student may get a mark lower than that of a weak student who has cheated. The problem of reliability is about the machines on which students attempt the MCQ tests; the machines should be robust and reliable. Unfortunately, the fact is some machines may crash down during an on-going test. Based on the above experience, we are planing to take a different approach to conducting MCQ tests over the Web. One test of about ten questions will be designed for each lecture; we will encourage students to discuss with their classmates about the answers, when attempting such a test over the Web. Students will be told that no credit will be given towards their effort of attempting those tests, yet will be informed, however, that a final credit, formal, paper form MCQ test will be given. The questions comprising this test will be selected randomly from the whole set of questions which they are encouraged to attempt over the Web, during the period of running the course. We expect a better learning outcome to be reached, after the new approach has been conducted.

Knowledge

Thanks to the innovative features of Microsoft PowerPoint, the slides of each lecture are packaged into a series of associated HTML files (i.e. the first page pointing to the second one, which in turn points to the third one and so on), makes it easy to view them. Besides, a small pop-up window has been designed. It floats on top of the main browser window all the time, displaying the list of lecture titles of the currently selected unit. Selecting another lecture title from this small pop-up window results in the main browser window jumping to the first page of the lecture slides of that lecture. This helps to achieve the so-called within-the-current-context browsing effect. Our experience shows that students feel it is convenient to have the small window float on the top of the screen all the time and control the changing over of slides between different lectures. This small window can be minimised to an icon and moved to one of the corners of the screen, or closed. The design of the self-asked questions of each lecture and the ease of accessing them has been proved to be useful (see Appendix). Attempting those questions helps students to obtain a better understanding of the taught lectures.

Teachers and students
They are the users of the CBE systems. Therefore, starting from the beginning of the development of such a system, this component of teachers and students needs to be considered. Teachers are mainly involved in providing teaching materials and marking submitted assignments, while students use the system to facilitate their learning process. Our preliminary experiences are, designing the CBE systems: first, according to different teaching and learning senior; and second, taking into consideration students' responses in using the developed system.

In the first instance, for the case of this paper, the communication facility of this Web-oriented CBE system may not be necessary, while such a facility is likely to be welcomed for distance learning situations. In the second instance, as discussed in the evaluations paragraph, the current approach of conducting MCQ tests ought to be changed accordingly.

This prototype system has been used to facilitate the delivery of a second-year Higher National Diploma unit. Out of the 28 students if the class, 21 have filled and returned the questionnaire (see the Appendix; the other seven students were absent from the class that day). Each question has five optional answers: strongly agree – SA; agree – A; neutral – N; disagree – DA; and strongly disagree – SD. In Table AI (see the Appendix), each row shows students' responses to a particular question. For example, the row for Q1 shows that six students strongly agree, 12 students agree, two students respond with a neutral answer and one student disagrees with this question. The total number of students (NS) who have submitted the questionnaires is 21. The average score (AS) in the table could be used as an indicator to show student's generic response to this prototype system. For example, the last figures of the SA and the A columns are 6.5 and 9.9; this could be interpreted that around 16 (6.5 + 9.9 = 16.4), out of the 21 students, are basically satisfied with the functionality of the developed teaching and learning environment. In contrast, the last figures of N, DA and SD are 3.5, 0.7 and 0.5. This shows that the system needs further improvements to bring its performance to the expectations of the students. Along this line, further investigations are worth being conducted to find out what exactly the students' complaints are.

Future work

From our work with this prototype, we think the following issues need attention, in order to best use the Web technology for developing innovative CBE applications.

Using the communication facilities for distance learning

As discussed in the previous section, the response to the communication facilities of this prototype is not positive. This may be due to the facts that the users using the system are full-time on-campus students (i.e. they meet and contact with each other face-to-face daily) and that they use the university default e-mail system as their communication tool. We are planning to try the system for a distance learning group of around 20 students and find out what the effectiveness will be for such students. These students have virtually no face-to-face contact with each other, except from communicating via e-mail and talking over the telephone occasionally. We think the importance of integrating communication facility into CBE systems is related closely to the applications of such systems; in the case of distance learning, such a facility is likely to be necessary and of importance. We expect a positive result from the trial of this system for the distance learning students mentioned

Internet Research: Electronic Networking Applications and Policy
Volume 11 · Number 1 · 2001 · 74–83

above, and will modify our system, especially relating to the communication facility, to facilitate teaching and learning processes in the situation of distance learning.

Integrating MCQs
For time constraints, we have not physically implemented the MCQ facility into the current prototype system. We are going to implement this facility in our system and plan to conduct trials during the next semester.

Evaluating other middleware alternatives apart from the CGI approach
Apart from the popular CGI implementation, various other approaches to implementing the server side logic are to be investigated to achieve better efficiencies. Other alternative approaches include Microsoft Active Server Page (ASP), Servlets, Java client-server model and Common Object Request Broker Architecture (CORBA).

Conclusions

The work been done leads me to make the following conclusions.

Exploiting the potential of the Web technology to facilitate the development of innovative CBE applications
Several factors (Coffman and Odlyzko 1998) below support the above claim:
- accessing networked computers (i.e. computers connected to the Web) is becoming easy;
- tens of millions of Web servers are accessible, providing a vast information reservoir, and the number of servers keeps growing exponentially;
- the two main Web browsers (Netscape and Internet Explorer) are free and support various hardware platforms;
- Web-oriented programming languages/ tools are becoming mature and widely available: e.g. HTML, Java, JavaScript, etc.

All these factors offer us great opportunities to develop innovative CBE applications over the Web. These Web-oriented CBE applications promise that users are able to access the applications from anywhere around the world, where networked computers and public domain Web browsers are available, no matter what kind of machines the users work on and what kind of operating systems are running on those machines.

It may not be appropriate to conduct credit, individual MCQ tests using network-oriented computer technology (e.g. over the Web)
From our experience, there are three main reasons for this claim:
(1) A poor reliability of the machines in the labs: some of the machines in the labs are down from time to time.
(2) When a machine is down, to fix the problem can be quite troublesome. On the one hand, quite often, help from departmental computer-support colleagues is neither prompt nor helpful, it may take a long delay to have the problem fixed, sometimes the problem would never get fixed. On the other hand, the tutor could not fix the problem on his or her own. The above situation leads us to a dilemma: we have to ask students to attempt credit MCQ tests with unreliable machines!
(3) It is difficult to prevent students from looking-over-each-others'-shoulders during a lab-based MCQ test unless dedicated testing areas have been arranged, where dividing boards are erected between computers.

As described previously, however, it may be appropriate to conduct uncredit MCQ tests using the network-oriented computer technology, compensated with a credit, end-of-unit, paper-form MCQ test.

Providing better Web graphic user interfaces (GUIs)
Better GUIs are needed for Web-oriented CBE applications, to facilitate users' accessing the applications over the Web. To compensate the limitations of HTML and to design comprehensive GUIs, various browser plug-ins and Web-oriented programming

Internet Research: Electronic Networking Applications and Policy
Volume 11 · Number 1 · 2001 · 74–83

languages should be investigated and used, including Shockwave, Java, JavaScript, etc. Among them, Java is a competitive candidate, because it is platform independent and its current version (Java 2) has adequate GUI support.

References

Coffman, K. and Odlyzko, A. (1998), "The size and growth rate of the Internet", *First Monday*, Vol. 3 No. 10 (http://www.firstmonday.dk/issues/issue3_10/coffman/index.html, peer-reviewed journal of the Internet), October.

Hall, M. (1998), *Core Web Programming*, Prentice-Hall, Englewood Cliffs, NJ.

Joy, M. and Luck, M. (1998), *The BOSS System for On-line Submission and Assessment*, CTI Workshop on Computer Assisted Assessment, University of Huddersfield, 7 July.

Kennedy, N. (1998), *WebMat – Experiences of Assessing LMU Students over the Web*, CTI Workshop on Computer Assisted Assessment, University of Huddersfield, 7 July.

Pownall, H. and Mobbs, R. (1998), *Integrating Computer Assisted Assessment and The CASTLE Project*, CTI Workshop on Computer Assisted Assessment, University of Huddersfield, 7 July.

Appendix

A questionnaire and its feedback from students, regarding the unit's Web site (see Table AI).

Table AI Students' feedback

	SA	A	N	DA	SD	NS
Q1	6	12	2	1	0	21
Q2	5	8	6	1	1	21
Q3	3	9	6	2	1	21
Q4	11	7	2	0	1	21
Q5	6	11	4	0	0	21
Q6	8	12	1	0	0	21
AS	6.5	9.9	3.5	0.7	0.5	

Q1. The Web site of this unit is useful.
Q2. The Web site of this unit is easy to access.
Q3. The password control mechanism of this unit's Web site is necessary.
Q4. The presentation of the lecture slides and tutorial notes of this unit's Web site is good.
Q5. The links to the self-asked questions of this unit's Web site are useful.
Q6. The communication facility (e.g. feedback, e-mail, comment) of this unit's Web site is useful.