

1997

## **WEBassess**

Justin Anthony Brown  
*Edith Cowan University*

Follow this and additional works at: [https://ro.ecu.edu.au/theses\\_hons](https://ro.ecu.edu.au/theses_hons)



Part of the [Software Engineering Commons](#)

---

### **Recommended Citation**

Brown, J. A. (1997). *WEBassess*. [https://ro.ecu.edu.au/theses\\_hons/1004](https://ro.ecu.edu.au/theses_hons/1004)

This Thesis is posted at Research Online.  
[https://ro.ecu.edu.au/theses\\_hons/1004](https://ro.ecu.edu.au/theses_hons/1004)

# **WEBassess**

by

**Justin Anthony Brown**

A Thesis

Submitted in Partial Fulfilment of the Requirements for the Award of  
Bachelor of Science (Communications & Information Technology) Honours

At the Faculty of Science Technology and Engineering  
Edith Cowan University  
Perth, Western Australia, Australia

Supervisor: Dr Arshad Omari

Submission date: 2<sup>nd</sup> of December, 1997.

2.4 Studies Similar to This Project	19
2.5 Literature on Methodology	25
2.6 Significance of Project	27
3. THEORETICAL FRAMEWORK	31
3.1 Project Development	31
4. METHOD	33
4.1 Design	33
4.2 Development	34
4.3 Actual Development	35
4.4 Procedure	38
4.5 The WEBassess System	39
4.5.1 Data Flows	40
4.5.2 Naming Conventions	41
4.5.3 Functionality	42
4.6 Testing	45
4.6.1 Student Testing Procedure	47
4.6.2 Staff Testing Procedure	51
4.6.3 Results of Testing	53
4.7 Limitations	55
5. FUTURE DEVELOPMENTS	58
6. SUMMARY	60
6.1 Design Goals and System Development	60
6.2 Conclusion	61
7. REFERENCES	63
8. BIBLIOGRAPHY	65

<b>APPENDIX A - Data Flows</b>	66
<b>APPENDIX B - User Guide</b>	69
<b>APPENDIX C - User Evaluation Forms</b>	84

## LIST OF TABLES

<b>Table 1: Student Evaluation Form Response Data Table</b>	50
<b>Table 2: WEBassess Staff Evaluation Data Table</b>	52

## LIST OF FIGURES

Figure 1: Test.com.Inc Visual Indicator Test Feedback Form	2
Figure 2: WEBassess Visual Feedback Indicators	3
Figure 3: Tutorial Gateway Testing Interface Example	20
Figure 4: John A. Kaliski's Online Classroom	24
Figure 5: John A. Kaliski's On-line Test Example	25
Figure 6: Webassessor Test Example	29
Figure 7: Example of Lasso Custom HTML Tags	37
Figure 8: Information Data Flow For WEBassess System	37
Figure 9: WEBassess HTML Files Information Flow	67
Figure 10: WEBassess Data Flow for Database Connectivity	68
Figure 11: Student Results Statistics Screen	42
Figure 12: WEBassess Admin User Menu Screen	43
Figure 13: WEBassess Staff User Menu Screen	44
Figure 14: WEBassess Student User Menu Screen	44
Figure 15: WEBassess Student User Evaluation and Consent Form	85
Figure 16: WEBassess Staff User Evaluation and Consent Form	86
Figure 17: Kandinsky Web Homepage for WEBassess	70
Figure 18: WEBassess User Login Screen	71
Figure 19: WEBassess Staff Menu Screen	72
Figure 20: WEBassess Question Pool Maintenance Screen	73
Figure 21: WEBassess Add Question Pool Result	74
Figure 22: WEBassess Pool Maintenance Screen	75
Figure 23: WEBassess Edit Question Screen	76

Figure 24: WEBassess Edit Question Result Screen	77
Figure 25: WEBassess Add Question Screen	78
Figure 26: WEBassess Add Question Result Screen	79
Figure 27: WEBassess Run Test Screen	80
Figure 28: WEBassess Run Test Score Result Screen	81
Figure 29: WEBassess Search Student User Screen	82
Figure 30: WEBassess Student Results Screen	83

## ABSTRACT

This project and research has resulted in the development of a World Wide Web (WWW) based testing system. WEBassess can be accessed and used from any point in the world that has a connection to the WWW. The WEBassess system allows for the creation and completion of tests through the WWW interface.

Scoring and feedback of submitted tests is completely automated, allowing students to see their test result immediately, including visual indicators for correct and incorrect answers. These test results may then be viewed by academic staff, showing complete test details, such as unit code, test title, score, percentage, date and time of test submission plus correct and incorrect answers. Through the logging of this information into the WEBassess database, profiles can be derived, indicating which students are having the most difficulty with certain questions. This information is stated against a student's identification number.

The entire WEBassess system can be administered through a consistent WWW interface, including the addition, deletion and maintenance of pools of questions. User maintenance is also included in this administration module, allowing for the addition, deleting and alteration of Admin, Staff and Student level users.

Security is implemented in the form of passwords linked to specific user identification details. A WEBassess user is presented with only the screens which their level of access allows, be it Administration (Admin), Staff or Student.



The WEBassess system has been developed using a variety of software and hardware tools, the most crucial of these being an application called Lasso (Blueworld Inc, 1997) and the Filemaker Pro 3 (Claris Corporation, 1997) relational database. The Lasso application communicates with the Filemaker Pro database via a WWW server.

The WEBassess system has been designed as a tool for student knowledge testing and feedback, rather than as a traditional assessment and scoring device.

Features found in the WEBassess system have been created to comply with specific requirements for Edith Cowan University staff and students, whilst other features have been derived from similar testing systems currently in use on the WWW.

The WEBassess system is presently at the level of preliminary user testing, from which initial changes to the system have already taken place. Several future developments for the system are envisaged, to be implemented after a prolonged period of end-user testing on the finalised program.

## DECLARATION

I declare that this thesis does not incorporate, without acknowledgment, any material previously submitted for a degree or diploma in any institution of higher education and that, to the best of my knowledge, it does not contain any material previously published or written by another author or organisation, except where due reference is made.

Signature:

Date:

2/12/97

## **ACKNOWLEDGEMENTS**

I would like to express my gratitude to my supervisor, Dr Arshad Omari, for his support and advice during the development of the WEBassess project and this written documentation. Dr Omari's knowledge, patience and understanding have contributed in a large degree to the completion of this project. I would also like to thank Dr Gulden Wagner for her valuable advice and direction in the writing of the thesis document, whilst also stating my appreciation to Judy Clayden for efforts in proof reading this work. Grateful thanks also go to Pamela Phillips for her patient help and assistance in the everyday development of this project.

## DEFINITION OF TERMS

WEBassess is based around Internet and communications technology. Thus there are several terms which will be used frequently during the application's development.

The most widely used terms are defined below:

- **Browser:** An application program designed to display information from HTTP servers in order to place visual information before the user. Browsers recognise tags embedded in HTML documents and lay out the screen according to the instructions given by those tags. Browsers are also capable of displaying information from other protocols and programs, such as FTP, Gopher, NNTP and E-Mail. Definitions of these terms may be found below.
- **Internet:** The Internet is the global network of computer networks which talk to each other via international communications lines using a standard communications protocol known as TCP/IP (Transmission Control Protocol/Internet Protocol).
- **TCP/IP:** TCP/IP connects computers from one point to another by routing data through intermediate networks between the two communicating computers, each network receiving the information and directing it onwards until the destination is reached.
- **World Wide Web (WWW):** Many people use the terms World Wide Web and Internet interchangeably. This is not strictly true, as the World Wide Web is a visual presentation system that uses the Internet framework as a transport medium.

The WWW allows the display of textual, visual and multimedia information using a browser program.

- **Homepage**: The first page that a web browser presents when connecting to a WWW server. The homepage is the electronic equivalent of the front door, from which other areas of the site can be accessed.
- **HTTP**: The WWW uses HTTP as a protocol to transmit visual and textual data, whilst using the Internet based TCP/IP protocol as a medium through which to transmit this information. Client request and server responses are defined using the HTTP protocol.
- **HTML**: Hyper Text Markup Language is the coding used by WWW documents to describe the content and appearance of a 'web page'. WWW browser applications interpret the information contained in the HTML document and display this according to the markup. There have been several revisions of HTML, the latest being version 3.2. With each revision, new functionality is added. This means updated releases of WWW browsers are required in order to interpret the new HTML tags.
- **URL**: The Uniform Resource Locator is an addressing scheme used to define the location of a resource on the Internet. Essentially, the URL defines a specific resource on the Internet according to the host computer address and the path to the resource on that computer. Client computers can find resources on host computers because all machines running as hosts have names assigned to them, in

a similar manner that names in a telephone directory have phone numbers assigned to them. Once the name has been found, so has the corresponding number. FTP and HTTP are two of the main methods for downloading and viewing information on the Internet, respectively.

- **Unix:** Unix is a command line based, high speed network oriented operating system which has been available in several different 'flavours' since the late 1960's. Unix is still considered the defacto Internet server and development operating system.
- **Filemaker Pro 3:** A relational database program used to store, retrieve and manipulate the data used in the WEBassess system.
- **Lasso:** A Macintosh plug-in to a web server application which provides the ability to access, update and manage a Filemaker database over the Internet.
- **FTP:** Acronym representing File Transfer Protocol. This protocol is a set of rules used between a client and server computer to send and receive files between the two machines. FTP is the standard used to send and receive files over TCP/IP based networks.
- **TELNET:** Acronym representing TELeNETworking, which allows remote computers on a TCP/IP network to access the files and services of another machine on the network.

# 1. INTRODUCTION

## *1.1 Background*

This research has resulted in the development of a system called WEBassess, which allows students to sit on-line, multiple-choice tests over the Internet / World Wide Web from any computer connected to the net. The research environment, described here, has resulted in a system that allows a student, upon completion of a given test, to receive a grade immediately, through the automated testing program. An automated testing system such as WEBassess is designed for use in conjunction with coursework materials as a feedback and analysis tool for lecturers and students alike.

Some of the issues associated with this project include the security of the system, in the context of preventing unauthorised access to the system's question and test creation facility. The question management aspect of the program will consist of a staff-only interface that is used to input, edit and delete question pools. This staff-only interface will also allow for the addition of new users to the system.

Tests are created from the information in the question pools and presented to the student in a multiple-choice format. Once students have selected all their responses and clicked the Submit option, their grade for the test is returned. Visual indications of correct and incorrect answers are included, as are explanations of answers deemed incorrect. This allows students to identify potential problem areas in their understanding of the topic tested.

The use of visual indicators for correct and incorrect answers was modelled on a concept taken from an on-line testing system produced by Test.Com.Inc [http://www.test.com, 1997]. In this system, the actual test form is presented to the user after test submission, with each incorrect answer being coloured red. Correct answers remain the original yellow colour. See figure below:

### The Inventors and Inventions Test

People have been inventing and discovering things throughout time. This quiz tests your knowledge on some of the oldest inventions and discoveries as well as the greatest inventors.

Question#	Question Statement
1	<i>Which brothers invented the first hot-air balloon?</i> You answered <b>The Montgolfier Brothers</b> , which is correct. <span style="float: right;">points = 1</span>
2	<i>What city erected the first red - green traffic light?</i> You answered <b>Boston</b> , the correct answer is <b>Cleveland</b> . In 1914 Cleveland, Ohio used the two-color traffic light.
3	<i>Who patented the first electric razor?</i> You answered <b>Jacob Schick</b> , which is correct. <span style="float: right;">points = 1</span>
4	<i>Which civilization first used concrete?</i> You answered <b>The Greeks</b> , the correct answer is <b>The Romans</b> . The Romans were known to have used concrete in the 6th century
5	<i>Pasta was first made in what country?</i> You answered <b>China</b> , which is correct. <span style="float: right;">points = 1</span>
6	<i>During WW II, the microwave oven was originally created for</i> You answered <b>radar detection</b> , which is correct. <span style="float: right;">points = 1</span>
7	<i>During the French Revolution in 1809, Nicholas Appert won twelve thousand francs for his practical method of preserving food. Which of the following was his invention?</i> You answered <b>Glass jars with corks</b> , which is correct. <span style="float: right;">points = 1</span>
8	<i>Alexander Graham Bell, in 1876, demonstrated that telephone communication between people over a distance by electricity was possible. What were the first words that were transmitted between Bell and his assistant?</i> You answered <b>"Mr. Watson, come here - I want you."</b> , which is correct. <span style="float: right;">points = 1</span>
9	<i>The Polaroid camera was invented in 1947 by which American?</i> You answered <b>Joseph Bell</b> , the correct answer is <b>Edwin Land</b> .
10	<i>19th century French chemist, Louis Pastuer developed pasteurization to treat what food?</i> You answered <b>milk</b> , the correct answer is <b>wine and beer</b> .

Figure 1: Test.com.Inc Visual Indicator Test Feedback Form.

Though WEBassess uses a slightly different paradigm for student feedback, it does use visual indicators for correct (green tick) and incorrect (red cross) answers. See figure on following page:



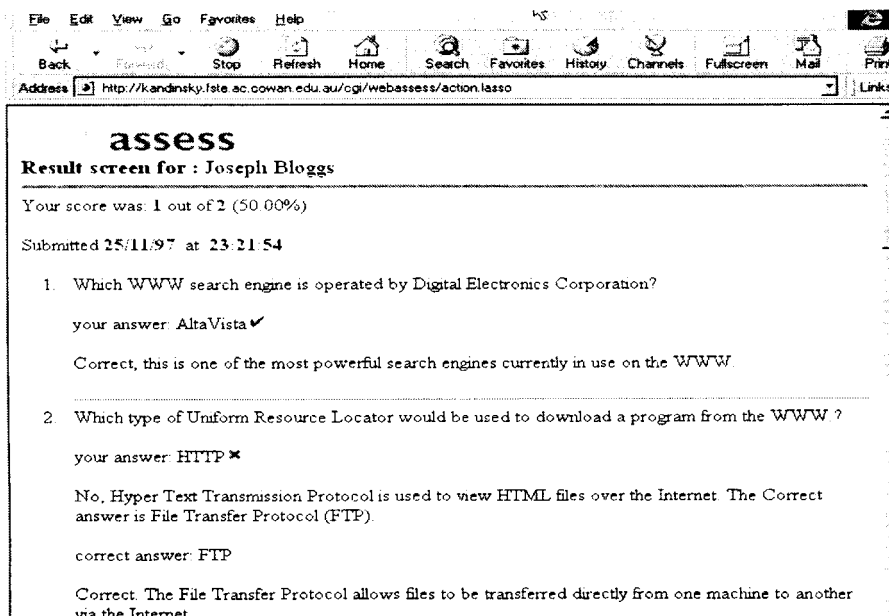


Figure 2: WEBassess Visual Feedback Indicators.

## 1.2 Rationale

As the WWW gradually becomes a legitimate research and learning tool, it seems inevitable that existing coursework materials are going to find their way onto the net. The WWW offers several benefits as a learning medium when viewed against traditional education methods. The most immediate benefit is the interest and enjoyment many people experience from using the WWW. Due to the currency and expansiveness of the net, many people, students and lecturers alike, make the WWW their first option when searching for a piece of new, obscure or irreverent information. By bringing unit materials to the on-line forum, interest in actively participating in learning activities should increase.

Ease of materials preparation and distribution is another factor that needs to be taken into account when considering the WWW as a teaching environment. Due to the popularity of the WWW, most new 'office suite' products allow word processing,

spreadsheet and database files to be written directly to HTML format, which can then be placed onto the WWW. This means that only one copy requires production, and that updates can be made straight to the file residing on the WWW. Interactive elements can be placed into these on-line materials, such as videos, animations, sounds and graphics. These can be used as demonstration and learning aids, or simply as tools to capture and hold the attention of the student audience.

Within such an environment, WEBassess could be an exceptionally useful tool, being used in concert with the WWW based materials. Students will work through a series of activities, using the on-line coursework material, after which they can sit an on-line test based on that material. WEBassess will allow students to measure how well they absorbed the on-line information, offering positive reinforcement for correct answers, whilst offering direction and explanations for incorrect answers. As Steinberg (1991, p67) states, "The courseware on the Web must follow the well-established theories of learning, and any lesson on the Web should incorporate some form of knowledge testing".

As stated within the Test.com.Inc homepage [<http://www.test.com>], 1997] "A test is a standardised sample of behaviour from which other, more important behaviours can be inferred or predicted". It is from the behaviour of students conducting coursework based testing that problem areas in the running of units, and patterns of student learning, can be viewed and refined where necessary.

### ***1.3 Computer Based Testing***

From as early as 1982 (Bugbee, 1990), computers have been employed for the purpose of administering tests using multiple-choice examination techniques, where the student is presented with a list of possible answers to a given question, one of the answers being correct. In such a system, the student selects the one answer they feel is correct for each question until the examination is completed. The system then provides a printout of the student's grade according to the number of questions answered correctly.

#### ***1.3.1 Test Procedures***

Multiple-choice tests are usually conducted in one of two ways. A student sits down in front of a stand-alone computer onto which the examination, complete with answers, has been loaded, or the test is delivered to the student's terminal via a network server or mainframe computer. This computer can be located in the same building, in another state or even another country. Both of these configurations require the student to be physically present at a set testing centre where the computers, software and supervisors are located.

The American College has been operating a system known as Examinations On Demand since the end of 1982. Students may sit an examination in a subject when they feel sufficiently prepared, at which time they make arrangements to do their test at a convenient testing centre. By 1987, 265 000 such tests were being completed by the College's students each year (Bugbee, 1990).

Other organisations have adopted the same testing methodology used by the American College, where students are required to sit electronic tests in a centralised location.

Another similar system is found at the WWW homepage [<http://www.ifebp.org/cecbt.html>, 1996] of the International Foundation of Employee Benefit Plans (IFEBS), discussing in-house Computer Based Testing. This system allows students to schedule a computer-based test at any one of 240 locations within the United States and Canada.

### ***1.3.2 Current Possibilities***

The field of computer based testing has been well established and developed for more than a decade, though up until very recently such testing systems have been reliant on the 'in-house' method of exam delivery. In-house testing systems require students to actually attend a specific testing centre, where the computer and test are contained in a supervised environment.

Less than five years ago, a medium became available which presented an alternate method for delivery of electronic examinations, the World Wide Web. The last two years have seen the development of powerful WWW 'browser' applications which can handle interactive forms and web based programs, independent of platform or operating system.

It is the integration of Internet hardware and browser software which now offers a new medium by which to deliver electronic, on-line examinations and tests.

## ***1.4 Objectives***

The purpose of this project was to develop a fully functional Internet based test delivery system that allows students to sit tests from a computer connected to the World Wide Web.

WEBassess was designed to incorporate two main modules, one for students sitting on-line tests and one for lecturers to create and edit tests. For both modules, ease of use and security were the primary areas of concern for the final implementation of the project.

### ***1.4.1 Security***

The development of an effective security procedure for staff verification was crucial for a testing system such as this, as the system's usefulness would be negated if students were able to access the staff-only area of the program. This security system was implemented in the form of password and user identification based verification.

### ***1.4.2 Automated Score Calculation***

A key component in the development of the WEBassess system is the implementation of an automated test grading system, where students receive their marks for a test the moment the completed test is submitted. This scoring system gives the student a percentage mark for the submitted test, plus a total of questions correct out of the total number of questions asked. Visual indicators are presented for correct and incorrect answers, as are explanations and remarks for each.

### *1.5 Problem Questions*

Projects such as WEBassess are usually undertaken to produce something useful or answer a specific question. Though this thesis will mainly concern the development and implementation of the WEBassess system, three fundamental questions arose.

- Will the development of WEBassess lead to an enriched learning environment through on-line materials, or will it simply be a different method for delivering teaching and assessment?
- How can the issues of security and prevention of student cheating be addressed? Although this system is currently only to be used as a coursework based testing and feedback tool, it is still necessary to perform at least a minimum level of user authentication. This is especially important when dealing with the test creation facility within WEBassess.
- Is there a definite need for such a system? Is this type of testing relevant to the needs of an institution such as Edith Cowan University, and will it be used as an eventual replacement of the pencil-paper based test or will it simply be used as an adjunct to the existing system?

These are some of the questions that are examined during the course of this work, specifically in the area of project development.

## 2. LITERATURE REVIEW

### *2.1 General Literature*

Upon commencing a review of the literature for this project, it became apparent that existing reference material was going to fall into two distinct categories, testing in general and computer based testing.

The subject area of testing in general covered the definition and makeup of a test, including the following:

1. What is a test? The definition of a test varies depending upon the context in which it is used. Within the literature on education and teaching, a test is seen as a tool used to obtain a student's ability to learn and comprehend. The test gives a quantifiable figure indicating student achievement over a course of study in a specific topic. When viewing literature relating to testing and computers, the definition of testing is somewhat different, being seen more as a means of ascertaining a piece of software's or hardware's working status. Both are measures of performance.
2. Why is testing used? Testing is used to gauge performance, of both people and devices. The subject of a test can usually rate between 0 and 100 percent on a scale relative to the item being tested. Appropriate actions in a given situation can be taken in accordance to the results of a test. If the subject of a test falls within a defined 'passing' range, it passes on to the next level of expectation, whilst those falling below will have to try the test again or

proceed no further. Testing is used as a filtering mechanism against what is defined (by the test) as unsuitable for a given task or operation..

3. How is it implemented? Tests can be performed in any number of ways, each depending upon the test subject and the data that is to come as a result of the test. In the field of learning and education, a test is usually implemented in the form of questions, these questions relating to information and skills the student is expected to have acquired over a period of study and practical work. Tests usually comprise a series of multiple-choice questions, followed by short-answer questions and an essay component, all within a limited period of time.
4. What is in the test? Tests usually contain questions relating to information the test subject is expected to possess. These questions will also be of a quantifiable nature, so that a definitive score can be assigned for each answer given. Multiple-choice and true/false type questions are the most empirical forms of test implementation, as there is a definite answer, correct or incorrect, with no ambiguity or 'grey' area, such as found in essay questions. Ramsden (1992, p195) feels that "Multiple-choice questions provide another excellent opportunity to offer feedback in an efficient form".

Another factor examined the general literature is the perception of what constitutes a well-designed test. The Test.com.Inc homepage states that:

“A good test must have these three characteristics:

1. It must be highly standardized - the conditions under which it is administered must be clearly and completely specified so that everyone taking the test is treated exactly the same.
2. It must be reliable.



3. Most of all, it must be valid.

Test administration and scoring represent the first steps in obtaining data for making judgments and decisions about people.”[<http://www.test.com>,1997].

The literature covering testing was very extensive and detailed. Multiple-choice testing was widely discussed in this literature, more for its benefits and drawbacks as a means for statistical testing of students than as a methodology for computer based tests. However, those that did discuss testing in relation to computer based education felt that multiple-choice questions provided the optimum method for student testing and assessment. Overbaugh (1994, p29-47) feels that "Web-based educational systems, like other computer-based education (CBE) systems, must provide certain basic instructional functions". Gagne and Aronson (1983, p75-100) state three of these functions as including:

1. evaluation of the student's understanding of each concept,
2. provision to the student of feedback concerning his/her performance during the evaluation, and
3. assessment of the student's complete understanding of each concept.

Multiple-choice testing is a recognised and widely used method of assessment in traditional educational systems, such as the classroom and paper based tests, and can be implemented in almost any situation where knowledge testing is required. It is for this reason, and the suitability for automated test grading, that the multiple-choice format has been incorporated as the core testing methodology within the WEBassess system.

## *2.2 Computer Based Testing*

Computer based testing describes the use of computer software and hardware to deliver a test to a student within a specific topic or knowledge area. The key factors influencing the use of computers as a testing and assessment platform are the ability to have tests marked automatically by the system, whilst also allowing users to sit a test at a time convenient to them.

"Computer administered tests are becoming a major force in measurement" (Bugbee, 1990, p87-100). This quote taken from a journal article examining the issues involved in computer based testing indicates that traditional paper and pencil testing is no longer the only way of accurately gauging student knowledge and course effectiveness. Even as long ago as ten years "approximately 265,000 tests were administered by computer" in the United States, according to figures gathered by Bugbee (1990, p87-100).

Bugbee bases his observations on the example provided by the American College in Pennsylvania, Philadelphia, one of the United States' oldest institutions for higher education in finance and business. In October, 1982, the college introduced Examinations on Demand (EOD), a system which allowed students to sit an electronic version of an examination whenever they felt capable of sitting the test. These tests were administered via an 'in-house' computer system, which required students to physically travel to a computer testing location run by Control Data Corporation. The students would sit the examination at an individual terminal and, upon completion of the test, receive their grade immediately, rather than wait the standard six weeks.

A similar system, known as TESTER, is operated by the University of Waikato, New Zealand. Students are presented with a series of randomly generated multiple-choice questions, results being made available immediately upon submission of the test. A journal article by Taylor (1983, p76-80) discussing this TESTER system states that “the system is not dependent on staff presence”, a major factor in the use of computers as testing tools.

### ***2.2.1 Scoring Comparisons***

One of the main focuses of Bugbee's article involved the relationship between scores achieved by students sitting the EOD and normal paper and pencil based tests. Students sitting the computer based test seem to achieve slightly higher results than those sitting the paper and pencil based test, but the score differences over a prolonged period of time are marginal at best. When viewing computer systems as a means for providing automated tests, some believe that the computer is only an efficient tool, and that actual student learning is not improved but more readily facilitated.

A 1979 paper by Vitale and Cole (p 15) discusses a computer based testing application known as the Teaching Information Processing System (TIPS). Vitale and Cole found that “although learning did not significantly increase, the climate for learning improved and students were enthusiastic about the system”. While this statement was made of a specific testing product, it does stand to reason that some systems, depending upon their design and useability, would not actually increase the users' ability to learn, but simply augment it. Whether WEBassess is indicative of this

type of testing system remains to be seen, perhaps after several months, or even years, of continuous use. Statistics presented by Bugbee (1990) indicate that initially students with a higher degree of computer literacy performed better in the computer based tests than those with lesser computer literacy. However, over a prolonged period of time this variable was eliminated, due to the college's intensive use of computers in all their course components.

This is relevant to my project as students and lecturers require only a rudimentary knowledge of computers in order to operate the WEBassess system. Lecturers require the ability to use a mouse for on-screen navigation and need to possess basic typing skills in order to enter unit, student and question information.

Students are only required to type in their student number when logging onto the system, after which all program interaction is handled via the computer mouse. Screen layout and design has been implemented in a fashion that allows for ease of use and efficiency. Buttons and graphic icons have been specifically designed so that their purpose is obvious to any user. In most cases, buttons and icons have text labels so that there can be little room for misinterpretation of the object's function.

Bugbee (1990) presents statistics that attempt to show which students are scoring better, those using the computer based test or those sitting a traditional paper based assessment. At first, this appears to be a relevant piece of data to be taken into account when designing the WEBassess system, but on closer inspection this type of statistical information is only relevant when a system is being used as a certified assessment and scoring system.

As WEBassess has been designed for comprehension testing and feedback rather than as a course accreditation testing system, such concerns are currently considered as outside the purview of this work. Developers of computer based testing systems have also taken this attitude, as Taylor (1983, p76-80) agrees when he says that "The facility to extend the system to include it for assessment purposes has been discussed, but at present the primary purpose of TESTER is seen as being of a self-help nature". Taylor's TESTER system was developed well before WEBassess, and tackled similar questions of testing and assessment.

The WWW homepage of the International Foundation of Employee Benefit Plans (IFEBS) [<http://www.ifebp.org/cecbt.html>, 1996] discusses in-house computer based testing and outlines findings similar to those of Bugbee (1990), especially supporting the idea that students take tests when they are ready and describing the benefits of receiving immediate feedback on a student's examination performance. In addition, the IFEBS Web site states that students taking computerised tests enjoy a "quieter and more secure test environment", though the rationale behind this statement is not made clear.

### ***2.2.2 Cost Efficiency***

Another key benefit of computer based testing described in the literature is the potential for cost efficiencies. An excerpt from a *Case Study: Intranet Based Testing and Assessment*, a homepage contained on the Advanced Consulting Inc WWW site [<http://www.acihome.com/case.html>, 1997], claims that "This new Intranet-based

testing solution has saved both time and money", and that "expenses associated with duplication have been totally eliminated". The actual system allows salespeople from a pharmaceutical company to download a test onto a laptop computer from the company's central system, complete the test at their leisure, then upload the result to the central computer for marking. Results are then E-mailed to the salespeople automatically.

Though cost efficiency is not an integral concern of the WEBassess system, there is always room for improvement in all institutions, be they academic or business, to actively lower costs while increasing efficiency and productivity. The potential saving in eliminating paper based copying, individual manual test marking and mailing costs would definitely be a value added component to any WWW based testing and education system. Though WEBassess is introducing a new service rather than replacing an existing one, the topic of cost efficiency is still worth considering.

### ***2.2.3 Portability***

In conclusion, there is an issue which, while not crucial, required some attention during the development of the WEBassess system. This relates to the portability of the system, as "it is difficult to develop a product to be used at more than one college if objectives and competencies for a course differ from college to college". This quote, taken from the Maricopa Center for Learning and Instruction web site [<http://www.mcli.dist.maricopa.edu/ocotillo/report94/rep11.html>, 1997], underlines the question of whether a web based testing system developed for a university such as

Edith Cowan University will be relevant in structure and scope for use in other tertiary institutions.

If the end product proves successful and useful for one institution, it may be attractive to others. Thus, the flexibility and customisation features of the system should be such that other institutions and businesses can adapt the system to their own needs, with the minimum of re-engineering. Though WEBassess has been designed specifically to fulfil a purpose at Edith Cowan University, there is always the possibility at some point that other organisations may wish to adopt the system through some agreement with this University.

Theoretically, the entire WEBassess application could be transported across to another Macintosh based system running a web server, Lasso and Filemaker Pro 3. Once these items were set up and configured, there is no reason why the WEBassess application would not work exactly as it does within this organisation.

There is another possibility for WEBassess to be used by an external organisation, and that is to simply create and run the tests from an Internet connected machine at the organisation's own offices, whilst the system itself resides on an Edith Cowan University server.

This is a concept that is currently being used by the ComputerPREP Corporation [<http://www.webassessor.com>, 1997]. "You can author and administer your assessment device or data collection instrument globally via the Internet (or an Intranet). This gives you worldwide reach, lowers costs and improves productivity for

all involved". As this extract from the ComputerPREP homepage indicates, their own web based testing system can be run externally over the Internet, or installed and configured on a local intranet running the appropriate server software.

### **2.3 WWW / Internet Based Testing**

When the initial literature review for this project was conducted, the final role of the WEBAssess system was still in a state of flux, so several areas of research were examined, some of which are no longer relevant. Of these research areas, however, security for on-line testing was considered the most important, as WEBAssess was originally to be considered as a tool for students to sit assessable exams over the WWW.

This objective was fraught with any number of security concerns, particularly in regards to student authentication and the use of on-line reference materials during an exam. It is not relevant to the WEBAssess project in its current form to go into greater detail concerning the security problems. Suffice it to state that complications were many, and solutions would have been far beyond the maturity expected of the WEBAssess program for some time to come.

Few of the sources located covered security in any notable detail, and those that addressed it did so only in the context of hidden directories and passwords. The Harte Hanks web site *On-line Testing* [<http://www.harte-hanks.com/demo/testing/>, 1997] raised an interesting security problem when they "discovered that test takers were pressing the 'Back' button after receiving a perfect score, changing the name of the applicant at the top, and re-submitting the same response".



This is a factor that could have been easily overlooked, but has now been taken into account in the final design of the testing interface in the form of date stamped tests. When a lecturer conducts a search of a student's results via their student number, all the tests they have submitted will be listed, including the dates and times of those tests. If the same test appears more than once within a few seconds or minutes of each other, the lecturer will know that the student has re-submitted the test. As tests are written into the database in a linear, chronological order, the lecturer can judge which test was submitted first by the position in which it appears in the submitted test list.

Students cannot simply click the 'Back' and submit the completed test under another student's number, as this would require logging into the system again, and running the test anew. Though this is possible, if a student knows another student's identification number and password, neither student would stand to gain much from such a collusion, as test results have no actual bearing on a student's unit grade. In fact, such behaviour would in fact be detrimental to a student, as they would not gain the benefits of the feedback associated with test results.

#### ***2.4 Studies Similar to This Project***

A site which proved extremely useful to my research was the Carleton University Department of Civil and Environmental Engineering's [<http://www.civeng.carleton.ca/~nholtz/tut/doc/doc.html>, 1997] discussion of *The Tutorial Gateway*. The Tutorial Gateway is a testing system designed by Carleton University to test its students via the WWW, and is similar to what I hoped to achieve with the WEBassess project.

The information presented at this site goes into fine detail concerning the design and implementation of The Tutorial Gateway, including examples of the Hyper Text Markup Language (HTML) code used to create the various components of the system.

The site also includes some interactive samples of existing tests, allowing the viewer to take a cut down version of several tests in order to see the system working. See following figure:

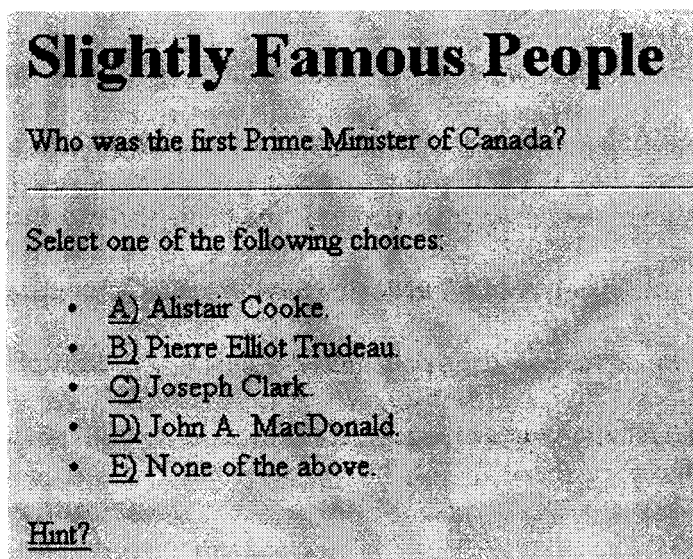


Figure 3: Tutorial Gateway Testing Interface Example.

The Tutorial Gateway is compared to similar systems being used at other locations, which proved invaluable for finding like testing systems and examining the differences in design and development. Interestingly, when the original literature review for the thesis proposal was conducted in March 1997, there were at least four sites listed for comparison with the Tutorial Gateway. When the site was visited again for the purposes of this literature review, only one link was still listed. What this says about the tenuous nature of the WWW, or of on-line testing, is open for interpretation.

An interesting feature of the Tutorial Gateway is the availability of 'hints' for students sitting tests. When this hyperlink is clicked, a hint appears next to the option, displaying some text designed to help the student ascertain the correct answer. At present, it is difficult to imagine such a feature being implemented in WEBassess, as students may be tempted to forgo the coursework component of the system and simply rely on the hints.

However, the reverse of this could also be true, where hints are used to promote students to complete the work attached to a WEBassess test. For example, a hint may be worded in the following manner: "The correct answer can be found in the History of Computing section of your on-line coursework materials, discussing pioneers in the field of digital computers". A 'loaded' hint such as this would require students to find the relevant section in the coursework material, and eventually lead them to complete the assigned research.

In order to test the positive and negatives of the hint system, the feature would need to be added to WEBassess after several months of constant use in its current form, thus allowing any difference in student performances to be more readily identified.

The Carleton University web site has the feel of an open forum for discussion, where interested people can see what developments are being made in the area of on-line testing systems.

One particularly propitious aspect of this information source is its broad coverage of the different types of testing systems capable of being implemented in a web centric manner. These testing types include:

1. multiple choice,
2. true or false,
3. one word, and
4. single numerical values.

For the purposes of WEBassess, the section on multiple-choice testing was most relevant, though the information regarding the other testing types will be a valuable resource pursuant to the functionality of WEBassess being expanded in the future. The only information lacking at this site is any reference to issues of security.

The most relevant of all the sources I located was the Ausweb 97 web site [<http://elmo.scu.edu.au/sponsored/ausweb/ausweb95/>] which contained an entire conference paper titled *The Development of a Multiple-Choice and True-False Testing Environment on the Web*.

The information contained within this paper was particularly relevant to the development of WEBassess as it described the essential core components of any on-line testing system. These components include:

- grading,
- tutorial building,
- tracking, and
- implementation.

Multiple-choice and true or false questions are the most effective types of questions which can be used for such a test if the process is to be fully automated, that is, if the students will receive their grades within moments of submitting the completed test. Other possible question types were discussed on the previous page, and will again be addressed in the Future Developments section of this document.

John A. Kaliski, of Mankato State University, discusses on-line testing within the scope of entire on-line learning environments, such as web-based classrooms and university courses. A review of Kaliski's testing system taken from a World Wide Web discussion bulletin board [<http://instruct.unc.edu:6080/wcb/schools/5/3420/kthomas/new11961/forums/forum1/messages/13.html>, 1996] states that "On-line testing is becoming more and more appealing as the use of technology increases in the classroom. There are the obvious advantages of ease of administration and grading that can save professors and their administrative staff (if they are lucky enough to have them) significant amounts of time". Due to the integration of on-line testing with on-line teaching, this site seemed a perfect candidate for examination.

Unfortunately, the entire on-line testing and classroom site [<http://web.business.mankato.msus.edu/Online.Classroom.html>, 1997] is protected by a series of passwords and user identification, so only students of the Mankato University can gain access. See following figure:

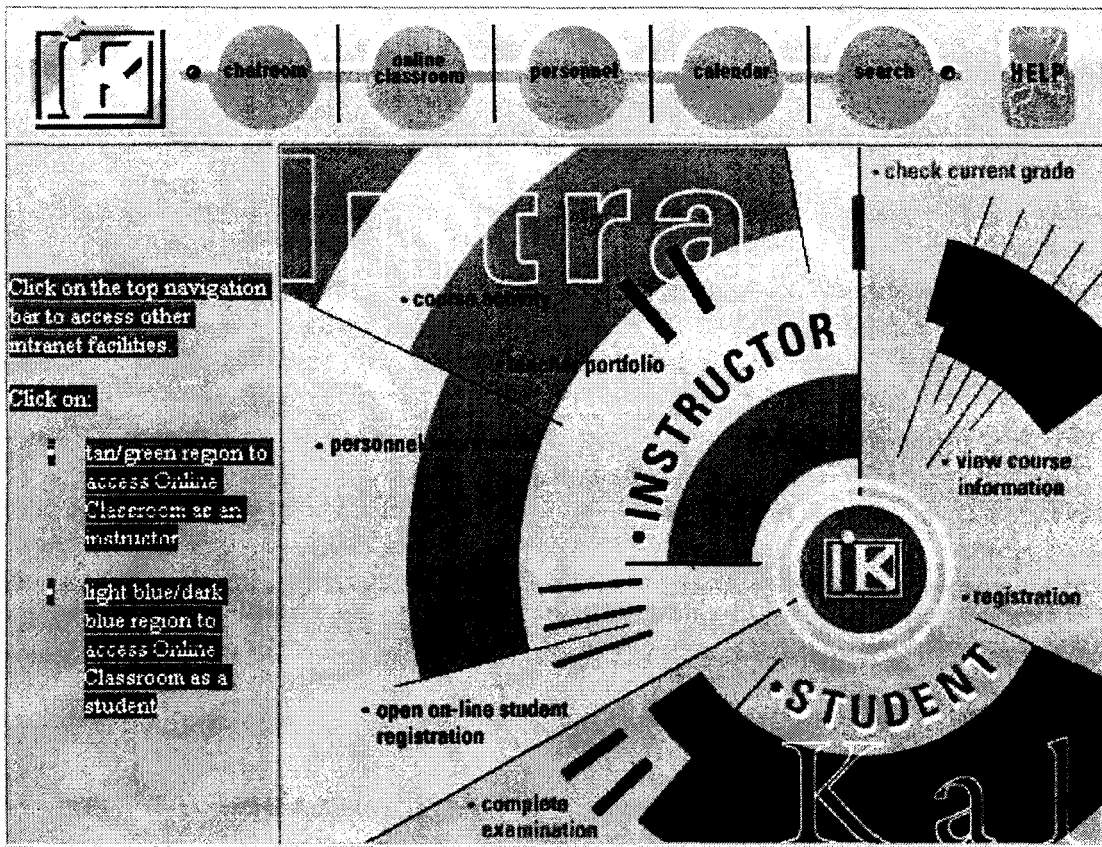


Figure 4: John A. Kaliski's Online Classroom.

However, several examples of previous tests were made available for the general public to view at the WWW site [[http://krypton.mankato.msus.edu/~johnk/mgmt.200/test\\_list.html](http://krypton.mankato.msus.edu/~johnk/mgmt.200/test_list.html), 1997

]. An example of the way these tests are presented to the student is shown in the figure on the following page:

**MGMT200 -- Introduction to MIS**  
**Winter 1997**  
**Test 1**

1. Office automation systems support:
  - c) No answer given yet
  - a) the accounting department
  - b) external suppliers
  - c) middle management
  - d) customers
  - e) all of the above
  
2. GDSS have addressed the following problems of traditional meetings except:
  - c) No answer given yet
  - a) lack of adequate documentation
  - b) lack of priorities and formalized decision making techniques
  - c) inadequate management influence in meetings
  - d) inadequate pre-planning
  - e) criticism of the contributor rather than of the contributed idea
  
3. An Executive Support System will typically automatically interface with many other systems to supply executives with the most up to date information.
  - c) No answer given yet
  - a) True
  - b) False
  
4. Which of the following systems would be most helpful to a business person considering opening up a new branch office in a distant city.
  - c) No answer given yet
  - a) transaction processing system
  - b) management information system
  - c) decision support system
  - d) office automation system
  - e) knowledge work system

Figure 5: John A. Kaliski's On-line Test Example.

This review of the literature must be concluded by stating that even though relatively few printed references were found, the information I gathered from the Internet was very useful overall.

### ***2.5 Literature on Methodology***

All modern research conducted at a scientific or academic level must adhere to or be based upon a recognised methodology or procedure by which to derive answers to a given problem. If research is conducted according to an ad-hoc series of steps, then

any results from that research must be deemed questionable due to the lack of accountability for each decision made during the work and a lack of result replication.

As this thesis covers the design, development and implementation of the WEBassess on-line testing system rather than actual research into the implications of such a system, there is no methodology which really complements the project as such. The methodology which comes closest to having direct relevance to this project is Action Research, as it is "practical and directly relevant to an actual situation in the working world" (Isaac, 1971, p99). "Action research is more systematic and empirical than some of the other approaches to innovation and change" (Mason & Bramble, 1978), which is exactly what is required when developing a new, technology based application such as WEBassess.

Action research sets out to provide a working solution to a given problem via the direct application of existing knowledge combined with new theories and/or hypotheses developed during the problem resolution process. In the case of WEBassess, a WWW based testing system was developed in order to solve the problem of how to test students via the Internet.

Using existing knowledge I have located, such as the information found at Carleton University's Tutorial Gateway [<http://www.civeng.carleton.ca/~nholtz/tut/doc/doc.html>, 1997] web site, in conjunction with my own implementation of the WEBassess system using HTML, Lasso and Filemaker Pro 3 database, a new piece of knowledge will be added to the field of on-line testing. This knowledge will cover more the development and implementation issues rather than the impact on testing as



a whole. Leedy (1993) states that "The approach in action research is to do something to see if it works". For this reason I feel action research has provided the best methodology for successfully completing the WEBassess project, though only through a very broad interpretation of what constitutes the principles of this methodology.

## ***2.6 Significance of Project***

This project is significant for what it offers in the way of new services, new teaching practices, efficiency and use of WWW technology.

The project allows Edith Cowan University students to sit multiple-choice tests from anywhere in the world that has an Internet connection available, be it via a personal Internet Service Provider (ISP) or a dial-up account provided to the student by the University.

The WEBassess system can be used by external students studying abroad or in remote locations, as well as for testing students internally from any one of Edith Cowan's metropolitan or regional campuses.

Due to the automation features used in WEBassess, a unit lecturer need only create a test and lodge it in the system. From this point onwards it is the student's responsibility to work through the research materials in the classroom, before sitting the relevant WEBassess test, most likely during a weekly unit workshop.

Of the Internet based testing systems currently available, the main applications that stands out in the areas of useability and similarity to WEBAssess are Carleton University's Tutorial Gateway, Test.com.Inc's testing system and ComputerPREP's Webassessor. These systems are designed to offer tests and examinations via the WWW using form based, multiple-choice testing interfaces. These systems are fully automated in the areas of student testing and exam creation, meaning that neither the student nor the unit lecturer requires any knowledge beyond an understanding of how the WWW browser applications work.

These systems and the way they use WWW and HTML technology offer an excellent reference point for the development of the WEBAssess system. Information from these systems have impacted several areas of the current WEBAssess system design, specifically those of test presentation and feedback.

The Webassessor system from ComputerPREP Corporation is unique in the testing programs reviewed thus far in that it allows for the inclusion of multimedia elements, such as pictures, animations and sounds, plus the ability to mark each question individually as a test progresses. Figure 6 shows the layout of this testing system, with navigation buttons and timer being contained in a separate window at the bottom of the screen.

This system is also unique in that it treats each multiple-choice question individually, rather than presenting a scrolling series of questions, as is used in WEBAssess.

During the early stages of development, WEBAssess was also going to use the one question per page option, in order to allow for easier test navigation, such as knowing

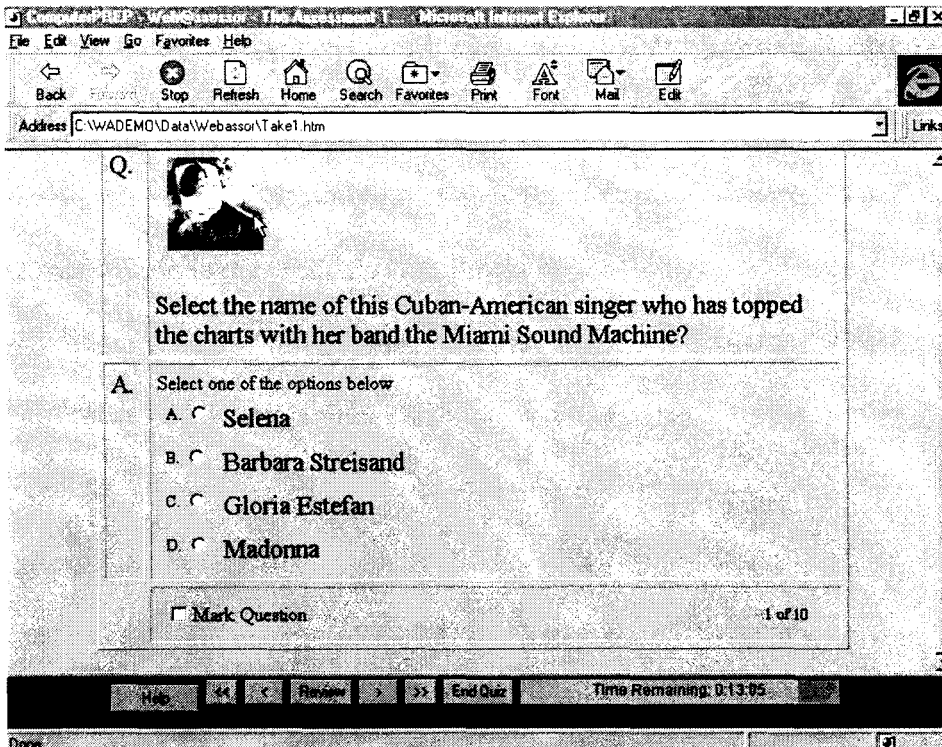


Figure 6: Webassessor Test Example.

how many questions had been answered out of the total number to be answered.

Three factors influenced the decision not to proceed with this concept. These are:

1. As no multimedia elements were intended to be incorporated into the WEBassess system at that stage, one question per screen was an inefficient use of available screen space.
2. Once Lasso and Filemaker Pro 3 were selected as the final development tools, problems occurred with submitting one question at a time. The scoring system needs to receive all answers to all questions in one database record, ie, one click of the Submit button on the test form.
3. The overall process of presenting then submitting each question individually slows down the entire testing process to a considerable degree.

These considerations ensure that the WEBassess system will be significant as a future reference for the development or other such WWW based testing and education systems, including the problems faced and how they were overcome. This is particularly relevant given the number of changes in this project within a period of six months, particularly in the area of development methodology.

### **3. THEORETICAL FRAMEWORK**

#### ***3.1 Project Development***

This project was developed within the framework of a WWW centric application capable of delivering multiple-choice tests within an on-line environment, in a manner that is cognitive and easy to use for staff and students alike.

WEBassess was thus implemented solely as a World Wide Web based application, being largely independent of both hardware and software, except that necessary to connect a computer to the Internet/WWW. WEBassess requires only a browser capable of viewing forms in order to deliver an on-line test. The forms feature is standard in most popular browsers since mid 1996. The HTML used to implement WEBassess was no later than version 3, which ensures an extremely high level of compatibility with any browser the user may possess.

Aside from the HTML code, WEBassess utilises the Lasso database linking application to create the form-based interactivity feature of the system. Lasso is an application designed to run on Macintosh computers in conjunction with a web server. Lasso receives requests for information from the user, these requests then being transferred to Filemaker Pro, where the appropriate actions are taken. Searching, addition, deletion and edition of records in the database are the main actions involved in the data entry/presentation process. This, of course, works transparently in the background so users are unaware of any processing other than the data they enter and the options they select.

The final WEBassess interface has been designed to be extremely easy to use and simple to understand, leaving little or no room for ambiguity, uncertainty or indecision. Not only has the interface been designed for an uncluttered and logical layout, the application leads users through a step-by-step procedure to achieve their final goals.

A feature which has found its way into a large amount of new software is the 'Wizard', a set of interactive questions and answers presented to the user when customising or setting up an application or part of an application, each question being asked in a generally logical order. WEBassess works in a similar fashion, with the user being led through a series of different screens and options until they have reached the area of the WEBassess application required.

The WEBassess user should require no greater computer skills or literacy than those needed to navigate on-screen with a mouse and should have the ability to type input, such as student or staff identification names and new questions and answers in the test creation module.

## **4. METHOD**

### ***4.1 Design***

The design of WEBassess is centred on multiple-choice questions presented to the user in the form of clickable options. Questions will be presented in a single form, which scrolls upwards as the user progresses through a test.

When using a multiple-choice system such as WEBassess, the student sees the written question followed by five possible answers, each answer having a letter assigned to it, ranging from A to E. A drop-down option box is located under the list of options, the student using this box to select the desired solution. The student then moves on to the next question, using the same answer selection method until the entire test is completed.

The final option available to the student after all questions have been attempted is the Submit button, which opens a dialogue box asking the student if they are sure they wish to submit their completed test. If the Yes button is selected, the test will be submitted and the resulting mark displayed. If, however, a student selects No in the dialogue box, he or she will be able to make any necessary changes to individual answers before submitting the test.

Originally a timer was to be incorporated into the WEBassess application, set to an allowed time for a test, including reading time. If the student had not completed the test by the expiration of the allotted time, the test would be automatically submitted and a result given against the answers the student managed to give. Such a feature

was incorporated successfully into the ComputerPREP WEBassessor program [<http://www.wessor.com>, 1997], allowing a test administrator to set a time limit for a given test. However, this feature was deemed unnecessary for the purpose of WEBassess due to its design as a knowledge feedback tool rather than a straight assessment system, and thus was subsequently dropped from the early planning and development stages of the project.

#### ***4.2 Development***

Once the Thesis Proposal had been presented and a detailed requirement for system operation was defined, development began on a Macintosh computer acting as a client to a Unix server. Using a TELNET application, the Macintosh computer acted as a terminal to the Unix computer, which was configured as a 'web' server, loaded with the Perl scripting language. Using a server based text editing program, code could be written directly to the Unix system, then tested through a 'web browser' program running on the Macintosh computer. Graphics and complicated HTML files could be created on the Macintosh using authoring tools such as Adobe PageMill before being transferred via an FTP program to the Unix server in order to be integrated with the existing Perl scripts.

The main problem encountered with this operating method came in the form of program errors, as it was difficult to ascertain whether the problem was caused by a system fault or a programming fault. Logical step by step analysis, combined with assistance from the project supervisor, eventually solved most coding difficulties.



After several weeks of working with the Macintosh system, problems began to occur with system resources.

To address this problem, the program development was moved to an IBM compatible Pentium computer running Windows NT. The actual development procedure was the same as that for the Macintosh, with the PC acting as a client to the Unix server.

Owing to the wide availability of software tools for the PC platform, several tools were located which substantially aided the progress of the project. These tools included a versatile TELNET client program and a Windows based Unix file editor. Up to this point many errors were occurring in even the simplest of code due to hidden characters being created in files developed on non-Unix machines.

This design procedure was maintained for several weeks, during which serious problems began to arise with both the Unix server and the level of programming required to achieve WEBassess design goals. Problems with the server included constant 'freeze ups', where the system simply did not respond, and a series of security violation errors which could not be overcome. At this point the development of the project stalled almost completely, as a majority of project time was being spent in attempts to solve inexplicable hardware and software problems.

### ***4.3 Actual Development***

At this point, a new software tool was located, the use of which has led to the WEBassess system as it now exists. Lasso, by Blueworld Software [<http://www.>

blueworld.com, 1996], is a software product which creates links between HTML pages and Filemaker Pro databases, allowing for the dynamic retrieval and input of data into a custom database via the WWW. This is achieved by inserting special Lasso 'tags' into HTML pages, these tags being visible to the Lasso program which runs in accordance with a WWW server.

For the purposes of this project, Lasso was run in conjunction with the Quid Pro Quo server on a Macintosh 8500/120 computer. All the files for the project are stored in a folder on the Quid Pro Quo web server.

The server receives all requests for information from the client computer, these requests being converted into actions, depending upon the type of request. The most common type of action is a POST-CGI operation, CGI standing for Common Gateway Interface. A POST-CGI operation tells the server to perform a task, then to present the results of this operation in another, pre-formatted file. For the purposes of WEBassess, the Lasso program is used to perform the actions specified by the user.

For example, an Admin user may search the WEBassess system for a student's user details. The user enters the student's number into a text field provided on the WEBassess WWW page. This field is contained within a series of HTML tags known as a Form. Attached to this Form is an Action= command, instructing the server what to do next. In the case of WEBassess, this command is expressed as Action="action.lasso", which instructs the server to run the Lasso application. Once the Lasso application has been started, the contents of the Form are presented to it, in this case,

the request to search for a student's user details using the student's number entered into the search field. The following figure shows how these commands are structured:

```
<Form Action="action.lasso"?[search]&[database]=Users&[layout]=Userinfo&[operator]=equals&Userid=[field:userid]&[format]=/cgi/WEBassess/search_results.html">
```

Figure 7: Example of Lasso Custom HTML Tags.

Reading from top left to bottom right, this POST-CGI command instructs the server that the Lasso program is to be run, and that Lasso is to conduct a search of the Filemaker Pro database called Users in the Userinfo layout form.

Reading on, Lasso is instructed to find a record where the Userid is equal to the number provided in the text field contained within the HTML Form. The results of this search are to be displayed in another HTML file called search\_results.html. The following figure shows the manner in which information flows in this system:

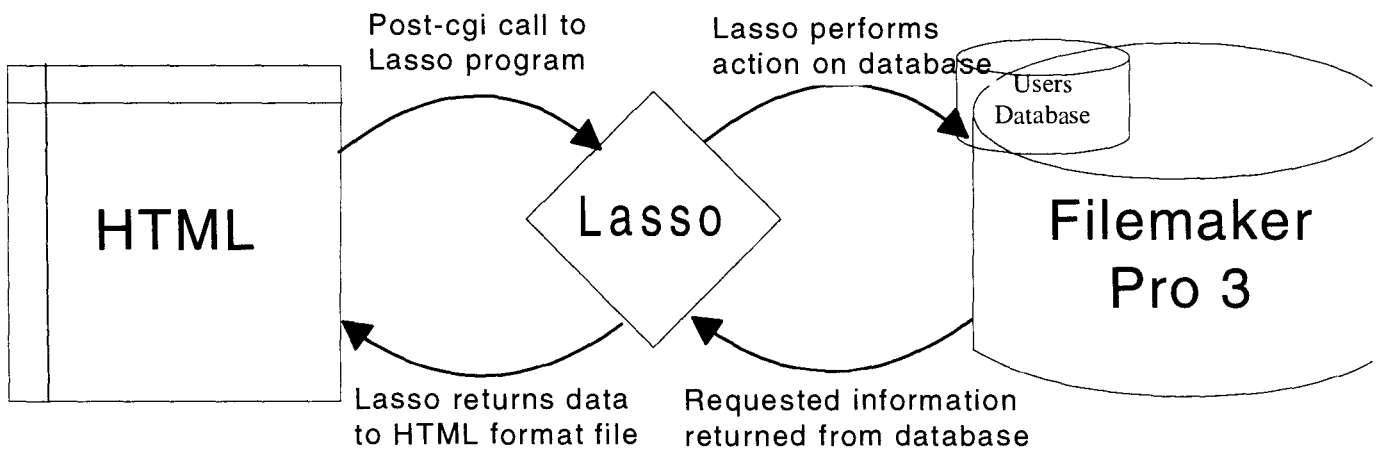


Figure 8: Information Data Flow For WEBassess System.

Periphery applications used to create other necessary files include Adobe Photoshop 3.05, Adobe PageMill 2.0, BBedit 4.0 and of course, Filemaker Pro 3. These programs were run on the same machine which operated the Quid Pro Quo web server and Lasso, namely a PowerMacintosh 8500/120 series. Due to this machine's relatively fast processor and adequate amount of RAM, no problems with project development were encountered, even with several applications running at once.

#### ***4.4 Procedure***

The procedure I adopted during the course of the project was to develop a systems analysis of how the WEBassess system should work, using this as the basis for the implementation and testing. The WEBassess was divided into distinct modules, such as the module for students and the module for lecturers. The lecturer's module was broken down into smaller modules, such as those necessary for the creation and editing of pools of questions, and new users.

Originally, rather than focus my efforts on developing and completing each module in a linear fashion, I attempted to interlace my efforts so that the different modules grew at approximately the same rate. As the modules are inter-dependent, completing them in a linear manner could have led to problems later in the project owing to unexpected occurrences, which could have meant having to make considerable alterations to finished modules.

However, this method for project development turned out to be impractical, mainly for reasons associated with testing and data flow analysis. Under the original plan,

the individual modules were to grow in parallel to each other, each one getting closer to completion at the same rate as the others. After a short amount of time, the shortcomings of this approach became apparent in the form of completed modules which did not work individually or when integrated with other modules. Due to the complex way in which information flows throughout the WEBassess system, errors occurred because this non-linear development scheme made it difficult see microscopic details of each module when working on a macroscopic scale.

At each stage of the project development I have fully documented all factors of the WEBassess implementation, particularly the problems or situations I encountered which I did not originally anticipate, the nature of these problems and how they were overcome. This documentation comprises the bulk of this thesis, as does the final description of the completed project and the level of success achieved by WEBassess. Future development possibilities for the application are also discussed.

#### ***4.5 The WEBassess System***

WEBassess has been developed to a level where all components of the system are complete and operational. The program has been in preliminary testing for several weeks, with user feedback being used to refine and enhance the look, feel and useability of the program.

The core WEBassess system consists of HTML files, Lasso data, Filemaker Pro databases and graphics for navigation and presentation within the system. These components interact with each other and with outside client requests via the Quid Pro

Quo web server running on a Macintosh computer, using Lasso as the glue between the HTML files and the WEBassess databases.

Currently, eighteen HTML files provide the functionality for WEBassess, interacting with four Filemaker Pro databases through the Lasso program. The functionality within WEBassess is broken up into three distinct categories:

1. staff;
2. student; and
3. administration.

Access to these individual functions is controlled by password authentication which is handled via a Filemaker Pro database called *wa\_users*. Users are presented with different options and may see only certain files, depending upon their level of access. Students, for example, only ever see four of the eighteen HTML files which make up WEBassess, one for logging in, one for test selection, one for answering questions, and another for viewing test results. See figure 9 in Appendix A Data Flows.

#### ***4.5.1 Data Flows***

As the user moves from one file to another, actions take place between the server, the Lasso plugin and the Filemaker Pro databases, which are open and residing in memory. Which actions are performed depend upon the option selected in each file within WEBassess. Selecting the Maintain User option button in the Admin menu invokes a [search] action in the Lasso plugin, which then instructs Filemaker to conduct the specific request in the *wa\_users* database. Lasso takes the result of this search and places the information in its own custom tags contained in a search result

file. If the specified user record is not contained in the *wa\_users* database, Lasso will instruct the server to return an error file, which indicates to the user that an error has occurred in performing the search. Figure 10 in Appendix A Data Flows gives a top-level representation of how the WEBassess system operates as a whole, particularly with regard to the Filemaker Pro databases.

#### ***4.5.2 Naming Conventions***

A problem that often occurs in the design of any kind of computer application is the tendency for numbers of required files to grow very large. This is particularly true of HTML based applications, as the need for linked pages, format files and multimedia elements, such as images and sound, tends to introduce a large amount of ‘sprawl’ to even a simple web site.

At this point a new procedure was adopted for describing files, namely placing **a\_** and **s\_** indicators in front of each Administration or Student file respectively. Though most of the file names used for the WEBassess system are still relatively cryptic, especially to those unfamiliar with the system, they are a great deal more manageable than before. This new naming system also lead to a far smaller number of files due to the elimination of duplicate files, which were located during an audit of all files within the WEBassess system.

### 4.5.3 Functionality

The WEBassess system is now fully functional and performs all the tasks called for in the original design specification. The following are the key features which the WEBassess system was designed to achieve:

- Present multiple-choice tests through a web browser, allowing each answer to be selected from a drop down box, the available options ranging from A to E. The final option the student sees is a submit button, which sends the questions and student's answers to the database for immediate marking.
- Record student test results for future retrieval and analysis, including the unit code and title of the test, the date and time the test was submitted, the total number of correct answers given and the overall percentage score. This information can be accessed by staff users, searching under a student number.

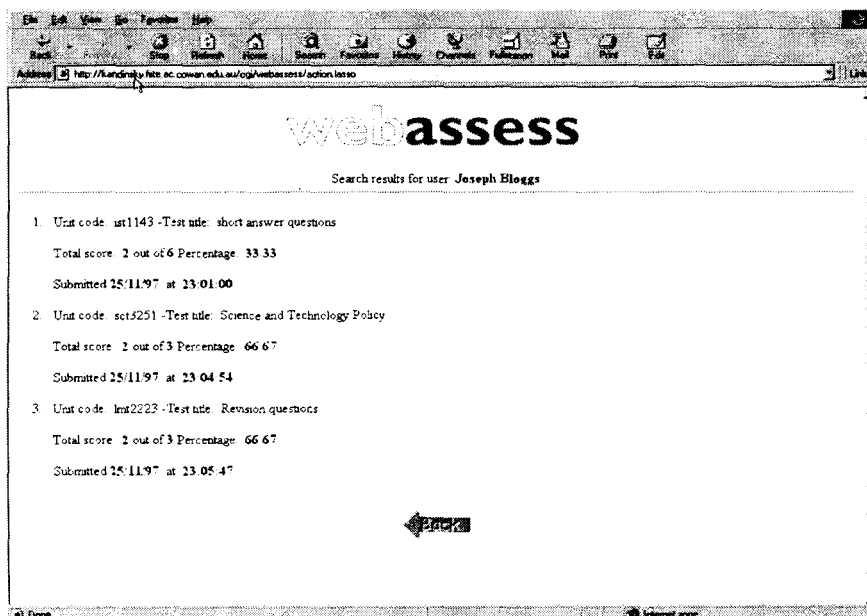


Figure 11: Student Results Statistics Screen.

- Present students with visual and written feedback to correct and incorrect answers. A green tick is presented for a correct answer, and a red cross for an



# Edith Cowan University

## Copyright Warning

You may print or download ONE copy of this document for the purpose of your own research or study.

The University does not authorize you to copy, communicate or otherwise make available electronically to any other person any copyright material contained on this site.

You are reminded of the following:

- Copyright owners are entitled to take legal action against persons who infringe their copyright.
- A reproduction of material that is protected by copyright may be a copyright infringement. Where the reproduction of such material is done without attribution of authorship, with false attribution of authorship or the authorship is treated in a derogatory manner, this may be a breach of the author's moral rights contained in Part IX of the Copyright Act 1968 (Cth).
- Courts have the power to impose a wide range of civil and criminal sanctions for infringement of copyright, infringement of moral rights and other offences under the Copyright Act 1968 (Cth). Higher penalties may apply, and higher damages may be awarded, for offences and infringements involving the conversion of material into digital or electronic form.

## USE OF THESIS

The Use of Thesis statement is not included in this version of the thesis.

incorrect answer. Explanations for both correct and incorrect answers can be attached to each question, to be entered where lecturers feel such feedback is relevant. (see Figure 2.)

- Add, edit or delete questions and question pools from the system. All aspects of question pool and question management can be conducted through the WWW interface, eliminating the need to alter individual databases manually.
- Add, edit or delete users from the system, with levels of security access being specified for each user by the system administrator. The system automatically presents users with the options that their level of access allows them to see.

These options include:

◆ Admin

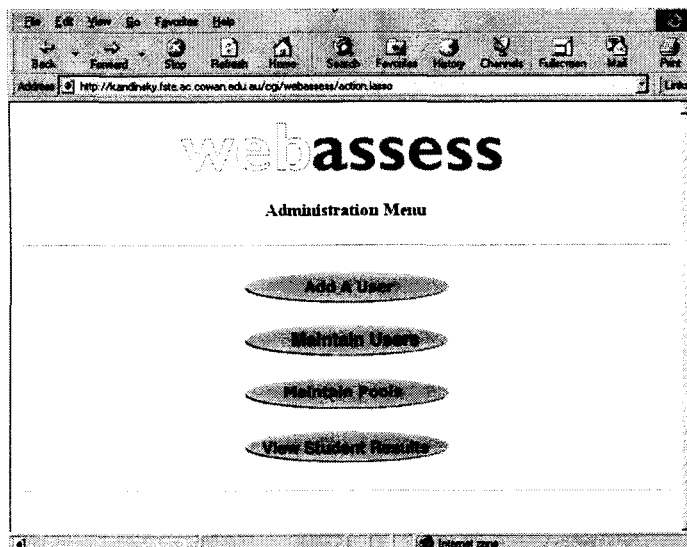


Figure 12: WEBAssess Admin User Menu Screen.

◆ Staff

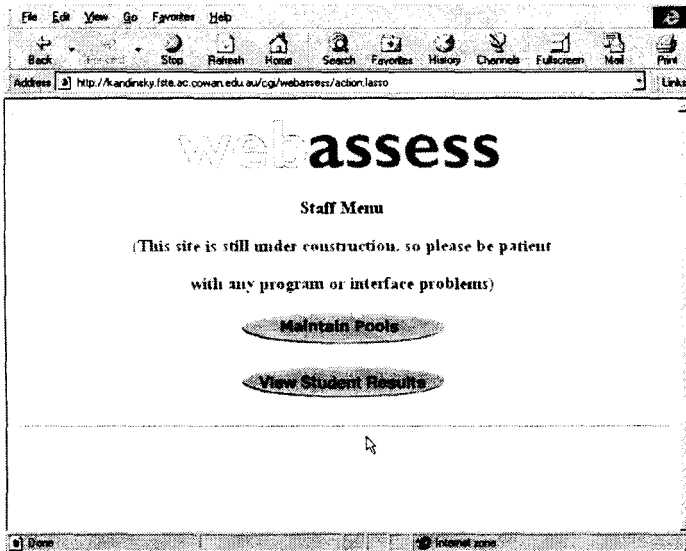


Figure 13: WEBassess Staff User Menu Screen.

◆ Student

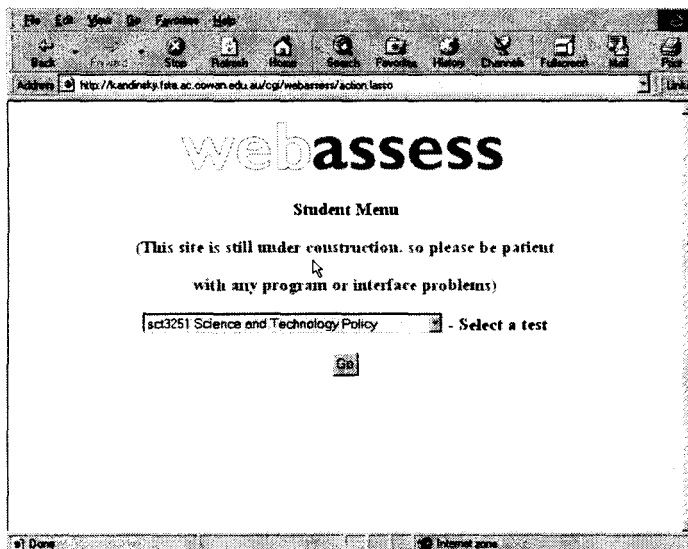


Figure 14: WEBassess Student User Menu Screen.

One of the original aims of WEBassess was to enable the system to be fully managed and administered using the WWW as a delivery medium, with no requirement for manual maintenance of HTML files or the system databases. During the early development stages of WEBassess, when Perl scripting was being used as the

programming tool, it seemed unlikely that this design goal would be met, and that a certain amount of manual administration would be required. However, use of the current development tools has eliminated any problems which would have required the manual alteration of files used within the WEBassess system.

As new versions of the Lasso software are released, a time may come when some of these files will require alteration, or new files will need to be created in order to enhance the capabilities of WEBassess. However, as far as the everyday use of WEBassess is concerned, users need know nothing about HTML, Lasso, Filemaker Pro or web servers. The functionality of the system has been made as efficient as possible, while making the actual Lasso, HTML and database interactions completely transparent.

#### ***4.6 Testing***

Testing for the project was conducted in two areas. Firstly, testing was carried out on a continuous level during all stages of the project development. As a new section of the program was implemented, it was run through a series of operational dry runs. These dry run tests, also known as black box testing, would indicate any problems with the new section of coding. Once a problem was identified, it would be put through a series of trouble shooting steps, each recorded and noted for its effects, be they positive, negative or inconclusive. This method of testing led to an incremental solving of problems, both small and large.

One difficulty encountered with this kind of testing methodology arose when using the newly debugged code in conjunction with another component of the system. The code that had already been corrected would in turn react to poorly structured data coming or going from the new component.

This created a situation where the newest component of the code had to be fully problem solved, including the use of customised test data. Once a piece of code had been validated, along with the component it communicated with, the necessity arose to look at the data flowing between the components. Problems could occur because of incompatible data types, such as text strings being written into a numeric-only field, or vice versa. In many cases, information would become lost due to spaces in text causing the URL encoding of information to be incomplete. For example, the URL string IST 1132 would drop the 1132 after the space following IST, unless a special command was inserted in the command line as well as the component receiving the data. Problems like these were numerous, and time consuming to locate and correct.

A second type of testing took the form of actual user interaction with the WEBassess program, these users being asked to test and evaluate various components of the system.

The WEBassess system has three levels of user access, these being student, staff and administration. The student interface allows students to log into the system via their student number, after which they are required to select which test they wish to take. Once students have completed all the questions and submitted the tests, they are

presented with the test results according to score and percentages. There are no further screens to which students have access.

The staff interface has a several available options, including the creation and deletion of question pools, as well as the creation, deletion and editing of existing questions within a question pool. Staff users can also search for and retrieve student test result according to student number.

The final level of access is reserved for administration users, relating to the role of managing the entire WEBassess system. The administration user can access all the options available to the staff user, with the extra options of adding, deleting and editing user information within the WEBassess system.

Of these three levels of user access to WEBassess, student and staff components have been selected for testing by relevant users. Administration level testing was considered to be less relevant, as the interface and code components are very similar to those found within the staff system, and more importantly, only a very few authorised users will have access to this facility within WEBassess.

#### ***4.6.1 Student Testing Procedure***

The WEBassess program only reached a stage suitable for testing in the last week of student lectures, which permitted only a limited amount of testing. A core unit for the Bachelor of Science in Communications and Information Technology was chosen as the test bed unit for a WEBassess trial. IST 1132 Introduction to Information

Technology is a first year unit that examines the issues and technologies involved with the use and implementation of information technology in society. As there are four tutorial/workshops in this unit each week, it was decided that each of these tutorial/workshops would provide a suitable testing environment.

A small test of only ten questions was created on the system, these ten questions representing questions that students may encounter in the examination for this unit. Once students had completed the test and had seen their score, a link to an on-line evaluation form was provided. Upon clicking this link, the students were presented with a disclaimer consent form which stated that the evaluation form was to be completed on a completely voluntary basis, and guaranteeing that no identification of individual students would take place.

A statement was also made to the effect that the test the student had just completed would not reflect in any way on the student's course average or final unit mark.

Students were presented with a total of seven questions on the useability and interface design of the program. An area for comments was also provided to allow students to express their thoughts more concisely regarding the WebASSESS system. The results from these submitted forms were automatically entered into a database, from which the figures on the following page were derived.



Table 1: Student Evaluation Form Response Data Table.

No	Question	YES	NO	More	Less	Did not answer
1	Did you find the interface logical and easy to understand?	22	4			
2	Did you find the method for answering questions logical?	23	3			
3	Would you find answering 50-60 multiple choice questions easier than a standard multi-format exam?	19	7			
4	Did the program function for you correctly?	21	5			
5	Did you like the method of feedback?	22	4			
6	Would you like more or less detail in the feedback form?			19	4	3
7	In a timed exam situation, would you prefer to sit a test like this one rather than a normal, multi-format timed exam?	18	7			1

The first time this evaluation form was used, it was presented to the students in a printed form. However, after collecting these first responses and seeing scratched out writing, it seemed inefficient not to use the medium in which the WEBAssess program was being developed. The printed form was then re-created as a HTML page, acting as a front end to a Filemaker Pro database. Figure 15 in Appendix C Forms shows the format of the on-line student evaluation form.

The figures returned from this evaluation form indicate that student attitudes to the system were generally positive, especially in respect to the WEBAssess testing interface. The WEBAssess interface was designed to be as simple and as logical as possible, and, from the data returned from student users, it appears that this particular aspect of the system has been implemented successfully.

The questions that brought about the lowest student support were those regarding the use of WEBAssess under full scale testing conditions. Question 3, *Would you find answering 50-60 multiple choice questions easier than a standard multi-format exam?*

returned 19 yes responses out of a total of 26 respondents. Question 7, *In a timed exam situation, would you prefer to sit a test like this one rather than a normal, multi-format exam?* returned a total of 18 yes responses out of a total of 25 respondents. These were the only two questions which returned positive results numbering less than 20, which is a favourable result considering the WEBassess system is still in the preliminary testing stage.

Results from questions 3 and 7, when viewed against the other 5 questions, indicate that students see the WEBassess testing environment in a positive manner, but would not prefer using the system for long tests of around 50-60 questions. Question 7 asked students responding No to indicate their reasons why. Of these comments, the following two seem indicative of the feeling amongst students regarding lengthy WEBassess tests:

*"I think something that has to be considered is the fact that sitting in front of a computer for the 50 - 60 questions could cause your eyes problems..."*

*"Because looking in the monitor for that long will be very tiring because of the reflection. In a timed exam there will be very stressed and maybe lost some question."*

The problem illustrated here is not just a problem with WEBassess, but with the whole concept of interacting with a computer over a prolonged period of time. This is particularly true of computers which have display units running at low refresh rates (the number of times the screen re-draws each second), which can lead to flickering, making the machine very difficult to use. The second student comment which talks about getting lost on screen when scrolling through a large number of questions is definitely a drawback of the WEBassess test presentation paradigm. However, when

weighed against the problems faced with the one question per screen system, such as the length of time required for the test, the current WEBassess design is a compromise between ease of use, speed and efficiency.

#### 4.6.2 Staff Testing Procedure

Staff members were also asked to use and evaluate the WEBassess's efficiency, ease of use and relevancy. A link to an on-line evaluation form was provided on the Maintain Pools screen of WEBassess, so that staff users could fill in the evaluation form once they had used the various components of the system.

The staff evaluation form contained some of the questions presented to the student users, particularly those dealing with interface issues and usefulness of the system. A total of ten questions plus a comments field were provided within the staff evaluation form. The following table shows the questions and responses gathered from this evaluation form:

Table 2: WEBassess Staff Evaluation Data Table.

No	Question	YES	NO	More	Less	Did not answer
1	Did you find the interface logical and easy to understand?	2	1			2
2	Did you find the method for inputting questions logical?	2	2			1
3	Would you find entering 50-60 multiple choice-choice questions easier than creating a standard multi-format (ie multiple choice - short answer - essay)test?	2	1			2
4	Did the program function properly for you?	3	1			1
5	Did you feel that the level of feedback to students is appropriate?	2	1			2
6	Would you like more or less detail in the students results feedback form?				1	4
7	Given the design and capability of the program, do you feel that it could be a useful tool for the creation of tests based on unit coursework?	4				1
8	In its current form, do you feel the WebASSESS interface is efficient enough to create large collections of questions?	1	3			1
9	Do you believe students would prefer such an on-line test in preference to a standard written test?	1	2			2
10	Are there any features missing from WebASSESS which you feel would enhance its capabilities or make it easier to use?	1	2			2

As Table 2 illustrates, the number of staff responses was far below that of the student responses. A total of 19 E-mail messages were sent out to members of Edith Cowan University staff requesting that they use and evaluate the system when convenient. Of these, only five evaluation forms were submitted at the time of writing.

Figure 16 in Appendix C Forms shows the staff on-line evaluation form, including the consent agreement at the top of the form, stating that completion of the evaluation document is purely voluntary, and that no user details or information are entered into the database.

From the responses listed in Table 2, it is clear that staff users of WEBassess feel it is an effective tool for the creation of on-line tests related to coursework (question 7), but that for tests requiring large numbers of questions (question 8) it would not be as useful. Due to the quantitative rather than qualitative nature of these evaluation questions, exact reasons for these results are difficult to define. One reason may lie in the method of question creation, as illustrated by the following staff user comment:

*“The system is a bit clunky for inputting a list of questions - you’ve got to go through a number of screens for each question.”*

This is a valid point and was raised by another staff user who felt that the:

*“User interface not logical plus inconsistent...”*

It would appear from the user response figures and these comments that the system for entering questions is far more tedious than that for reading and answering questions.

Another issue raised by at least two members of staff on an informal level regards assigning levels of difficulty to each question, so that tests are structured in a way that has easy questions at the beginning of the test leading into more challenging questions towards the end. The suggested difficulty range was one, two and three, one being the easiest and three being the most difficult. Tests would be arranged so that a proportionate number of questions from each difficulty level were used, such as twenty level one questions, ten level two questions and approximately five level three questions.

Overall, the staff evaluation figures seem to indicate that these users see WEBassess as potentially very useful, but currently lacking in areas of efficiency and logical layout.

#### ***4.6.3 Results of Testing***

Though this evaluation is based on a relatively small sample of the total staff population, these responses, in addition to remarks received on an informal basis from other staff members, indicate that the key source of difficulty with the WEBassess system was the number of screens used for each action.

As a result, a change has been implemented in the system, significantly reducing the number of steps required to perform a task within WEBassess. This change takes the form of the [inline] command, which is used by the Lasso application to perform more than one operation on each HTML page. For example, a user may wish to delete a question from a question pool. Previously, the user would be presented with a screen

confirming that the question had indeed been deleted, plus two option buttons, one for the main Maintain Pools screen, and another for adding another question. Each component of WEBassess which allowed for the deletion or addition of records had a similar message and an option page appeared after the delete or add action. The reasoning behind this navigation methodology was to lead the user through each system action in a logical manner.

This system, thought logical at the time of development, caused difficulties for nearly every user of the system. Consequently, the method for deleting and adding records was changed, so that the screen re-loads itself automatically after each action. For example, if a user now deletes a record from a question pool, the list of questions in the pool is re-loaded immediately, though this time showing one less question.

Before this change, each confirmation page presented a Back button which contained Lasso command tags, such as [search], followed by the name of the file the user has left. The commands in these tags would re-load the original page, reflecting the changes made by the action, whether it be deletion or addition.

As the [inline] command allows multiple actions to be performed within a single HTML page, the [search] command can be placed within the [inline] tag at the beginning of the HTML file, allowing the necessary search to be conducted every time the page is loaded.

Although this is only a small change in the overall structure of the WEBassess system, it now seems an obvious part of the program, and a significant omission during the

initial development process. As stated previously, the user testing conducted for the purposes of the WEBassess project was meant only to be a guide to the level of success achieved by the project. Also, due to the time constraints placed on the user testing and evaluation, all questions were quantitative in nature, except for the comments section where users had the opportunity to enter some qualitative data.

However, results from even this limited user evaluation have been exceptionally helpful in the current and continuing development of the WEBassess system. Though wholesale changes to the system are not currently deemed necessary, some user suggestions and criticisms will definitely find their way into future additions and refinements of the system.

#### ***4.7 Limitations***

The greatest limitation of the development of the WEBassess project was the time frame in which it was conducted. The change in development system from Perl to Lasso and Filemaker Pro caused a significant investment in time and energy to be lost, though this situation was outside the control of the researcher at the time.

Within the literature review and introduction section of this document several references were made to testing, both as a system of assessment and its application to computers. The bulk of this information deals with computers as a tool for the dissemination and automated scoring of multiple-choice questions, though some mention was made of the concept of testing as field of its own.

It is not the purpose of this document to create a new definition for how tests are conducted, or to describe the effect computer based testing will have on students. A study on this topic was actually conducted by Bugbee (1990). The aim of the WEBAssess project was to create a program specifically for the purpose of testing students on coursework material via the Internet/WWW. This written document is an account of the process and criteria by which the WEBAssess system was developed to its current state of application.

The WEBAssess system is not intended to be a replacement for traditional paper based exams for purposes of assessment. As the WEBAssess system currently exists, including the technology base upon which it is built, security is a good measure short of that required to offer official assessments by this means. To allow assessment of this kind to take place, security problems such as realms-based user access and student supervision would have to be solved. At the time of a full implementation of the WEBAssess as an assessment tool, these problems will need to be addressed and solved. However, at present, the password based security system is adequate for the purposes of continued project development.

The final limitation that needs to be stated for the purposes of this document concerns user testing and evaluation as a measure of the project's success. As shown in the previous section, both staff and student users of the WEBAssess system were asked to give feedback on various areas of the system, this feedback taking the form of a simple Yes or No questions. Normally, survey forms are used as a method of theory testing, with data being gathered to prove or disprove a supposition.



This was not the intention of the survey forms used in this project. The questions asked of users of the WEBassess system were designed purely to facilitate user feedback and evaluation concerning certain aspects of the systems design. These aspects included:

1. Interface issues;
2. ease of use;
3. logical layout;
4. problems encountered; and
5. level of feedback in scoring and answers to questions.

The data taken from this level of user feedback cannot be used in evaluating the system in regards to its merits as a new paradigm in testing, or as a comparative analysis of computer versus paper-based tests. The figures taken from the feedback serve one purpose only, to raise any issues and problems that may exist within the system under real world test conditions. Real world conditions refer to the conditions encountered by people using the system under everyday conditions, rather than those conditions created by the developer, who can consciously use and test the system in a way which is least likely to cause a 'crash'.

## **5. FUTURE DEVELOPMENTS**

A number of possible extensions may be made to the initial multiple-choice test format, including the provision of Yes/No, True/False and short answer questions. The ability for WEBassess staff users to create and insert graphics, diagrams or multimedia elements into a test is also considered extremely desirable, especially in technical units, such as electrical engineering and systems analysis.

Of these developments, Yes/No and True/False question types are the most likely to be introduced in the very near future. These question types were omitted from the WEBassess project during development for purposes of difficulty and time economy. In order to get the system to a workable stage within the allotted time scale, it was decided to use a single question type, so problems with integrating different question types into one form could be avoided.

Now that the WEBassess system has been developed to a viable stage where the underlying application methodology has been validated, the addition and integration of multiple question types should prove relatively straightforward.

Staff users will wish to create tests from randomly generated pools of questions once existing question pools grow to very large proportions. This will be implemented in a manner that allows a user to specify the number of questions required in a test, the question pool(s) from which questions are to be drawn and the name of the new test.

As a [random] tag already exists within the Lasso application, this enhancement to the system should prove relatively straightforward to implement.

This development will be followed by the addition of multimedia objects into the system, though this will be a great deal more complicated in design and implementation. Just as the Lasso program was used to tie HTML files to Filemaker Pro databases, an application is needed to handle and control multimedia objects whilst working in accord with these other programs. All this must be implemented so as to be completely transparent to the end user.

The major benefit of the current WEBassess development environment is that it allows for rapid changes or additions to the existing system. Some of the changes already made to the system as a result of user feedback were done so in a time frame measured in minutes and hours. Generally, changes can be made to the system by creating new fields in existing databases, or creating new databases, then creating links to these databases using Lasso and HTML files. As a large number of the required Lasso and HTML tags already exist in current WEBassess files, this code can be re-used with the minimum of alteration and re-engineering. With the change of some tags, page text and graphics, a completely new component of WEBassess can be created, working transparently with the rest of the system.

This ability to rapidly develop and enhance new modules for the WEBassess system means the functionality of the application can be constantly updated as required. In traditional programming and development environments, such as the aforementioned Perl scripting, such changes would be drastic and prohibitively time consuming.

## **6. SUMMARY**

### ***6.1 Design Goals and System Development***

Owing to a need to be able to deliver content specific, automated on-line tests to students at Edith Cowan University, the WEBassess project was undertaken. The aim of this project is to allow staff members to create, edit and delete pools of multiple choice questions, while being able to view student scores as a result of these tests. Students sit the multiple-choice test by selecting an answer from a series of possible answers, then submit this test once all questions have been answered. The result is then displayed to the student, with visual indicators for correct and incorrect answers, including explanations why an answer is deemed correct or incorrect.

The WEBassess system has been designed to allow for the administration of tests, questions, users and security, accessed entirely through a World Wide Web browser application. The major design goal for the WEBassess system was ease of use and logical design of the user interface. Though some difficulties in this area were identified as a result of a staff evaluation form, it is believed that WEBassess does now meet this design objective.

WEBassess has been designed specifically for the needs of Edith Cowan University, though many aspects of the overall system have been fashioned on the capabilities and features found in other, similar testing systems. WEBassess encompasses the most effective features of these other systems, while using some original concepts of its own.

## *6.2 Conclusion*

The literature indicates that computer based testing has been slowly developing as a recognised method of assessment for more than two decades, mainly due to the automated marking and statistical analysis features that are at the very cornerstone of this type of testing. Though multiple-choice, true/false and yes/no are currently the only viable question types, each of these is a recognised method for testing a person's knowledge of a given topic.

Since the introduction of the WWW in 1990, an entirely new medium has been created for the delivery of computer based tests. A large number of testing systems are currently available on the WWW. The graphical, point and click interface of the WWW, when combined with form compliant browser applications, allows for the development of exceptionally versatile testing systems. One of the most prevalent features of the WWW is the currency of the information contained within it, such as international news items, which can appear on the 'net' within minutes of an event happening. This same ability can be used for presenting on-line tests, where very specific tests can be created within a short time of some knowledge being acquired or a new system being introduced.

For example, if an on-line driver's licence test existed, and if several changes had been recently made to the road rules, these changes could be entered into the system, so that all tests generated from these rules would then be correct. In a manual system, entire tests would have to be re-written, printed and distributed.

The WEBassess system was designed to offer the best features of a variety of testing systems, so that staff and student users feel comfortable and confident with the program.

The system described in this document is complete in all areas stated in the original design specification, and though it is currently still in the preliminary stages of user testing, it appears that the testing methodology is sound, as is the overall interface component of the system.

It is hoped that the WEBassess system tool will offer staff and students at Edith Cowan University a relevant, interesting and enjoyable method for creating and sitting on-line tests, with a view to enhancing the learning process and experience.

## 7. REFERENCES

Advanced Consulting Inc. *Case Study: Intranet Based Testing and Assessment*. [on-line]. Available WWW: <http://www.acihome.com/case.html> [1997, 26 March].

Aronson, D. T, and Briggs, L. J. (1983). "Contributions of Gagne and Briggs to a Prescriptive Model of Instruction", *Instructional-Design Theories and Models: An Overview of Their Current Status*. Hillsdale, New Jersey: Lawrence Erlbaum Associates. p75-100.

Ausweb 97. *The Development of a Multiple-Choice and True-False Testing Environment On The Web*. [on-line]. Available WWW: <http://www.scu.edu.au/ausweb95/papers/education3/byrnes/swish.txt> [1997, April 12].

Bugbee, Alan C., Jr. & Bernt, Frank M. (1990). Testing by Computer: Findings in Six Years of Use 1982-1988, *Journal of Research on Computing in Education*, 23, 87-100.

ComputerPREP. *Your natural resource for training technology*. [on-line]. Available WWW: <http://www.webassessor.com>. [1997, 20 October].

Gagne, R. M, and Briggs, L. J. (1979). *Principles of Instructional Design*. (2nd ed.). New York: Holt, Rinehart and Winston.

Harte-Hanks. *Online Testing Welcome*. [on-line]. Available WWW: <http://www.harte-hanks.com/demo/testing/> [1997, 11 April].

Holtz, Neal. *The Tutorial Gateway*. [on-line]. Available WWW: <http://www.civeng.carleton.ca:80/~nholtz/tut/doc/doc.html> [1997, April 11].

Isaac, Stephen. (1971). *Handbook in research and evaluation : a collection of principles, methods, and strategies useful in the planning, design, and evaluation of studies in education and the behavioural sciences*. San Diego : Edits Publishers.

International Foundation of Employee Benefit Plans. *Computer Based Testing*. [on-line]. Available WWW: <http://www.ifebp.org/cecbt.html> [1997, March 29].

Kaliski, John A. *The Online Classroom*. [on-line]. Available WWW: <http://web.business.mankato.msus.edu/Online.Classroom.html>

Leedy, Paul D. (1993). *Practical research : planning and design*. New York : Maxwell Macmillan International.

Maricopa Center for Learning and Instruction. *Report '94: Technology Based Testing*. [on-line]. Available WWW: <http://www.mcli.dist.maricopa.edu/ocotillo/report94/rep11.html> [1997, April 15].

Mason, Emanuel and Bramble, William J. (1978). *Understanding and conducting research : applications in education and the behavioural sciences*. New York : McGraw-Hill.

Online Classroom test example. [on-line]. Available WWW:  
[http://krypton.mankato.msus.edu/~johnk/mgmt.200/test\\_list.html](http://krypton.mankato.msus.edu/~johnk/mgmt.200/test_list.html)

Overbaugh, R. C. (1994). Research-Based Guidelines for Computer-Based Instruction Development. *Journal of Research on Computing in Education*, Vol 27. No 1, Fall 1994, p29-47.

Ramsden, P. (1992). *Learning to teach in higher education*. London: Routledge.

Review of John A Kaliski's Online Classroom website. [on-line]. Available WWW:  
<http://instruct.unc.edu:6080/wcb/schools/5/3420/kthomas/new11961/forums/forum1/messages/13.html>,[1996]

Steinberg, E. R.. (1991). *Computer-Assisted Instruction: A Synthesis of Theory, Practice and Technology*. Hillsdale, New Jersey: Lawrence Erlbaum Associates.

Taylor, D.B. (1983). Computer-based testing in an introductory marketing course. *Programmed Learning and Educational Technology*, 20, 1, 76-80.

Test.com Inc. *Professional and Personal Testing*. [on-line]. Available WWW:  
<http://www.test.com/> [1997, May 23].

Vitale, E.& Cole, C.A. (1979). Another pedagogical innovation: a description and empirical evaluation of TIPS, a computer-aided instructional system. Paper presented at the annual conference of the American Marketing Association, Chicago.



## **8. BIBLIOGRAPHY**

- Bugbee, Alan C., Jr. (1996). The Equivalence of Paper-and-Pencil and Computer-Based Testing, *Journal of Research on Computing in Education*, 28, 282-299.
- Crock, M. & Dekkers, J. (1994). *Issues in the use of CML for distance education in Australia*. Melbourne:
- Educational Testing Service. *ETS Highlights - Computer-Based Testing*. [on-line]. Available WWW: <http://www.ets.org/aboutets/arhigh.html> [1997, 24 March].
- Gwinn, John F. & Beal, Loretta F. (1988). On-Line Computer Testing: Implementation and Endorsement, *Journal of Educational Technology Systems*, 16, 238-252.
- SkillMaster Software Inc. *SkillMaster - Overview of SkillMaster*. [on-line]. Available WWW: <http://www.trainingnet.com/smaste/welcome.htm> [1997, March 27].
- Zandvliet, David Bryan. (1993). *A comparison of the computer administered and written forms of multiple choice testing*. Canada: University Of Victoria.

## APPENDIX A - DATA FLOWS





## **APPENDIX B - USER GUIDE**

## USER GUIDE

As mentioned in the body of the thesis, WEBassess is currently configured for use by staff and student users. The following guide will present the steps necessary to use all of the features of the WEBassess system for Staff level users.

### Login in

To log into WEBassess, open a WWW browser (such as Netscape Navigator) and go to the following URL address:

*<http://kandinsky.fste.ac.cowan.edu.au>*

- This address will present the following screen:

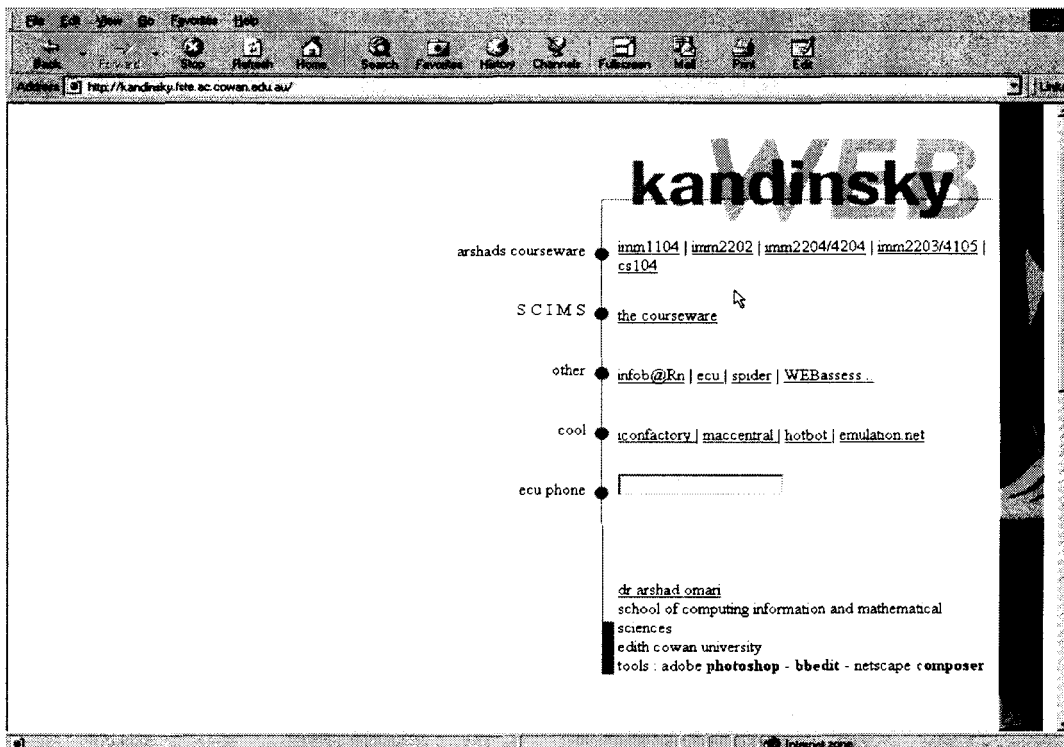


Figure 17: Kandinsky Web Homepage for WEBassess.

- From the *Other* group on this page, click *WEBassess*.

The following screen will now appear:

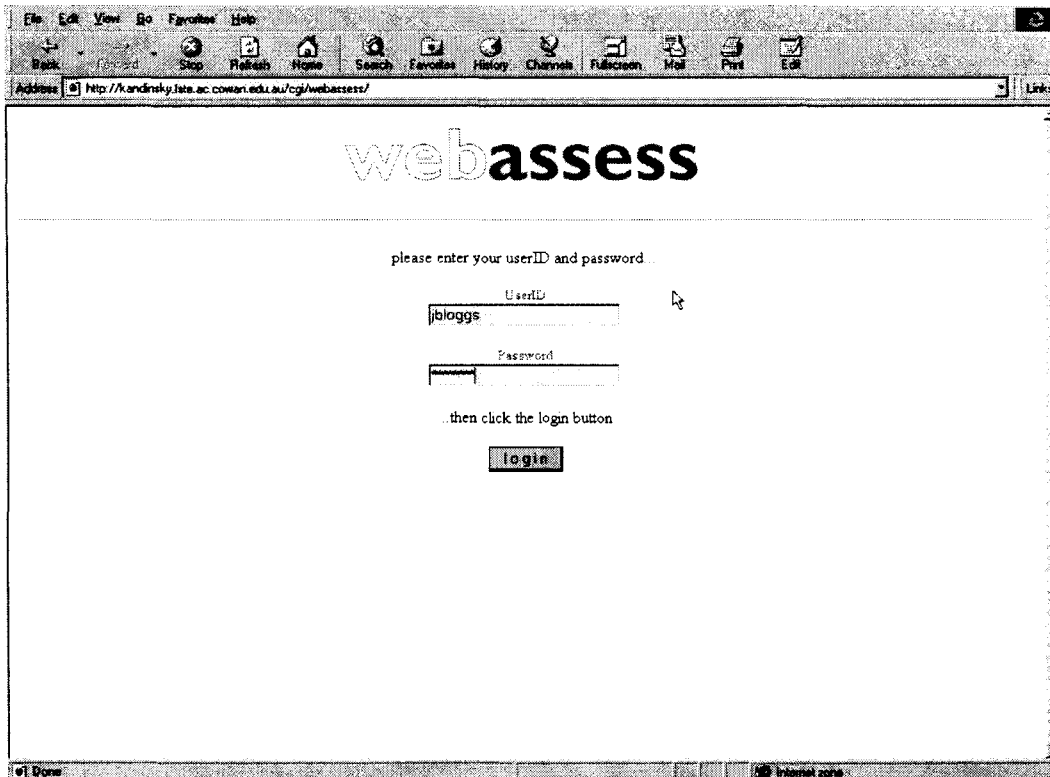


Figure 18: WEBassess User Login Screen.

- In the *UserID* field type your user identification.
- In the *Password* field type your password.
- Now click the *login* button.

The screen on the following page will now appear:

## Staff Menu

- If you are a Staff user, the Staff Menu for WEBassess should now be presented, as shown in the following figure.

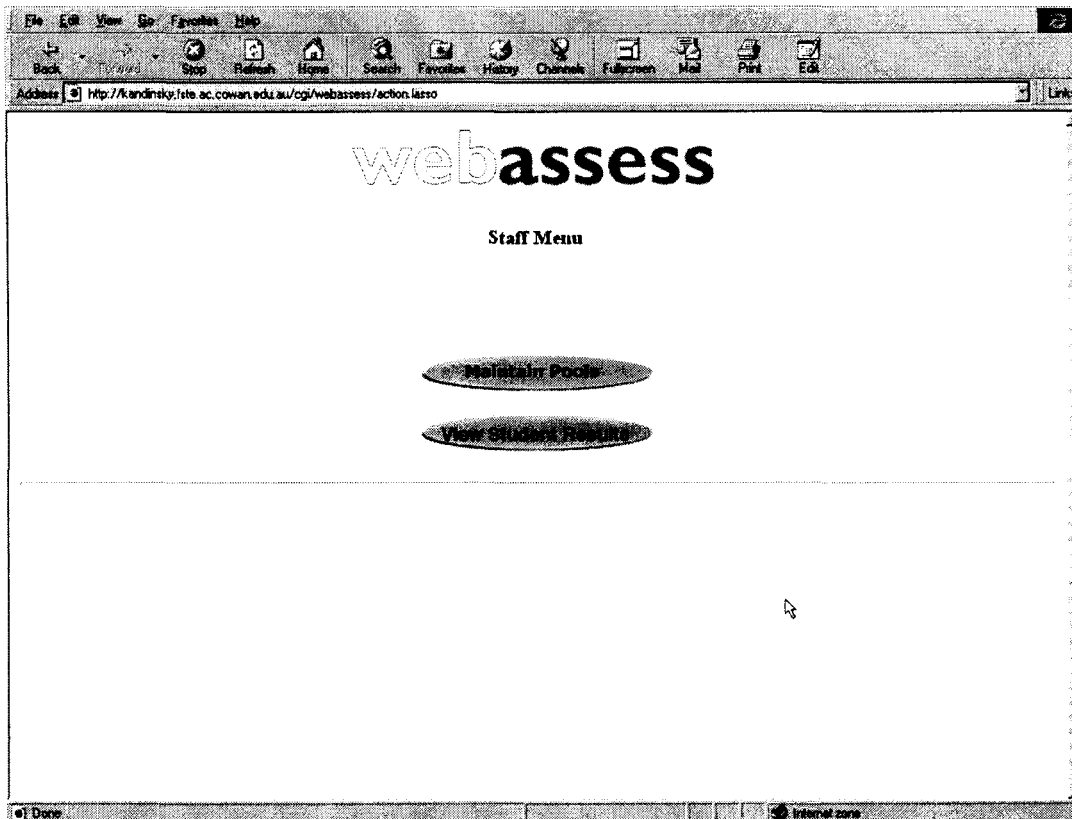


Figure 19: WEBassess Staff Menu Screen.

- Click on the **Maintain Pools** button to enter the WEBassess Pool Maintenance area.

## Pool Maintenance

- The figure on the following page shows the WEBassess Question Pool maintenance screen. From this screen a user may maintain a question pool, delete a question pool or add a new question pool
- For the purpose of this guide we will add a new question pool, then maintain the questions within it, after which we will delete the pool.



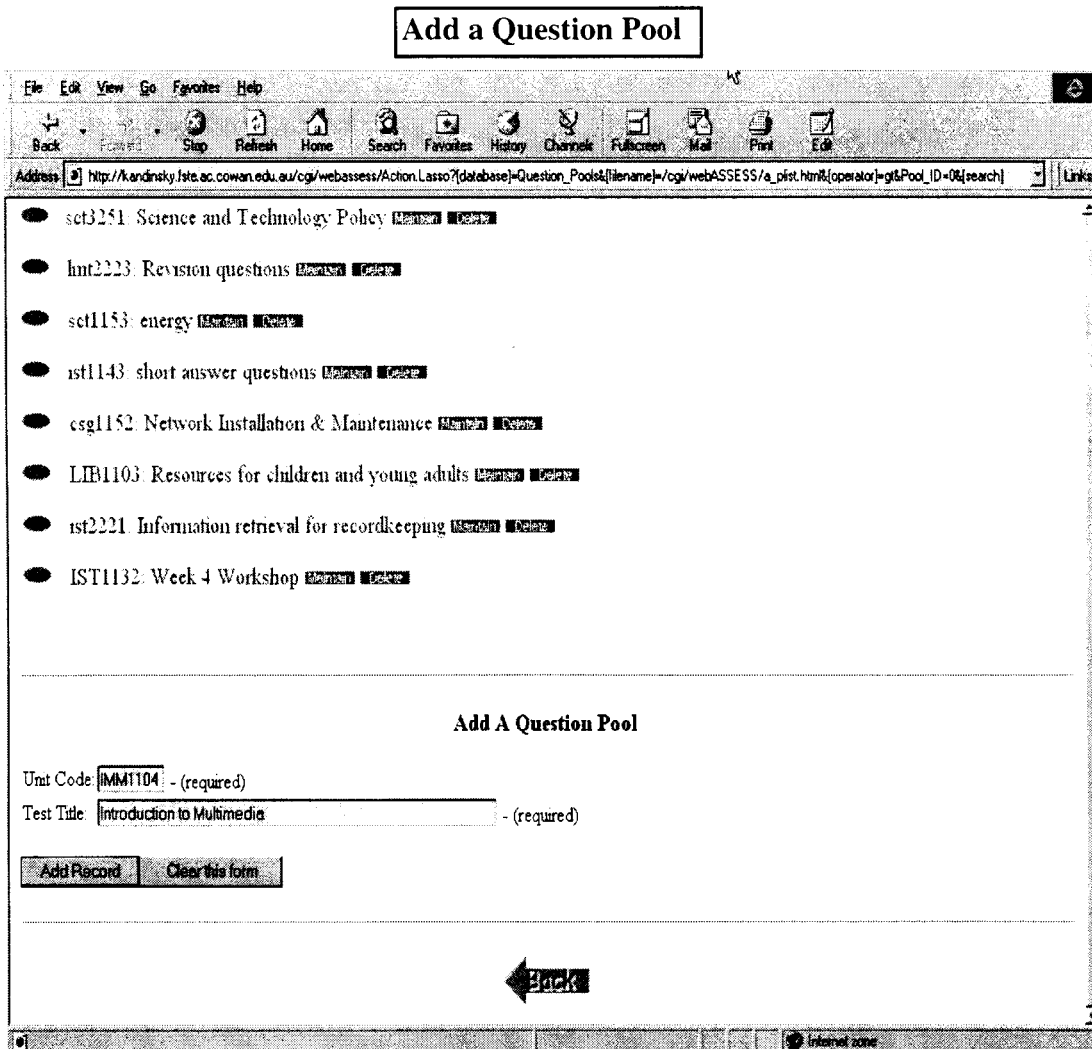


Figure 20: WEBassess Question Pool Maintenance Screen.

- To add a new Question Pool, simply type the **Unit Code** in the field provided, followed by the **Test Title** in the field directly below Unit Code. Once the information has been entered, click the **Add Record** button. See

- The Screen should now re-loads itself, displaying the new Question Pool at the bottom of the list. See figure below:

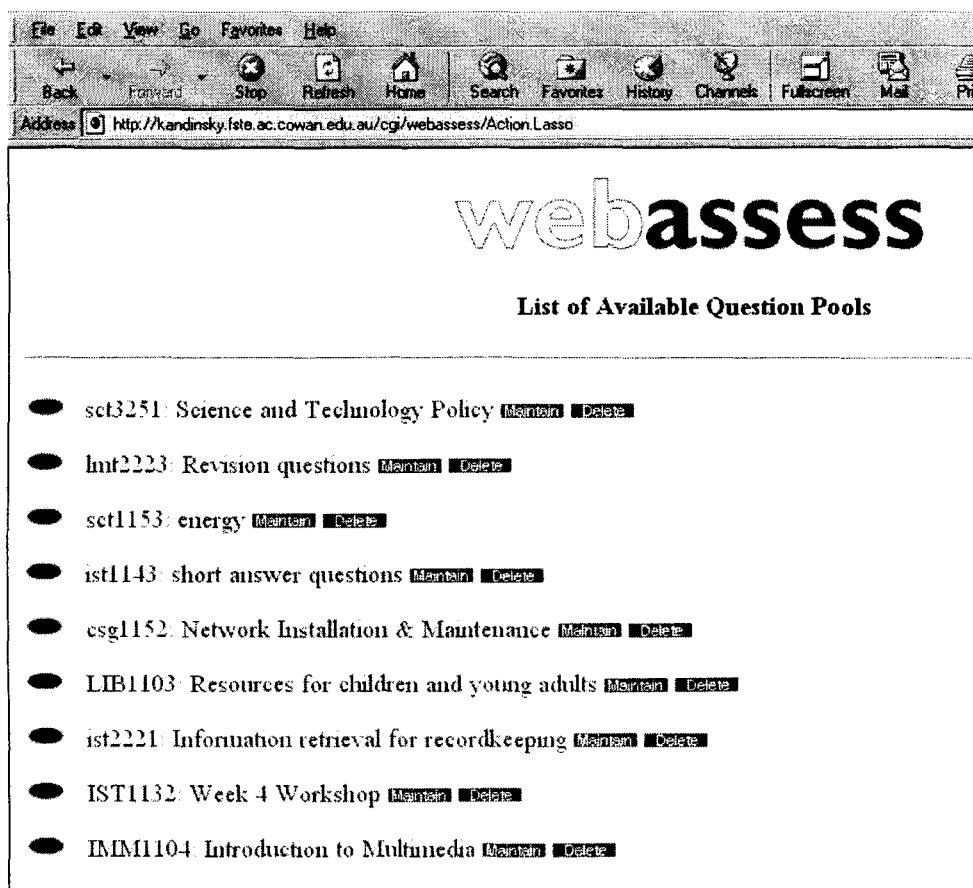


Figure 21: WEBassess Add Question Pool Result.

### Delete a Question Pool

- To delete a Question Pool, simply click the blue Delete button to the right of the question pool unit code and test title. This will cause the page to re-load, minus the question pool which has been deleted.

## Edit a Question

- Now that the new question pool has been created, questions need to be entered into the pool. To do this, click on the blue Maintain button located to the right of the question pool unit code and test title.
- The following figure should now appear:

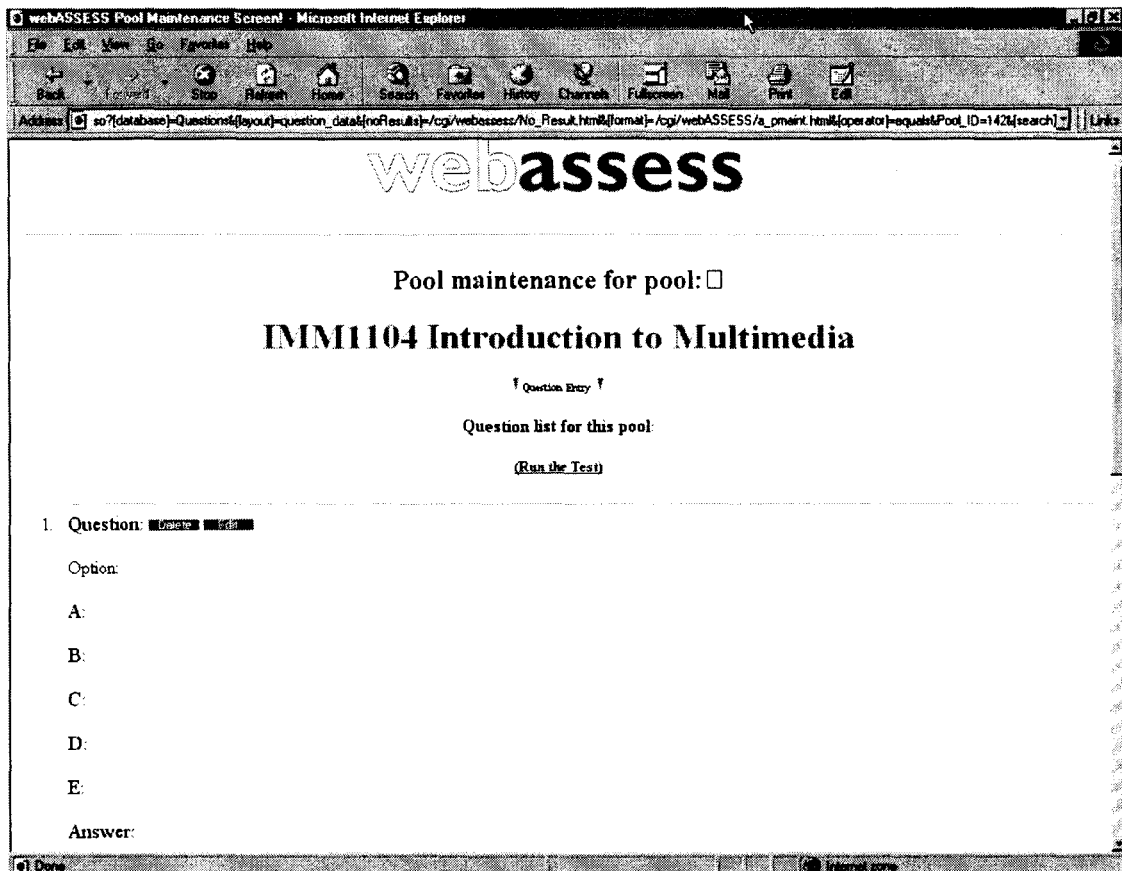


Figure 22: WEBassess Pool Maintenance Screen.

- This screen shows a single blank question in the question pool we have just created. This blank question is created automatically so that this particular pool can be accessed. We will now edit this blank question so that it actually contains some text.
- To do this, click on the blue Edit option located to the right of the blank question. The screen shown in the following figure will now appear:

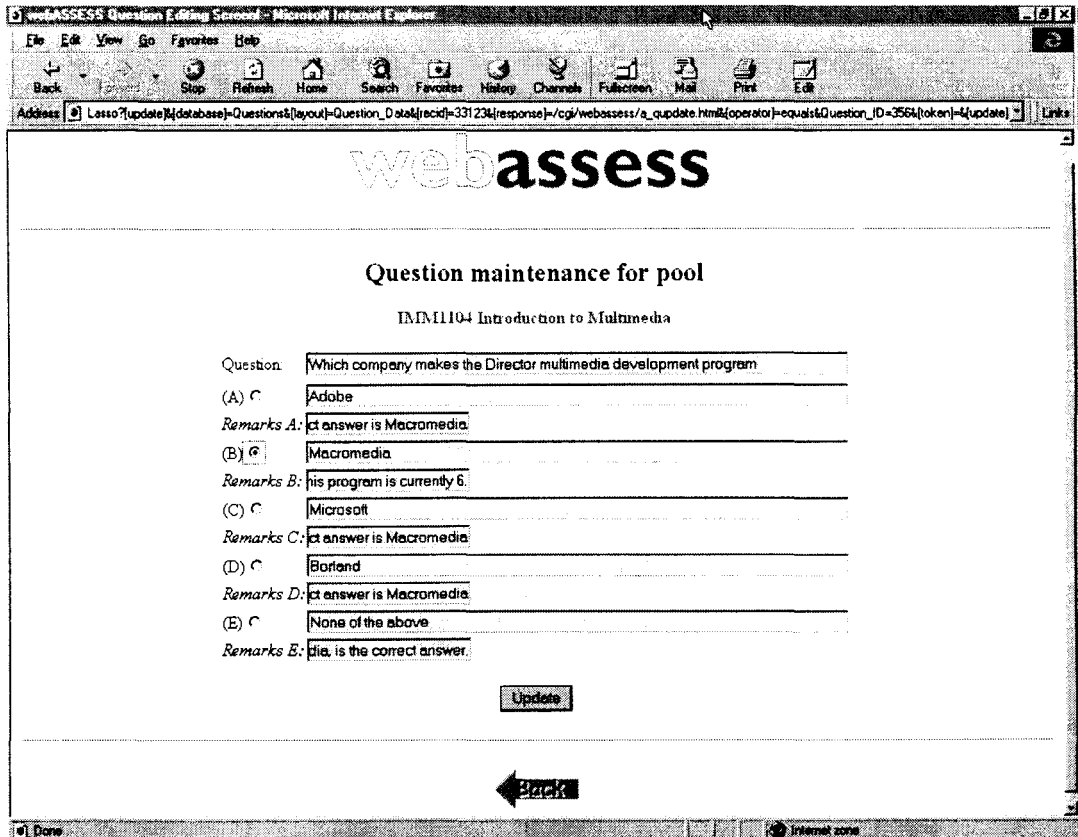


Figure 23: WEBAssess Edit Question Screen.

- This screen shows the actual fields into which the question, options and remarks are entered. Once the appropriate information has been typed into each field, click once on the radio button located to the left of the correct answer.
- Now click the *Update* field to enter this information into the database.

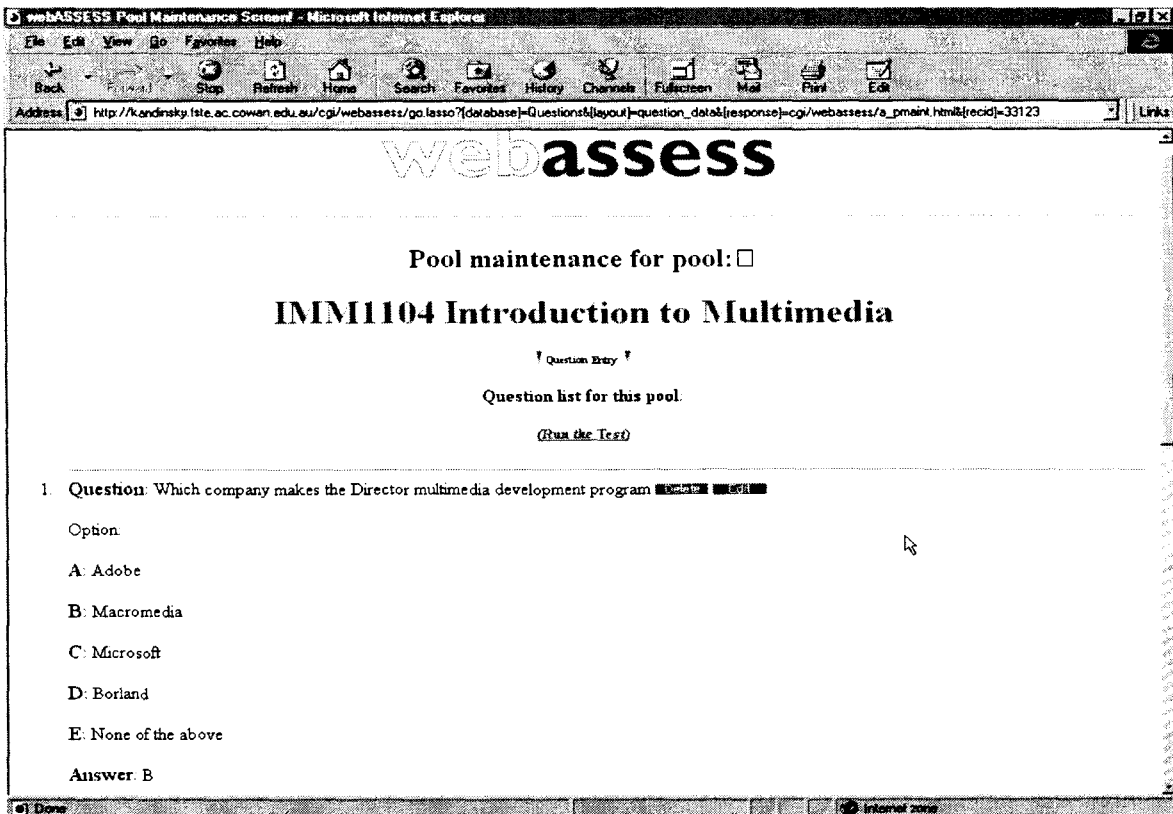


Figure 24: WEBassess Edit Question Result Screen.

### Add a Question

- This screen once again shows the list of questions within this particular pool, though the blank question has been replaced by the information updated into the record.
- Now that we have one complete question in the database, we will add another by using the actual Add Question section of this screen. Simply scroll down this screen until the following fields appear:

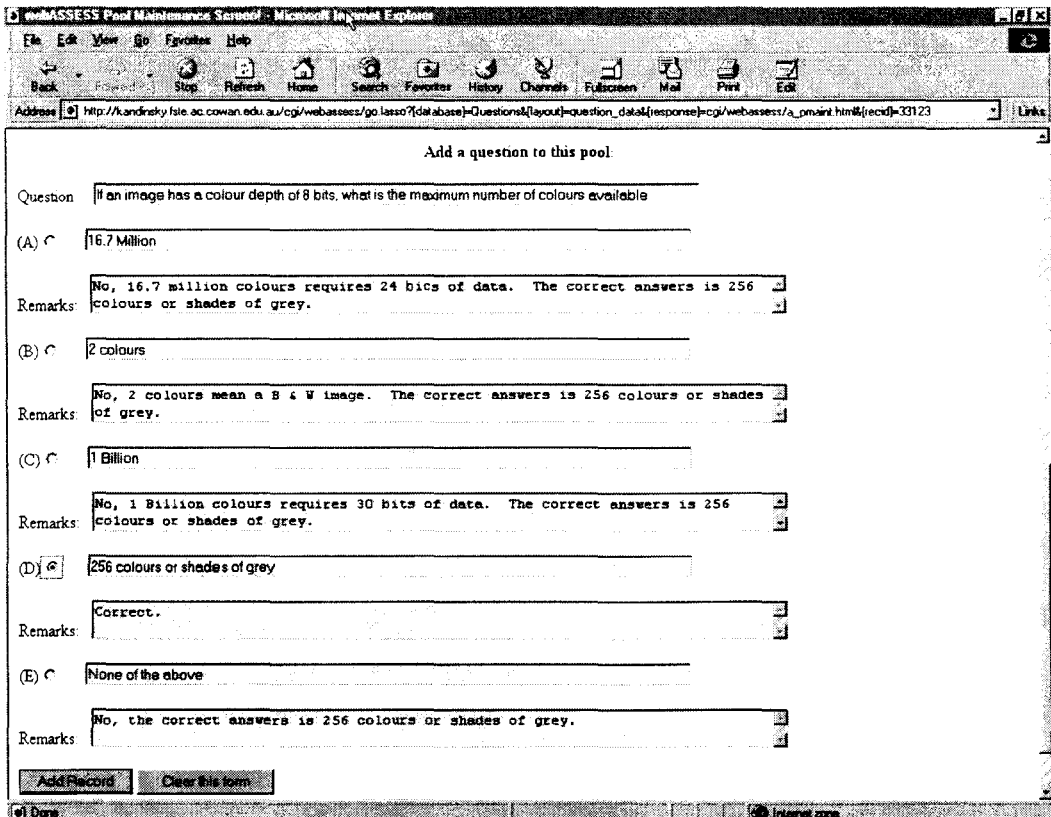


Figure 25: WEBAssess Add Question Screen.

- Entering questions in these fields is exactly the same procedure as editing a blank question. Simply enter the information into the appropriate question, options and remarks field, then click the **Add Record** button.
- This will re-load the current page, simply adding the new question to the bottom of the existing question list, as shown in the following figure:

### Delete a Question

- To delete a question from a pool, simply click the blue Delete button to the right of each question. This will cause the page to re-load, minus the question which has been deleted.

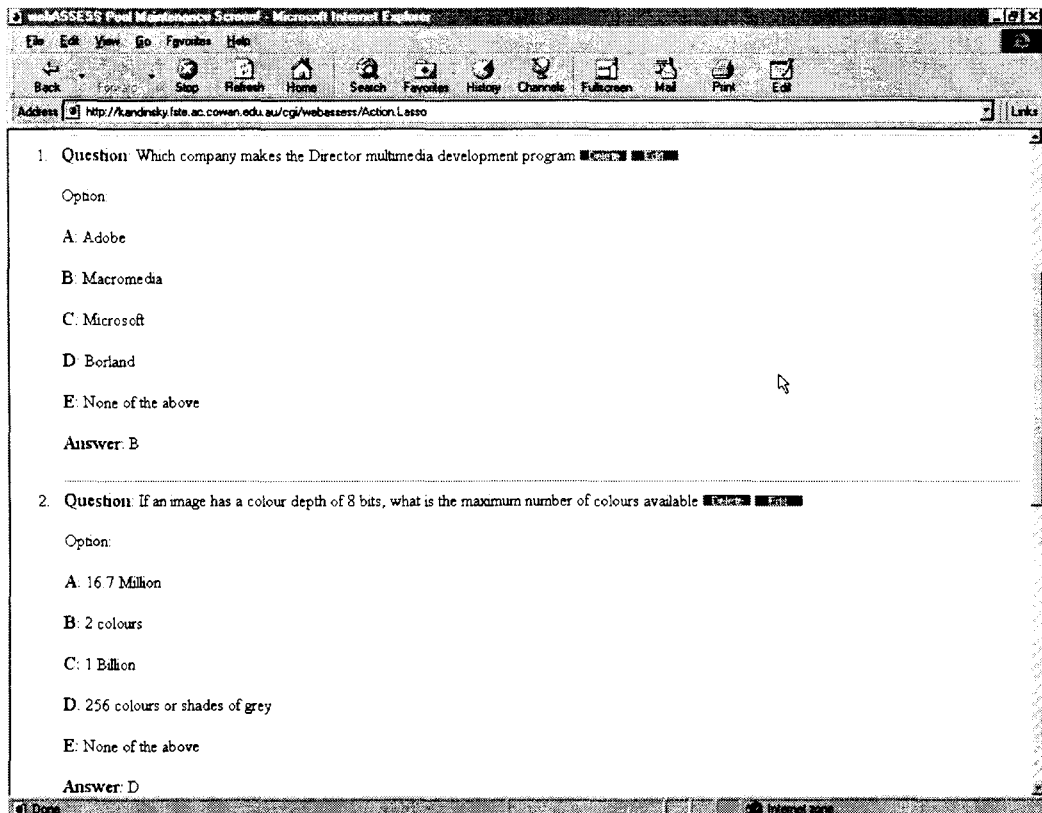


Figure 26: WEBassess Add Question Result Screen.

### Run a Test

- Now that we have two questions in our question pool, it is time to see how the test works, seen from a student's point of view. At the top of the current screen is a hyperlink called (Run the Test). Click on this link to go to the testing screen shown in the following figure:

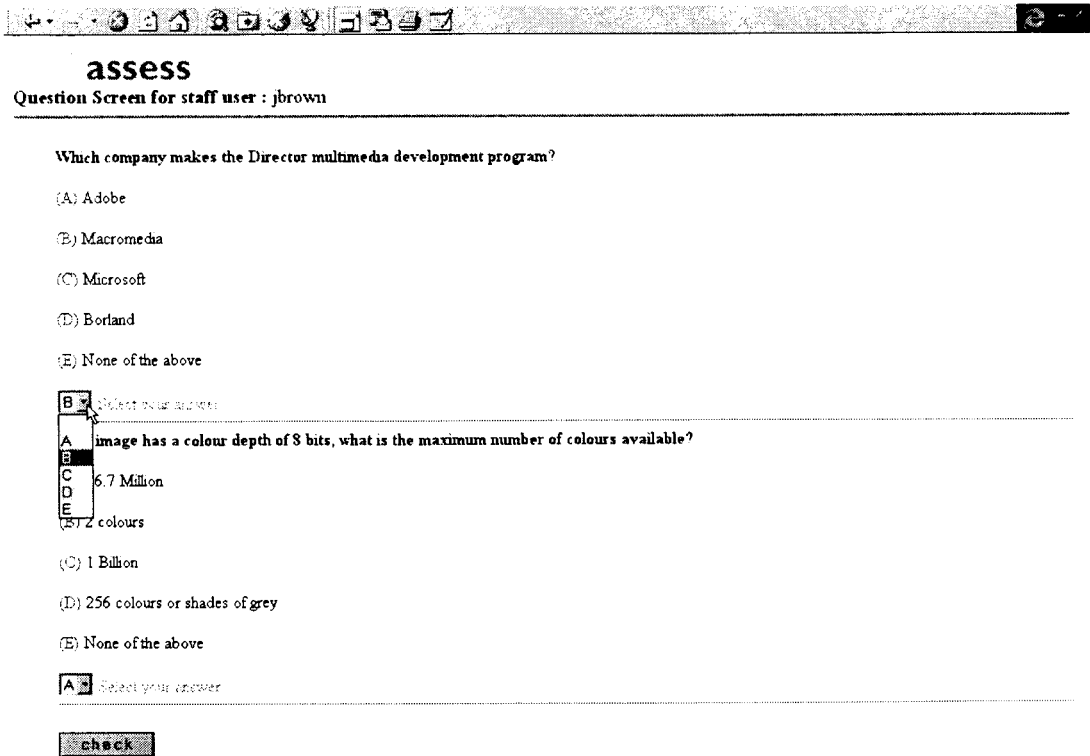


Figure 27: WEBassess Run Test Screen

- Simply select the desired answer to each question by selecting an option from each drop-down list. Once all questions have been answered, click the **check** button to see a score for the test. The following figure shows all the information a student might see upon submitting a completed test:





## assess

Result screen for staff user: jbrown

---

Your score was: 1 out of 2 (50.00%)

Submitted 26/11/97 at 23:05:20

1. Which company makes the Director multimedia development program?

your answer: Macromedia ✓

Correct. The latest version of this program is currently 6.

---

2. If an image has a colour depth of 8 bits, what is the maximum number of colours available?

your answer: 16.7 Million ✗

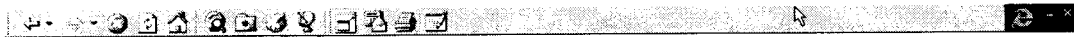
No, 16.7 million colours requires 24 bits of data. The correct answers is 256 colours or shades of grey.

correct answer: 256 colours or shades of grey

Figure 28: WEBassess Run Test Score Result Screen

- This figure shows the score for the test, the time and date of submission, the users name, visual indicators for correct and incorrect answers and remarks, explaining why an answer is deemed correct or incorrect.
- To return to the question pool maintenance screen, simply scroll to the bottom of this result screen and click the ***Back to Question List*** button.





# webassess

Search results for user: Joseph Bloggs

1. Unit code: ist1143 -Test title: short answer questions  
Total score: 2 out of 6 Percentage: 33.33  
Submitted 25/11/97 at 23:01:00
2. Unit code: sct3251 -Test title: Science and Technology Policy  
Total score: 2 out of 3 Percentage: 66.67  
Submitted 25/11/97 at 23:04:54
3. Unit code: lmt2223 -Test title: Revision questions  
Total score: 2 out of 3 Percentage: 66.67  
Submitted 25/11/97 at 23:05:47
4. Unit code: IST1132 -Test title: Week 4 Workshop  
Total score: 1 out of 2 Percentage: 50.00  
Submitted 25/11/97 at 23:21:54



Figure 30: WEBAssess Student Results Screen

- This screen shows the results for all tests that a student has submitted. Unit code, test title, score, percentage, date and time of submission are the fields shown here.

## Navigation

- The figure below shows the arrow icon, which is to be found at the bottom of most screens within the WEBAssess system. Clicking the button will take you back to the page from which you have just exited.



Figure 31: WEBAssess Back Arrow Icon

## APPENDIX C - FORMS

# Project Evaluation

## Consent Agreement

The project I am currently working on is a World Wide Web based testing and assessment system known as WebASSESS.

The key aim of WebASSESS is to allow students to sit multiple choice tests through a web browser, and receive their grade immediately upon completion of the test. It is envisaged at this point that students will work through a series of coursework study materials before actually sitting a test based on that material.

I would greatly appreciate any feedback that you may wish to offer at this time, and I invite you to enter your reaction and comments into the spaces provided below. This evaluation is intended to offer knowledge of the system's usability to both yourself and I.

This is a totally voluntary evaluation which you do not need to take a part in if you do not wish to. No names or any other identification details will be taken for the purposes of this evaluation, and the example test provided has no bearing on your coursework whatsoever.

Many thanks,

Justin Brown.

---

### Questions

1. Did you find the interface logical and easy to understand?  
 Yes  No
2. Did you find the method for answering questions logical?  
 Yes  No
3. Would you find answering 50-60 multiple choice questions easier than a standard multi-format test?  
 Yes  No
4. Did the program function properly for you?  
 Yes  No
5. Did you like the method of feedback?  
 Yes  No
6. Would you like more or less detail in the results feedback form?  
 More  Less
7. In a timed exam situation, would you prefer to sit a test like this one rather than a normal, multi-format exam?  
 Yes  No

Comments:

---

Submit Form

Reset

Figure 15: WEBassess Student User Evaluation and Consent Form.

# Project Evaluation Consent Agreement

The project I am currently working on is a World Wide Web based testing and assessment system known as WebASSESS.

The key aim of WebASSESS is to allow lecturers to create multiple choice tests through a web browser which will allow students to receive their grade immediately upon completion of the test. It is envisaged at this point that lecturers will create a series of multiple choice tests based upon coursework study materials.

I would greatly appreciate any feedback that you may wish to offer at this time, and I invite you to enter your reaction and comments into the spaces provided below. This evaluation is intended to offer knowledge of the system's usability to both yourself and I.

This is a totally voluntary evaluation which you do not need to take a part in if you do not wish to. No names or any other identification details will be taken for the purposes of this evaluation.

Many thanks,

Justin Brown.

---

## Questions

1. Did you find the interface logical and easy to understand?  
 Yes  No
2. Did you find the method for inputting questions logical?  
 Yes  No
3. Would you find entering 50-60 multiple choice questions easier than a creating a standard multi-format (ie multiple choice - short answer - essay) test?  
 Yes  No
4. Did the program function properly for you?  
 Yes  No
5. Did you feel that the level of feedback to students is appropriate?  
 Yes  No
6. Would you like more or less detail in the student results feedback form?  
 More  Less
7. Given the design and capability of the program, do you feel that it could be a useful tool for the creation of tests based on unit coursework?  
 Yes  No
8. In its' current form, do you feel the WebASSESS interface is efficient enough to create large collections of questions?  
 Yes  No
9. Do you believe students would prefer such an on-line test in preference to a standard written test?  
 Yes  No
10. Are there any features missing from WebASSESS which you feel would enhance its' capabilities or make it easier to use? If you select Yes, please describe your ideas and any other thoughts in the Comments field below!  
 Yes  No

Comments:

---

Figure 16: WEBassess Staff User Evaluation and Consent Form.