Virtual Learning Environment For

C Programming

by

FARYANIE ABDULLAH

Dissertation submitted in partial fulfillment of

the requirement for the

Bachelor of Technology (Hons) Information Technology

JUNE 2005

Universiti Teknologi PETRONAS

Bandar Seri Iskandar

31750 Tronoh

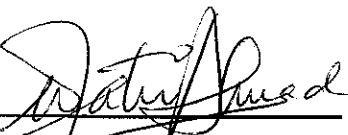Perak Darul Ridzuan

CERTIFICATION OF APPROVAL

Virtual Learning Environment For C Programming

by

Faryanie Abdullah

A project dissertation submitted to the
Information Technology Program
Universiti Teknologi PETRONAS
In partial fulfillment of the requirements for the
Bachelor of Technology (Hons) Information Technology

Approved by,

_____

(Dr. Wan Fatimah Wan Ahmad)

UNIVERSITI TEKNOLOGI PETRONAS
TRONOH, PERAK
June 2005

CERTIFICATION OF ORIGINALITY

This is to certify that I am responsible for the work submitted in this project, that the original work is my own except as specified in the references and acknowledgements, and that the original work contained herein has not been undertaken or done by unspecified sources or persons.

_____

FARYANIE ABDULLAH

ABSTRACT

The use of computers and communication technologies in learning has a history going back at least 30 years. Recent development in Information Technology establishes a new education paradigm, for which self-directed learning is foundation strategy. Nowadays, web-based learning, which has the potential to affect fundamental changes in the design of learning process and the education system, has been gaining momentum with an irreversible trend. A Virtual Learning Environment is one of the examples of web-based learning methods. It is an area of computing and intelligent systems which is becoming increasingly important in both research and industry. These technologies have been adopted by many educational institutions to explore better interaction and dynamics changing on teaching-learning environment. The virtual learning system act as virtual tutors and learning companions that help learners in learning. This report is a final year project involving a research and a mini-version of prototypical Virtual Learning Environment, teaching C Programming. A software development methodology has been created and proposed, namely "Water-Spiral" model, combining the Waterfall model and the Spiral model. The main goal of developing the Virtual Learning Environment of C Programming is to utilize the solid foundation of intelligent system as a platform for a more sophisticated control of multimedia objects in a learning environment. The desirability of new instructional media which allows students to interact with course materials in more exploratory manner has become a dominant theme in current educational technology development. However, computer-based lecture already has a relatively long history and has been shown to positively influence the amount of material learned, the time taken to learn it, and the enjoyment of the learning experience. This system will provide attractive lecture notes, interactive problem-solving situations appropriate to the level of the student and track the student's performance.

ACKNOWLWDGEMENT

TABLE OF CONTENT

LIST OF FIGURES

ABBREVIATIONS AND NOMENCLATURES

WWW	:	World Wide Web
HTML	:	Hypertext Markup Language
KBS	:	Knowledge Based System
IMMS	:	Intelligent Multimedia System
GUI	:	Graphical User Interface
VLES	:	Virtual Learning Environment System
MLE	:	Managed Learning Environment

CHAPTER 1

INTRODUCTION

1   INTRODUCTION

Chapter 1 features the basic information of the project, consisting of its background, its problem statement, its objective and the scope of the study involved. This report arises out of two main areas of interest that will be discussed furthermore: first, to explore the research issues involved in introducing educational technology, namely Virtual Learning Environment Systems into these types of training environments. These issues include knowledge acquisition, system design and implementation, and ultimately, testing and evaluation of the effectiveness of this methods. The second areas is to produce rapid prototypes of some of the different modules in order to demonstrate proof of concept, and to justify the further investment in supplementing and/or replacing the traditional courseware with this new modality.

1.1   Background Study

Nowadays, our society has entered into the information age, in which people intend to acquire the information as quickly as possible. The widely used World Wide Web (WWW) can be featured by the high availability in Internet, short learning time in using the graphic interface browsers and open standard, and portability between different operating systems and platforms. The information released in the Internet is no longer limited to text and graphics as normally presented in the conventional textbooks. The linked material could be in variety of formats, including sound, picture, video, 3-D modeling, animation or application. When new packages of educational materials are developed, these characteristics are capable to enhance the feasibility of the standard HyperText Markup Language (HTML) generally used in the World Wide Web [1].

If a close watch and detailed review are made on the existing educational systems and the population of students under these systems, quite a wide ranges and variations among students can be detected. In recent years, the demand for higher education at different age groups has been increased, particularly in adult age group. The existing of the Virtual Learning Environment Systems will help not only students but even the working adults to get better educational quality so as to enhance the productivity. As such, if working adults under time and location constraints desire to earn money in order to improve living conditions for their family, they have to upgrade themselves through various channels. The Virtual Learning Environment Systems becomes one of these viable alternatives.

With the recent technological advancements in the fields of computer technology, artificial intelligence technology, education and instructional technology, web-based instruction has been becoming somewhat common place in what would be considered traditional higher educational settings [2]. Many schools try to conduct a minimum of some types of web-based instruction. Additionally an entire new industry

of web-based instruction has risen to compete with the universities in this endeavor [3]. A knowledge-based system (KBS) is able to encapsulate systematically the heuristic expertise and knowledge and to accomplish a level of performance comparable to that of a human expert. It is considered suitable for solving problems that demand considerable expertise, judgment, or rules of thumb, which can broadly classified into the following categories: interpretations; design; diagnosis, education and planning. KBS has been widely and successfully applied in different fields [4-8].

One of the vital drawbacks on traditional teaching and learning system is that it offers no facility in tracking student's progress or keeping courseware up-to-date. It is not enough just to supply materials to the learners. It has to monitor whether they have accessed the pertinent information, or whether they need to be chassed at times. It could be monitored and recorded. However, new technology is invented to change this conventional teaching and learning systems. Educators can now deliver far more sophisticated and effective pedagogical programs, through the implementation of Virtual Learning Environment Systems (VLES) [9].

Furthermore, VLES provides the opportunity for interaction from the learners, thus allowing them to tailor learning experiences to meet their specific needs or abilities. The VLES allows a student to dip-in and dip-out of the different course sections. As such, participants can directly access to different sections, and if they encounter any queries, they can point straight back into the pertinent sections or into the references and back again. It supports an active and dynamic learning environment and offers the possibility of providing a stimulating learning environment to engage learners in meaningful learning reflection, application, and interaction. Some organizations use VLES in order to raise the effectives of their education and training operations for the ultimate goal in enhancing the productivity.

This paper delineates the development and implementation of an Virtual Learning Environment System learning package on C programming, by employing the latest KBS technology and web production software. The VLES offers various features for learners as well as for teachers. The most important are handling of multimedia documents, easy to use Graphic User Interface, On-line Help, easy to integrate multimedia documents, easy to manage course material, online discussion. The online discussions and email provide opportunities for all the learners to participate in the discussion of key issues as well as contributing to flexible learning approach. The other features are online quiz template which, helps putting quizzes online, learners and teachers registration interface.

1.2   Problem Statement

The Virtual Learning Environment System can be considered as a solution to the traditional teaching and learning system problem. It has a number of limitations that must be considered and studied before deciding to develop the Virtual Learning Environment Systems.

1.2.1   Access and Resources

To develop VLE systems, it is a must to have an access to the internet and computers to enable the access for both end-users; students and lecturers. The problem of the traditional teaching and learning system is the access to the resources are very difficult seems the students need to get the hard copies own their own from the lecturers. This problem may cause to lake of time and poor in handling the course documents. Whether students have access to the web and what type of access they have is an important consideration in the design of the Virtual Learning Environment Systems.

1.2.2   Cost

For more traditional, on-campus students, access to the web are available at no charge via on-campus computer networks. It is free for the students to get the course material online. Compared to the traditional learning system, students need to buy A4 papers to print out the lecture notes and this will cause money. Any form of usage based charging can lead to student anxiety about making extensive use of the systems. Anxiety and learning do not make a good combination [9].

1.2.3 Training

Although the use of the web is becoming widespread, many students will require some sort of training on how to use it. It is also important to recognize that for many people, the web is a very intimidating environment with its own rules and regulations. It is important that both the students and staff are comfortable with the new environment of the Virtual Learning Environment Systems.

1.2.4 Infrastructure, Support, and Administrator

The changes brought on by the adoption of a web-based classroom do not stop with the educator and the students. A web-based classroom requires infrastructure, support, and administrative procedures that are quite different from those of a traditional classroom. Especially if the move to VLE systems is widespread within an institution, the resources allocation and administrative procedures of an institution must change.

1.2.5 Copyright, Privacy, Security, and Authentication

The issues of copyright, privacy and authentication are all important to consider in any classroom, not just a web-based classroom. Solutions to these problems are still a long way away for both VLE systems and traditional classrooms. Current copyright laws are consistent and are still grappling with the demands of long-standing media such as print, painting, and music, so it's difficult to apply them to the fast-moving world of the web.

1.3   Objectives and Scope of study

This project is divided to two phases. In the first phase, the concept of Virtual Learning Environment System is to be studied in detail. In second phase, the design and development of building a database to track and record the students performance, implementing a prototype of an Virtual Learning Environment Systems teaching C programming, and applying a proper user interface design are to be carried out.

As such, the objectives of this study on Virtual Learning Environment Systems of C Programming comprise the following:

1. To apply the VLE systems of C Programming to all learners.
2. To offer a user-friendly environment for the users to learn C Programming using computer aided learning techniques.
3. To teach C programming through VLE systems.
4. To provide less expensive, more convenient and user-friendly teaching and learning tools for learners.
5. To make the users feel inspired and attracted to learn through the Internet instead of textbook.
6. To progress from using the internet to augment courses, to create virtual courses.
7. To develop a tool aiming at the integration of textual information and graphical expedient.
8. To create a simple and yet attractive application – Virtual Learning Environment Systems of C Programming prototype. As for this project, flash elements will be included in order to develop eye catching interfaces besides to attract the users' interest.

CHAPTER 2

LITERATURE REVIEW AND THEORY

2   LITERATURE REVIEW AND THEORY

Chapter 2 features a specific and more focused scope on Virtual Learning Environment Systems concepts and methods. It provides the background information on how the VLE systems can affect the learning environment for learners. This chapter contains the acknowledged findings on this topic, consisting of relevant theories, hypothesis, facts and information which are relevant to the objective and the research of the project. An overview of a Virtual Learning Environment System and the preliminary remarks on the theory of Virtual Learning Environment System will be introduced. More elaborations are to be given on subjects about the benefits of a Virtual Learning Environment Systems, a review of some learning environment, a framework for integrating multimedia and agents for VLE systems, and some details of the VLE systems.

2.1   What is *Virtual Learning Environment System?*

There is some interchangeability and indeed confusion regarding the terms applied to VLEs. This section looks at common definitions and delineates the functionality these terms represent. Based on the definition by Oxford Dictionary (1974), Virtual reality (VR) describes an environment that is simulated by a computer. Most virtual reality environments are primarily visual experiences, displayed either on a computer screen or through special stereoscopic goggles, but some simulations include additional sensory information, such as sound through speakers. There is growing interest in the use of Virtual Learning Environments (VLEs) by schools and local education authorities.

Although there is some confusion about the definition of Virtual Learning Environments, they are generally a combination of some or all of the following features:

- communication tools such as email, bulletin boards and chat rooms
- collaboration tools such as online forums, intranets, electronic diaries and calendars
- tools to create online content and courses
- online assessment and marking
- integration with school management information systems
- controlled access to curriculum resources
- student access to content and communications beyond the school.

Britain & Liber (1999) in their evaluation of VLEs state that their definition of VLEs describes systems which are known as online learning environments, learning management systems and collaborative learning software [10]. These definitions imply a high level of interaction between the VLE and the surrounding MLE. The principal functions of this interaction would be:

1. Controlled access to the curriculum, which has been mapped to elements that can be separately assessed and recorded.

2. Tracking of student activity and achievement against these elements, using simple processes for tutors to define and set up a course with accompanying materials and activities to direct, guide and monitor learner progress.

3. Support of online learning, including access to learning resources, assessment and guidance; the learning resources might be self-developed or professionally authored and purchased, and can be imported and made available for use by learners.

4. Communications between the learner, the tutor and other learning support specialists to provide direct support and feedback for learners, as well as peer group communications that build a sense of group identity and a community of interest.

5. Links to other administrative systems, both in house and externally.

In addition to these features, it is generally regarded that:

1. There will be a level of security built into the system, including a form of password protection.

2. VLEs normally provide two views of the system, one for the tutor and one for the student

3. Tutors will have access to a wide range of tools and privileges in a VLE that allow them to add materials, create tools and track student progress

4. MLEs and VLEs will be browser based and use web technologies, but will not require knowledge of HTML in order to use or contribute content to the system

5. The use of web technology for VLEs means that the system can be accessed anywhere, including at school and home; with portable technologies, the 'any time, anywhere' learning model will be possible.

6. The elements that go together to make up the system should interrelate and be interoperable, allow for the sharing of data, and provide a consistent interface for students and staff.

The above definitions of VLESs are relatively recent, and involve two significant changes. First, until the definitions were clarified as above, most people (including suppliers) used the term Managed Learning Environment (MLE) to refer to what is now considered to be a VLE. Secondly, the definition of the term VLE has been narrowed. Given the functions above, a system which was purely a conferencing and communications system, with no place to put courseware, would not qualify as a VLE, nor would a system which had content, but no way of tracking student activity and achievement against that content. To be a VLE, a system must contain, at least to some extent, all the principal functions defined above. However, the last function (the links to other administrative systems and the interoperability implied by this) is the least developed.

2.2   International research on *Virtual Learning Environment System?*

This section outlines findings from research on the use of VLESs in higher education schools in countries other than the UK.

2.2.1   Current situation

WebCT and Blackboard are both widely used throughout universities. There is consequently a greater amount of evidence on these products than those from other developers, and this report reflects that. This, however, does not imply endorsement of these products, nor does the omission of other products imply the opposite.

2.2.2   Research findings

Most of the studies identified to date seek to evaluate specific aspects of WebCT and Blackboard. International differences and the anecdotal nature of much of the evidence necessitate caution when assessing the wider applicability of these findings.

In the evaluation of Blackboard at Haverford College (USA), tutors stress both the ease of system management and course creation:

> *'It took me very little time to figure out how to do what I wanted to do without even having to resort to documentation... It was very easy to add most forms of content. Adding or subtracting elements/tools from a course is pretty easy. Administration was easy to understand.'*

(Course Management Systems Evaluation Committee 2001)

Adelaide University (Australia), reports positive feedback from the testing of Blackboard, and highlights specific features:

> *'the personalised portal interface that provides quick access according to enrolment and/or your role in the institution; easy navigation and use for students accessing material and staff uploading material; the flexible interface and ability to use Blackboard as the front-end that then provides links to further material; the integrated nature of the software e.g. e-mail, calendar, discussion forums, course material, online chat rooms, announcements and staff details are all accessible within the one package.'*

(DVCE 2000)

At the University of Western Sydney, a comparative review of WebCT and Blackboard was carried out to assist in strategic decision-making and planning in the choice of a course management system (Rankine 2001). In regard to WebCT, the report highlighted its customisability and flexibility, which make it suitable for both beginners and the more advanced online teachers and learners. At the University of

Texas at Austin, WebCT was chosen again. This time the features that allow complete creation of online courses were praised:

> *'Of the products recommended, WebCT is the most feature-rich and provides the closest approximation of a package that allows complete creation of online courses.'*

(Center for Instructional Technologies 2001)

The ability of a VLE to enable distance learning, and the value of specific features, is examined by Musgrove (2001) [11]. Musgrove sees the growing sophistication of WebCT in its latest incarnation as assisting universities in their delivery of web-based learning, through such features as an improved designer interface, a rich variety of communication tools and the capability to customize course delivery to suit individual preferences. A broad implementation of WebCT as a support to traditional teaching techniques within 20 courses at Creighton University (USA) was the subject of a study during 1997-98 (Morss & Fleming 1998) [12]. A survey based on the results of the first semester of use drew these conclusions about WebCT computer-aided instruction:

i. it is not time-consuming; most students connected to WebCT for four hours a week or less, though some students printed off pages to read at their desks rather than on screen. Student opinion was split in terms of the workload imposed: older students thought it increased yet the same proportion of younger students believed it did not

ii. the level of use of the system matches students' computer abilities, although the system does require a number of skills that are not achieved without training

iii. the central tools (Path Editor and Bulletin Board) are relatively easy for students to use

iv. the younger students perceive the system as helping the learning process

v. using WebCT supports the perception of both younger and older students that computer-aided instruction is important for success in their future careers

vi. lack of unambiguous evidence made the researchers cautious in drawing conclusions about the efficacy of the WebCT tools.

Another study of WebCT's use in course delivery was made by the University of British Columbia, where the computer science department involved 730 first- and third-year students in the use of WebCT (Goldberg 1997) [13]. It investigated student acceptance of the system, and the academic effectiveness of various modes of course delivery. Web-based aspects of courses were presented using the WebCT course tool. Courses for both year groups included complete course content online and access to the same set of WebCT tools. The results for student use were as follows:

- third-year students made more use of the online resource and conferencing system than first years
- approximately 30 per cent of the access to the online resources came from outside the university labs, indicating extensive use by students' offsite.

14

Students considered that the online resources had improved their understanding of the course materials. Away from the university environment, Follows (1999) described the use of Thirst for Knowledge, a VLE that simulates the workplace of the Quaker Oats Company, used as part of an introductory marketing course [14]. It was found that using the system to simulate a workplace environment outside the classroom saved time and reduced risk. In addition, the system:

- provided the students with a context for the learning process to take place within
- allowed the students to control the learning process
- developed an ability to solve high-level problems
- made learning a personal experience for the student
- modelled the complexity and uncertainty of working in the real world
- could accommodate a wide range of students' learning styles
- was many students' preferred way to learn.

Assessment

At the University of Calgary, WebCT was used to construct, administer and mark a mid term examination made up of randomly selected items from an item bank, as part of an undergraduate course in computer science. Jacobson & Kremer (2000) reported that the students identified benefits including [15]:

- The flexibility of 'any time, anywhere' access
- Being able to sit the exam at a time most convenient to them
- Being able to set up their work space for the exam
- The open-book nature of the exam, with some students reporting increased confidence because they could access their textbooks and notes.

However, students also perceived certain disadvantages:

> The potential for unethical conduct among their fellow students in an unsupervised exam

> Difficulty in contacting an instructor during the exam, despite contact information for the instructor being made available

> Inability to jot calculations in the margin (unlike the paper-based exam)

> Confusion over time elapsing during the exam and uncertainty about how to set up the workstation

> Problems with home internet access

> Weaknesses of multiple-choice questions as a style, for example it is not possible to demonstrate the thought processes that lie behind an answer, and there is no opportunity for partial marks.

It is clear that if an existing course is redeveloped for online provision via a VLE, technical design and delivery issues will come to light, but training of both staff and students will also need to be addressed. Many of the issues that arose here are related specifically to online assessment. Since online assessment is a function that many will expect a VLE to perform, it is valid to consider its effectiveness. Robertson & Klotz (2001) highlight five specific issues which came to light when converting a course for delivery by WebCT [16]:

- ➤ Students' technological capacity to function as online learners
- ➤ Technological support for students and instructors
- ➤ Time zone and international server problems
- ➤ The compatibility of America Online (AOL) with WebCT
- ➤ Firewall protection blocking the chat function.

Communication

Curtis & Lawson (2001) carried out a study of the use of the computer conferencing component of Blackboard Classroom, with 24 students at the University of South Australia, on a Bachelor of Education course [17]. They investigated the extent to which components of collaborative learning that had been found in studies of face-to-face interactions could be identified in the online interactions of students. An analysis of participants' postings revealed many of the behaviours associated with face-to-face situations. Challenging others and offering explanations were absent, though this may have been as a result of students' unfamiliarity with each other. More social interaction and online self-introductions might help overcome this barrier. The researchers concluded that the medium did influence the students' interactions. They all undertook the course as an off-campus course, and without the Blackboard Classroom software their interactions would have been limited to email. While it would have been possible to set up an email discussion list, to have provided the range of interactions required (group email, the discussion board, file transfer and chat) would have necessitated a range of software tools. Blackboard provided a consistent interface for all the interactions. It was concluded that successful collaboration as described in face-to-face situations is possible in online learning environments.

Nine volunteers from a teacher education programme at Queens University (Canada) participated in a study of how a private WebCT forum may foster self study (Russell 2000). Conclusions were that:

1. Individuals that took the time to participate showed a tendency to reflect on their personal professional learning

2. The forum gave the participating teachers insights into reactions to his/her teaching that would have been difficult to gain in any other way, providing a valuable opportunity to improve teaching.

Participation in the forum was voluntary, and had it not been so, the end result might have been very different. Russell notes that willingness to participate in discussion is not universal, and that patterns of access and use vary widely. While he continues to encourage his students to participate in the forum, he is seeking meaningful alternatives for those who do not wish to take part. Communication tools were also highly appreciated at the University of British Columbia (Goldberg 1997), with 85 per cent of third-year students and 62 per cent of first years claiming that online conferencing produced a greater feeling of community [18]. Similarly, the majority of students in both groups found the course more personal as a result of using the conferencing tool. It was also found that students lacking confidence to participate in class discussions were willing to contribute to an online discussion.

CHAPTER 3

METHODOLOGY AND PROCEDURE

3 METHODOLOGY AND PROJECT WORK

Chapter 3 contains the detailed description of the methodologist and procedures to complete this project, including producing the Virtual Learning Environment Systems teaching C Programming. A newly-created methodology namely "Water-Spiral" is to be introduced and explained. This modal is a combination of the Waterfall model and Spiral model. The waterfall provides an orderly sequence of development steps and helps ensure the adequacy of documentation and design reviews to ensure the quality, reliability, and maintainability of the developed software.

3.1 Software Development Methodology

A development compasses the process, methods, and tools of the software engineering layers. This strategy is usually referred to as a process mode or a software engineering paradigm. Software engineering is the practice of using selected process techniques to improve the quality of a software development effort. This is based on the assumption, subject to endless debate and supported by patient experience, that a methodical approach to software development results in fewer defects and, therefore, ultimately provides shorter delivery times and better value. The documented collection of policies, processes and procedures used by a development team or organization to practice software engineering is called its software development methodology (SDM) or system development life cycle (SDLC).

The challenge in selecting and following a methodology is to do it wisely. It helps to provide sufficient process disciplines to deliver the quality required for business success, while avoiding steps that waste time, squander productivity, demoralize developers, and create useless administration. The best approach for applying a methodology is to consider it as a means to manage risk. Risk can be identified by looking at past projects and based on own experiences.

Every step along the system development life cycle has its own risks and a number of available techniques to improve process discipline and resulting output quality. Moving through the development life cycle, the following major steps might be encounter:

- Definition of the project process and project requirements
- Documentation of user, functional and system requirements
- Top level architecture, technical approach, and system design
- System decomposition into component and unit specifications and design
- Coding, unit test planning, and unit test
- Generation of test data for unit testing and system testing
- System integration and testing
- Implementation, delivery and cut-over
- Training and user support
- System upgrades and routine software maintenance

Written guidance for all these steps would constitute the core of the methodology used. It wouldn't take long to fill a number of big binders with development processes and procedures. Hence, the importance of selecting processes wisely (to address known risks) keeping the methodology streamlined, and allowing for some discretion on the part of the project team.

3.2   Waterfall Model

The waterfall model is a popular version of the systems development life cycle model for software engineering. Often considered the classic approach to the systems development life cycle, the waterfall model describes a development method that is linear and sequential. The waterfall model derives its name due to the cascading effect from one phase to the other as is illustrated in Figure 3.1. In this model each phase well defined starting and ending point, with identifiable deliveries to the next phase. Waterfall development has distinct goals for each phase of development. Imagine a waterfall on the cliff of a steep mountain. Once the water has flowed over the edge of the cliff and has begun its journey down the side of the mountain, it cannot turn back. It is the same with waterfall development. Once a phase of development is completed, the development proceeds to the next phase and there is no turning back.

The advantage of waterfall development is that it allows for departmentalization and managerial control. A schedule can be set with deadlines for each stage of development and a product can proceed through the development process like a car in a carwash, and theoretically, be delivered on time. Development moves from concept, through design, implementation, testing, installation, troubleshooting, and ends up at operation and maintenance. Each phase of development proceeds in strict order, without any overlapping or iterative steps.
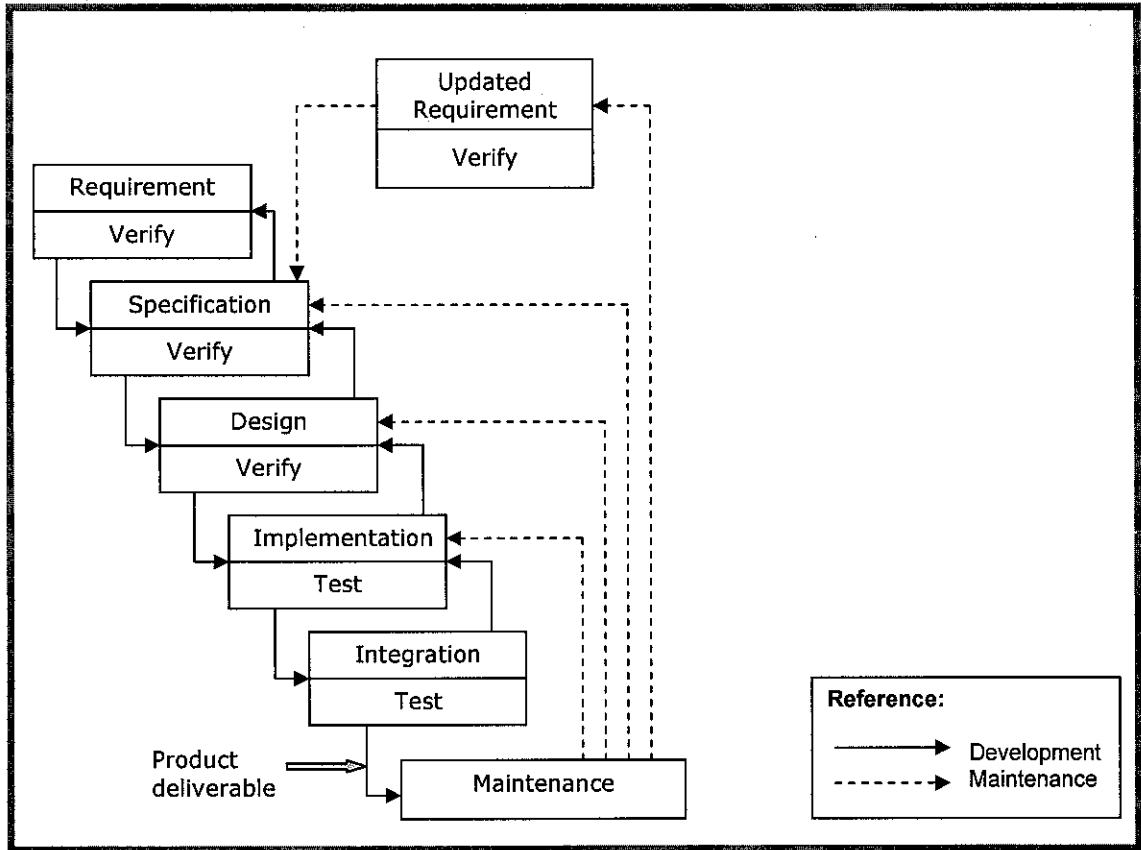
Figure 3.1: Waterfall model

The waterfall model consists of six distinct stages:

1. In the *requirements analysis* phase

   (a) The problem is specified along with the desired service objectives/goals

   (b) The constraints are identified

2. In the *specification* phase the system specification is produced from the detailed definitions of problems and constraints in the requirements analysis phase. This document should clearly define the product function.

22

3. In the system and software *design* phase, the system specifications are translated into a software representation. The software engineer at this stage is concerned with:

    - Data structure

    - Software architecture

    - Algorithmic detail and

    - Interface representations

    The hardware requirements are also determined at this stage along with a picture of the overall system architecture. By the end of this stage should the software engineer should be able to identify the relationship between the hardware, software and the associated interfaces. Any faults in the specification should ideally not be passed 'down stream'.

4. In the *implementation and testing* phase stage the designs are translated into the software domain

    - Detailed documentation from the design phase can significantly reduce the coding effort.

    - Testing at this stage focuses on making sure that any errors are identified and that the software meets its required specification.

5. In the *integration and system testing* phase all the program units are integrated and tested to ensure that the complete system meets the software requirements. After this stage the software is delivered to the customer.

6. The *maintenance* phase the usually the longest stage of the software. In this phase the software is updated to:

- Meet the changing customer needs

- Adapted to accommodate changes in the external environment

- Correct errors and oversights previously undetected in the testing phases

- Enhancing the efficiency of the software

Observe that feed back loops allow for corrections to be incorporated into the model. For example a problem or update in the design phase requires a 'revisit' to the specifications phase. When changes are made at any phase, the relevant documentation should be updated to reflect that change.

3.3   Spiral model

Spiral model is a classic approach to project management. The spiral model is an iterative approach that begins by developing basic requirements, building small simple prototypes, evaluating those prototypes, and then expanding into refined requirements and consecutively larger prototypes, until a complete deliverable product is created. Prior to the final product, each prototype is deliberately built with the expectation that it will be thrown away. It combines the iterative nature of prototyping with the controlled and systematic aspects of the waterfall model, therein providing the potential for rapid development of incremental versions of the software. In this model the software is developed in a series of incremental releases with the early stages being either paper models or prototypes. Later iterations become increasingly more complete versions of the product.
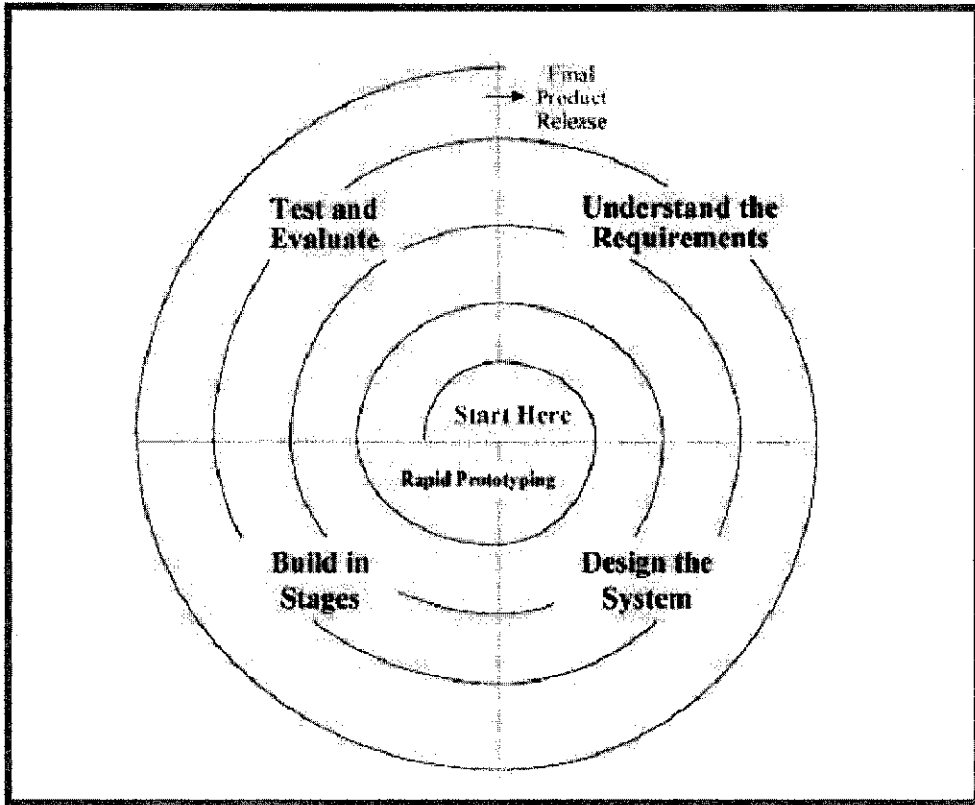
Figure 3.2: Spiral model

The evolutionary process begins at the centre position and moves in a clockwise direction. Each traversal of the spiral typically results in a deliverable. For example, the first and second spiral traversals may result in the production of a product specification and a prototype, respectively. Subsequent traversals may then produce more sophisticated versions of the software.

An important distinction between the spiral model and other software models is the explicit consideration of risk. There are no fixed phases such as specification or design phases in the model and it encompasses other process models. For example, prototyping may be used in one spiral to resolve requirement uncertainties and hence reduce risks. This may then be followed by a conventional waterfall development.

25

- Each passage through the planning stage results in an adjustment to the project plan

- Each of the regions is populated by a set of work tasks called a task set that are adapted to characteristics of the project to be undertaken. For small projects the number of tasks and their formality is low. Conversely, for large projects the reverse is true.

3.3.1   Advantages of the Spiral Model

- The spiral model is a realistic approach to the development of large-scale software products because the software evolves as the process progresses. In addition, the developer and the client better understand and react to risks at each evolutionary level.
- The model uses prototyping as a risk reduction mechanism and allows for the development of prototypes at any stage of the evolutionary development.
- It maintains a systematic stepwise approach, like the classic life cycle model, but incorporates it into an iterative framework that more reflect the real world.
- If employed correctly, this model should reduce risks before they become problematic, as consideration of technical risks are considered at all stages.

3.4   "Water-Spiral" Model

A software model has been proposed as to assisting in the control and coordination of a real software project. This model is named as "Water-Spiral" model, as shown in figure 3.3 below, has a combination of Waterfall model and Spiral model.
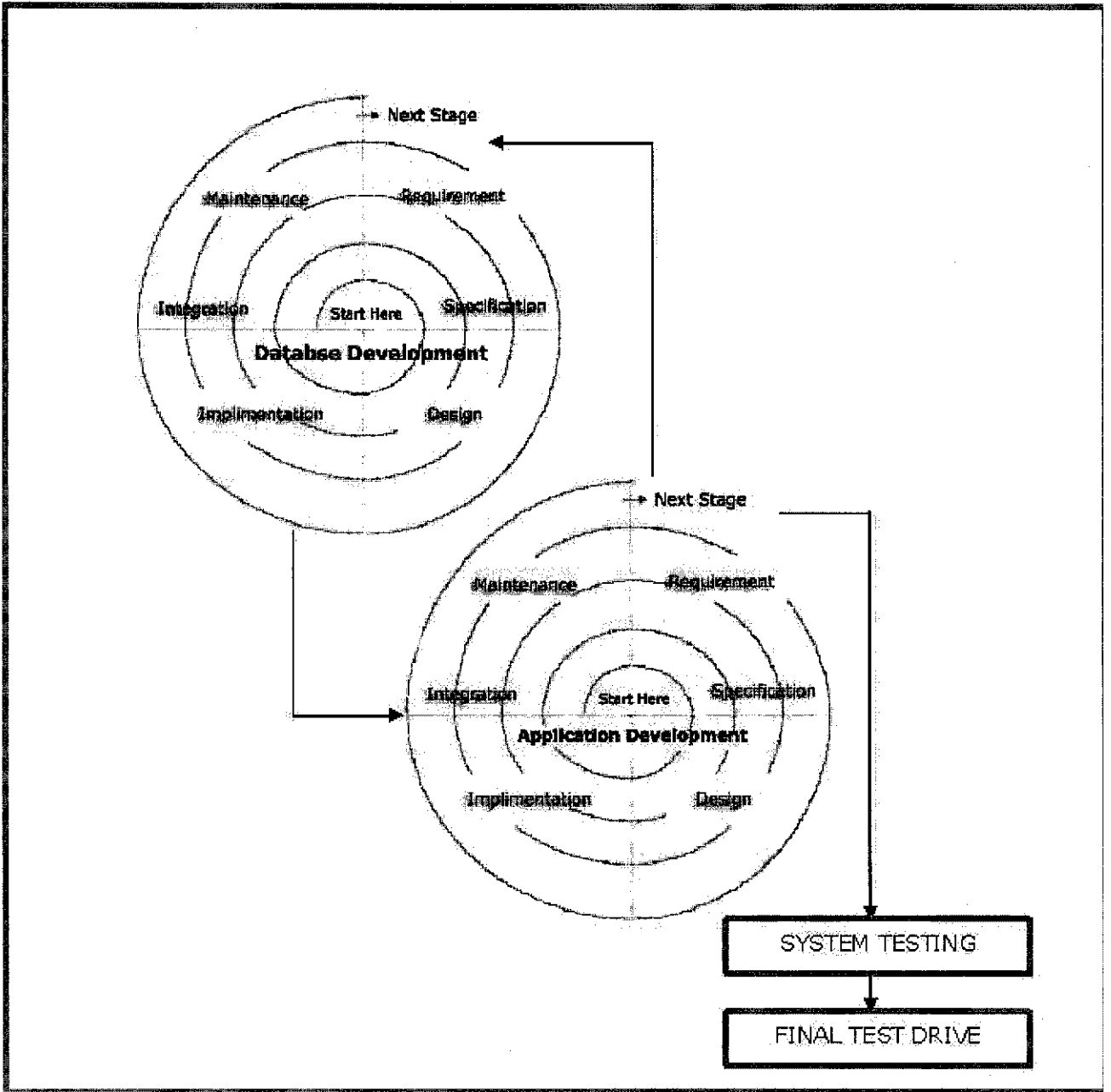
Figure 3.3: "Water-Spiral" model

All projects can be managed better when segmented into a hierarchy of chunks such as phases, stages, activities, tasks and steps. In system development projects, the simplest rendition of this is called the "Waterfall" methodology, as shown in figure 3.1. In looking at the graphic, which was for major defense systems developments, probably is the most important step towards project success. Nevertheless, the graphic illustrates a few critical principles of a good methodology:

- Work is done in stages,
- Content reviews are conducted between stages, and
- Reviews represent quality gates and decision points for continuing.

The waterfall provides an orderly sequence of development steps and helps ensure the adequacy of documentation and design reviews to ensure the quality, reliability, and maintainability of the developed software. While almost everyone these days disparages the "Waterfall methodology" as being needlessly slow and cumbersome, it does illustrate a few sound principles of life cycle development.

While the waterfall methodology offers an orderly structure for software development, demands for reduced time-to-market make its series steps inappropriate. The next evolutionary step from the waterfall is where the various steps are staged for multiple deliveries or handoffs. The ultimate evolution from the waterfall is the spiral, taking advantage of the fact that development projects work best when they are both incremental and iterative, where the team is able to start small and benefit from enlightened trial and error along the way.

The spiral methodology reflects the relationship of tasks with rapid prototyping, increased parallelism, and concurrency in design and builds activities. The spiral method should still be planned methodically, with tasks and deliverables identified for each step in the spiral.

3.4.1 Database Development As First "Water-Spiral" Stage

The database development is the first stage of "Water-Spiral" model. This process will follow all the Waterfall model sequence but in a Spiral model manner (iterative approach). In the requirement analysis phase, the tables needed are listed down together with the attributes. Then, the listed data required will be analyzed in order to specify their relationship. Next, all the tables will be design and develop according to the agreed requirements and specifications. Different levels of testing needs to be carried out. Testing focuses on making sure that any errors are identified and that the software meets its required specification.

In the integration phase, all the tables are integrated and tested to ensure that the complete database meets the system requirements. Lastly, the maintenance phases usually the longest stage of the software. In this phase, the database is updated to meet the changing user needs, correct errors and oversights previously undetected in the testing phase and enhancing the efficiency of the tables created. Spiral model allow for corrections to be incorporated into the model. It maintains a systematic stepwise approach, but incorporates it into an iterative framework that more reflect the real world.

3.4.2 Application Development As Second "Water-Spiral" Stage

The second stage will be exactly same as the first stage. The only difference is it applies to the application development. The sequence of the phases are the same as before which uses Waterfall model sequence and incorporate it with Spiral model which allows an iterative approach that begins with developing the basic requirements, building small prototypes, evaluating those prototypes, until a complete deliverable product is created.

In the requirement analysis phase, the developer needs to start from the compulsory functions and taking feedback from users, before moving to another set of functionalities. This step refers to the gathering of system requirements, with the goal of determining how these requirements will be accommodated in the system. Extensive communication between the end users and the developer is essential. The developer needs to identify the targeted users and understand the technical background of the users. Different types of users may require different types of way to represent or retrieve the information from the system.

Once the requirements have been collected and analyzed, it is necessary to identify in detail how the system will be constructed to perform necessary tasks. More specifically, the System Design phase is focused on the data requirements (what information will be processed in the system?), the software construction (how will the application be constructed?), and the interface construction (what will the system look like and what standards will be followed?). The developer needs to prepare a storyboards and a sets of questionnaire at the initial design phase, and seek users' comments, ideas and preferences before moving on to the Implementation phase. This is to avoid massive rework on the application development.

In the implementation and testing stage, the designs are translated into the system domain. The detailed documentation from the design phase can significantly reduce the coding effort. Coding is also known as programming, this step involves the creation of the system software. Requirements and systems specifications from the System Design step are translated into machine readable computer code. As the software is created and added to the developing system, testing is performed to ensure that it is working correctly and efficiently. Testing is generally focused on two areas: *internal* efficiency and *external* effectiveness.

The goal of external effectiveness testing is to verify that the software is functioning according to system design, and that it is performing all necessary functions or sub-functions. The goal of internal testing is to make sure that the computer code is efficient, standardized, and well documented. Testing can be a labor-intensive process, due to its iterative nature.

Integration phase involves building the system and testing the resultant system for problems that arise from component interactions. These tests should be developed from the system specification and should begin as soon as usable versions of some of the systems components are available. Two categories exist, which are top-down and bottom-up testing. Top-down integration testing is an incremental method to construction of program structure. The modules are integrated by moving downward through the control hierarchy, starting with the main control module.

3.5   Tools Required

On hardware requirements, a personal computer (PC) with specifications of Intel Pentium III, running on windows XP system is needed. Software involved a center of database, utilizing software such as Apache2, PHP4, MySQL4 and phpMyAdmin, which is sufficient to store the data for the project context. One of Macromedia proprietary software, Macromedia Dreamweaver MX is to be used for development and designing the interfaces purpose. Seems this project will be dealing with multimedia components, Macromedia Flash MX and Swish Max will be used to create an animation graphics as well as icons.

CHAPTER 4

RESULTS AND DISCUSSION

4 RESULTS AND DISCUSSIONS

Chapter 4 compiles the current findings of the project work. There have been several interesting and informative information, coming from journals and online resource. It is interesting to highlight that until now, there has not been much learning products that uses Virtual Learning Environment Systems. Basically, the Virtual Learning Environment Systems of C Programming will be consisting of C Programming e-book, lecture notes, test and tutorials. So, we will discuss further more each of the elements on the next sections.

4.1 Database Development

The database, in which MySQL and PhpMyAdmin has been chosen, is created and inserted with data that will be needed to store all the VLES user information. Apache HTTP server will be used as a stand alone server to manage and run the VLES of C Programming online.

4.1.1 Apache HTTP Server Project

The Apache Project is a collaborative software development effort aimed at creating a robust, commercial-grade, featureful, and freely-available source code implementation of an HTTP (Web) server. The project is

jointly managed by a group of volunteers located around the world, using the Internet and the Web to communicate, plan, and develop the server and its related documentation. These volunteers are known as the Apache Group. In addition, hundreds of users have contributed ideas, code, and documentation to the project. This file is intended to briefly describe the history of the Apache Group and recognize the many contributors.

4.1.2 MySQL Database Management System

MySQL is the most popular Open Source SQL database management system, is developed, distributed, and supported by MySQL AB. MySQL AB is a commercial company, founded by the MySQL developers. It is a second generation Open Source company that unites Open Source values and methodology with a successful business model.

- MySQL is a database management system.

    A database is a structured collection of data. It may be anything from a simple shopping list to a picture gallery or the vast amounts of information in a corporate network. To add, access, and process data stored in a computer database, a database management system such as MySQL Server is needed. Since computers are very good at handling large amounts of data, database management systems play a central role in computing, as standalone utilities or as parts of other applications.

33

- MySQL is a relational database management system.

  A relational database stores data in separate tables rather than putting all the data in one big storeroom. This adds speed and flexibility. The SQL part of ``MySQL'' stands for ``Structured Query Language.'' SQL is the most common standardized language used to access databases and is defined by the ANSI/ISO SQL Standard. The SQL standard has been evolving since 1986 and several versions exist. In this manual, ``SQL-92'' refers to the standard released in 1992, ``SQL:1999'' refers to the standard released in 1999, and ``SQL:2003'' refers to the current version of the standard. We use the phrase ``the SQL standard'' to mean the current version of the SQL Standard at any time.

- MySQL software is Open Source.

  Open Source means that it is possible for anyone to use and modify the software. Anybody can download the MySQL software from the Internet and use it without paying anything. The MySQL software uses the GPL (GNU General Public License), to define what the users may and may not do with the software in different situations.

- The MySQL Database Server is very fast, reliable, and easy to use.

  MySQL Server also has a practical set of features developed in close cooperation with other users. A performance comparison of MySQL Server with other database managers can be found on the benchmark page.

  MySQL Server was originally developed to handle large databases much faster than existing solutions and has been successfully used in highly demanding production environments for several years. Although under constant development, MySQL Server today offers a

34

rich and useful set of functions. Its connectivity, speed, and security make MySQL Server highly suited for accessing databases on the Internet.

- MySQL Server works in client/server or embedded systems.

The MySQL Database Software is a client/server system that consists of a multi-threaded SQL server that supports different backends, several different client programs and libraries, administrative tools, and a wide range of application programming interfaces (APIs).

MySQL Server also can be known as an embedded multi-threaded library that can be link into an application to get a smaller, faster, easier-to-manage product.

- A large amount of contributed MySQL software is available. It is very likely that the users favorite application or language supports the MySQL Database Server.

4.1.3 PhpMyAdmin 2.6.0

PhpMyAdmin is used to manage a whole MySQL server (needs a super user) as well as a single database. To accomplish the latter, users need a properly set up MySQL user who can read/write only the desired database. It's up to the person to look up the appropriate part in the MySQL manual.

Currently phpMyAdmin is used to:

- create and drop databases
- create, copy, drop, rename and alter tables
- do table maintenance

- delete, edit and add fields
- execute any SQL-statement, even batch-queries
- manage keys on fields
- load text files into tables
- create (*) and read dumps of tables
- export (*) data to CSV, XML and Latex formats
- administer multiple servers
- manage MySQL users and privileges
- check referential integrity in MyISAM tables
- using Query-by-example (QBE), create complex queries automatically connecting required tables
- create PDF graphics of the Database layout
- search globally in a database or a subset of it
- transform stored data into any format using a set of predefined functions, like displaying BLOB-data as image or download-link
- support InnoDB tables and foreign keys
- support mysql, the improved MySQL extension
- communicate in 47 different languages

4.2   Application Development

There are several considerations to be taken care of, before developing on the system itself. Fields such as the users, level of users, user interfaces, elements to be included and the user requirements need to be checked on, as wrong planning will make these resources unusable and wasted. Basically, there are two application tools needed to develop the VLES of C Programming; Macromedia Dreamweaver MX and Macromedia Flash MX.

4.2.1 Macromedia Dreamweaver MX

Macromedia Dreamweaver MX is used to design the interfaces, writing the codes, adding the functions and make it online. Before designing the VLES, the developer need to build a web application, setting up a web server (Apache 2.0), setting up an Application server (PHP application server), creating a root folder for the application, defining a Dreamweaver site and lastly connecting to the database. After configuring all the settings, then the designing phase can be started.

4.2.2 Macromedia Flash MX

Flash is an authoring tool that allows to create anything from a simple animation to a complex interactive web application, such as an online store. The Flash applications media can be made rich by adding pictures, sound, and video. Flash includes many features that make it powerful but easy to use, such as drag-and-drop user interface components, built-in behaviors that add ActionScript to your document, and special effects that you can add to objects. The Flash file will be in a Flash document, a file that, when saved, has the file extension .fla. When the flash content are ready to be deployed, it will be published, creating a file with the extension *.swf.

4.3 Human Engineering

Good user interface is critical to the success of a particular system. An interface that is hard to understand will result in a high level of user errors. For a worst case, the users will refuse to use the software system, regardless of its functionalities and usages. Thus, this Virtual Learning Environment System has served to provide a set of Graphical User Interface (GUI) that is similar to other Intelligent system but in a more attractive and interactive way.

4.3.1   Target user groups

A Virtual Learning Environment Systems has been produced, as to show how a VLES is, and how it can be used to teach C Programming interactively. The user group for this system is intended for the students who want to learn C Programming. It is assumed that these groups of users possess minimal knowledge as how to use a VLES and online resources, but not having any expertise on C Programming course.

4.3.2   Virtual Learning Environment System Interface

The user interfaces for this VLES of C Programming have been kept on simple and structured manner. This is to let users have an impression that this system is easy to use (user-friendly), and can provide the information and knowledge that they need. In this section, there will be 3 different types of users explained; Student, Lecturer and the Administrator, based on their interfaces. There will be also a bit explanation for the other elements that has been included in the VLES; e-book, lecture notes and tutorials given.

4.3.2.1 Virtual Learning Environment Systems

To access Protected or Private courses, or to create courses on the VLES, user must first create an account. Once registered, the same login information can be used to access any course on the system. To create an VLES account, users can click on the *Registration* link at the top of the screen and fill out the registration form (Figure 4.1).

Figure 4.1: Registration Form

After the registration have been done successfully, user will be required to login to private and protected courses through the login screen shown below (figure 4.2). Login is optional for Public courses. User also can set the Auto-Login checkbox on the login screen if they wish to login automatically when you access VLES. But, beware that when auto-login is enabled, anyone using that particular computer will be able to access thr account. User should not use auto-login if they are using VLES on a public or shared workstation.
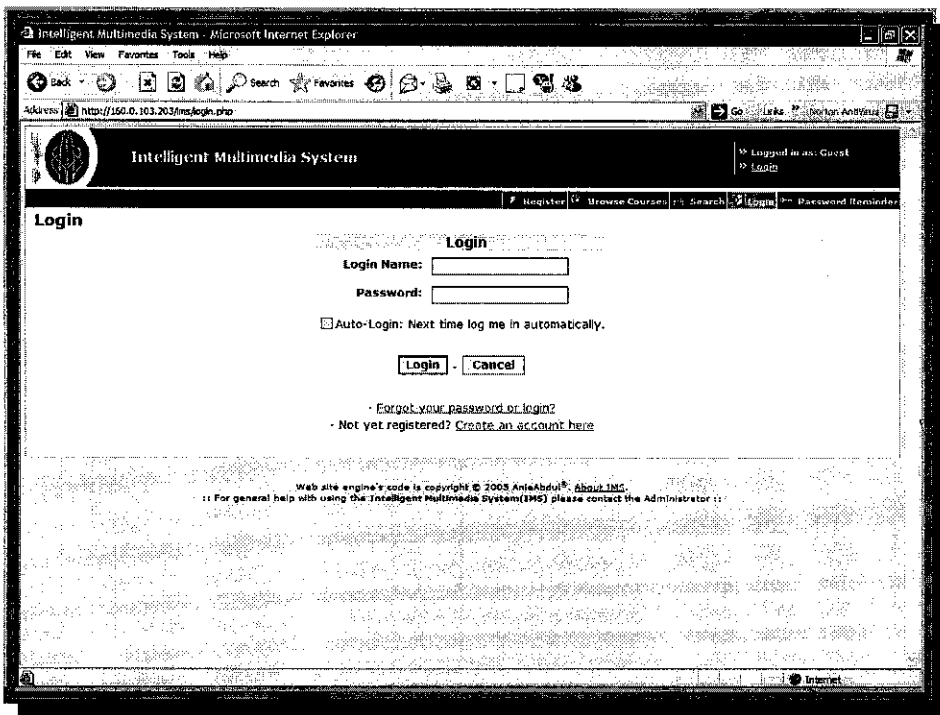
Figure 4.2: Login Page

4.3.2.1.1 Student User Interfaces

(a) My Courses

The ✿ **My Courses** page allows the user to manage the VLES courses in which they are enrolled. For each course, user can *Contact the Course Lecturer* or *Remove* the course from the list. A Shortcut to Forums, and a Shortcut to ACollab groups will also be available when these features are enabled.
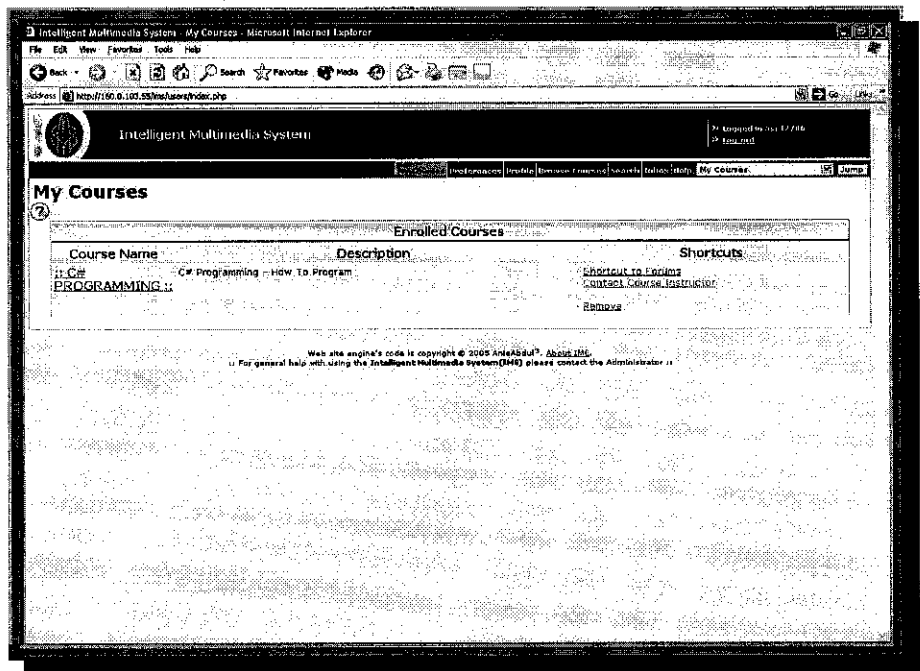
Figure 4.3: My Courses User Interface

The *Enrolled Courses* table lists all of the courses in which users are enrolled. Enrolling in a course adds a listing to the *Enrolled Courses* table. Being enrolled in a course gives the user access to the Users Online functionality so they can communicate with others in the course, adds the course listing to the Jump menu for quick access, and adds their name to the course mailing list so they will receive course email notifications.

Enrollment in public and protected courses is optional, but *enrollment in private courses is required.* In order to access a private course, enrollment must be approved by the lecturer. Private courses will be listed in the Enrolled Courses table as *Pending,* until the enrollment has been approved

41

(b) Preferences

A variety of preference settings are available which allow VLES users to control the appearance and layout of the student's environment. The *Preferences* page can be accessed from the Member Navigation bar, or from the Tools page.

Figure 4.4: Personal Preferences Form

- **Position Options**: alter the location of the side menu, sequence links, and table of contents

- **Display Options**: options to show or hide topic numbers, breadcrumbs, page heading links, help boxes, and pop-up help. Enable **Direct Jump** to view the same page when jumping between courses (content pages excluded).

- **Text & Icons**: set either icons only, text only, or icons and text to show for each of the member, course, and sequence navigation elements, and for content page headings

- **Menu Options**: the side menu consists of several boxes or modules - set which ones you would like to be shown or hidden from your menu. Arrange the most used modules at the top of the menu, and the less used modules at the bottom. Hide a menu module if you do not use it.

- **Themes**: you may set the look and feel of your VLES by choosing from the available themes. On some systems this feature may be disabled.

(c) Profile

The Profile page is similar in appearance to the registration form, and allows the students to update their personal information. The Profile is accessed through the Member Navigation bar. Students may change their password, email address, language preference, auto-login setting, and other personal data on this page.

(d) Inbox

VLES users can send private messages to those enrolled in the same course(s) as they are. Messages sent via Private Messaging are only available to the intended recipient. Private messages can be sent from a number of different places throughout VLES. In the Forums, for example, login names attached to postings are automatically linked to the Inbox.

Selecting a login name from the Users Online list will also open private messaging.

A link to Inbox is located in the Member Navigation bar and on the **Discussions** page. When a new message is received, the inbox icon will flash, or the associated text label will be highlighted, depending on the preference settings. Once a new message has been read, the icon, or text label, will return to its normal state.

(e) Course Features

In addition to the general Member Navigation functions, each course provides a variety of features that enhance the learning experience. Links to these areas are located in the *Course Navigation* bar.

Figure 4.5: Course Navigation Bar

❋ Home

The Home page created for each course is accessible from within a course through the Home link in the course navigation bar, in the Sitemap, at the start of the Breadcrumb links, or at the top of the Global or Local menus. Each course Home page contains a list of announcements posted by the course lecturer and/or lecturer's assistant. The Home page is always the first page displayed when logging into a course.

✳ Tools

The tools page provides a collection of utilities that can be used to search through a course's content, export content packages, take tests and manage test results, and monitor student's tracking information for a course (when enabled).

The Search tool allows student's to search through content and keywords found in courses on an VLES. It is available in the Member Navigation bar, through the Tools page, or it can be attached as a menu module with a preference setting so it is always visible.
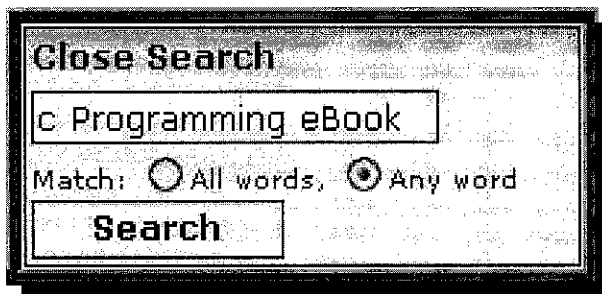
Figure 4.6: Search Tools

Search terms are generally single *words separated by spaces.* Select from the radio buttons below the search field to search for any, or all of the search terms. Do not use words like "and", "of", "the", or other such words. Select *where to look for results;* either in the course students are currently viewing, in all of the courses they are enrolled in, or in all available courses on the VLES. Then select *how to display the results* as a list of individual content pages, content pages grouped by course, or as course summaries with the number of pages found in that course.

Glossary items embedded in pages are identified with a superscript question mark [?] next to the word or phrase, or the mini glossary icon [▣]. For those with a browser that supports Javascript, positioning the mouse pointer over the glossary icon will open a small floating definition window. For those using a browser that does not support Javascript, or for those who cannot access dynamic content with their assistive technology, selecting the glossary icon directs them to a separate page with definitions displayed in an accessible format. The glossary can also be accessed through the *Tools* page, from the bottom of the Global menu, or through the Glossary Terms menu module if it is available.

The Export Content feature creates a "Content Package" that can be downloaded and viewed offline in the viewer included with each package. Lecturers can turn this feature on or off. If it is turned on, learners will find a link titled "Export Content" next to the title on top level content pages, or all content pages (depending on what the lecturer has set). Content can also be exported using the Export Content tool linked from the Tools page.
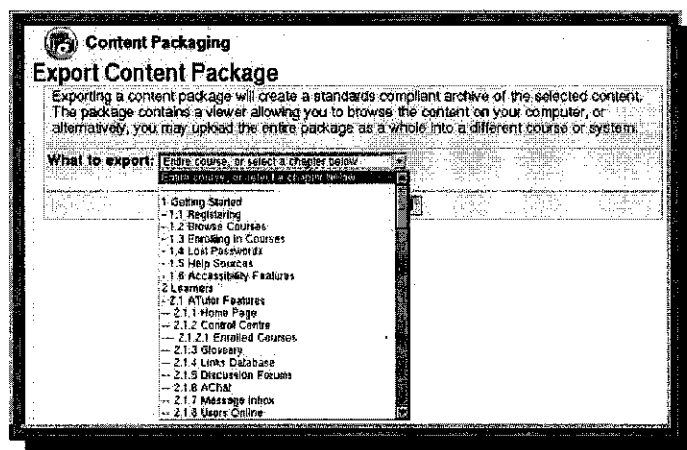

Figure 4.7: Export Contant Packaging Tool

46

The downloaded file can be unpacked with a common archiving application. *Unzip* the file into an empty directory then **click on** *"index.html"* to open the viewer. A list of topics appears in the left frame of the viewer, and the content itself appears on the right. Click on a topic to display it's corresponding content on the right.

The *My Tests & Surveys* table displays a list of tests and surveys for the current course. When available, the title becomes an active link to the test or survey itself. The *Completed Tests & Surveys* table displays a list of students' submissions. When a test has been marked by the lecturer, the *View* link beside its entry will become active. Selecting the link will display their results with possible feedback from the lecturer. All submissions, for all courses, are available in the Completed Tests table.
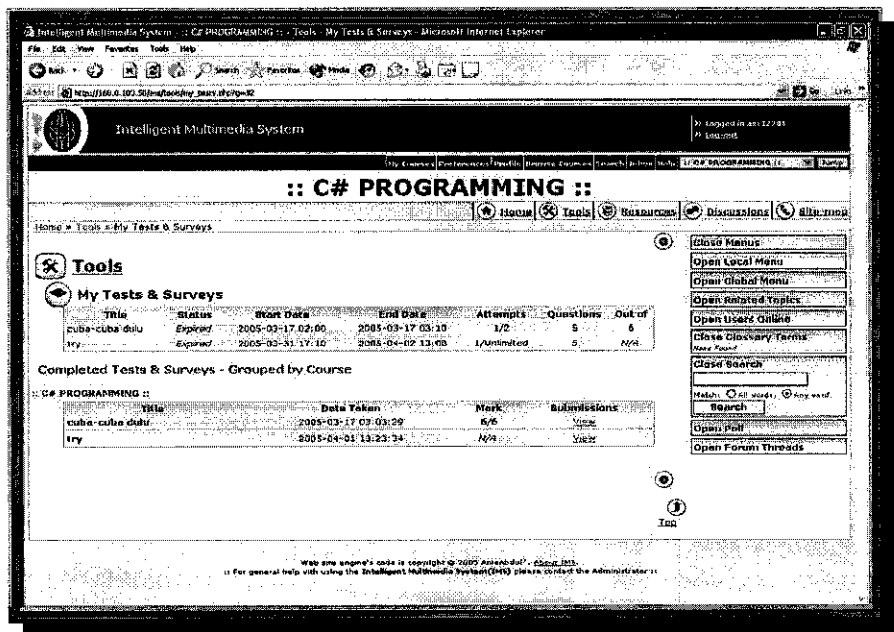

Figure 4.8: My Test & Servey Tool

My Tracker provides student with a summary of the navigation and learning tools they have used, as well as the click path they have taken through the content of a course. This information is also available to lecturers, who might use it to confirm that students have covered the content of a course thoroughly. My Tracker is only available if the VLES administrator has enabled it for a course, at the request of course lecturers.

Tracking Summary

On the opening screen of MyTracker appears a list of pages a learner has visited in a course, including the number times each page was accessed, as well as the cumulative time spent on each page. At the bottom of the summary the total number of pages visited in a course is displayed.

Immediately below the summary is a list of pages in the current course that have not yet been visited by a learner. Use this list to ensure that all pages in a course have been visited. Click on the page titles to go directly to the pages that have not yet been viewed.
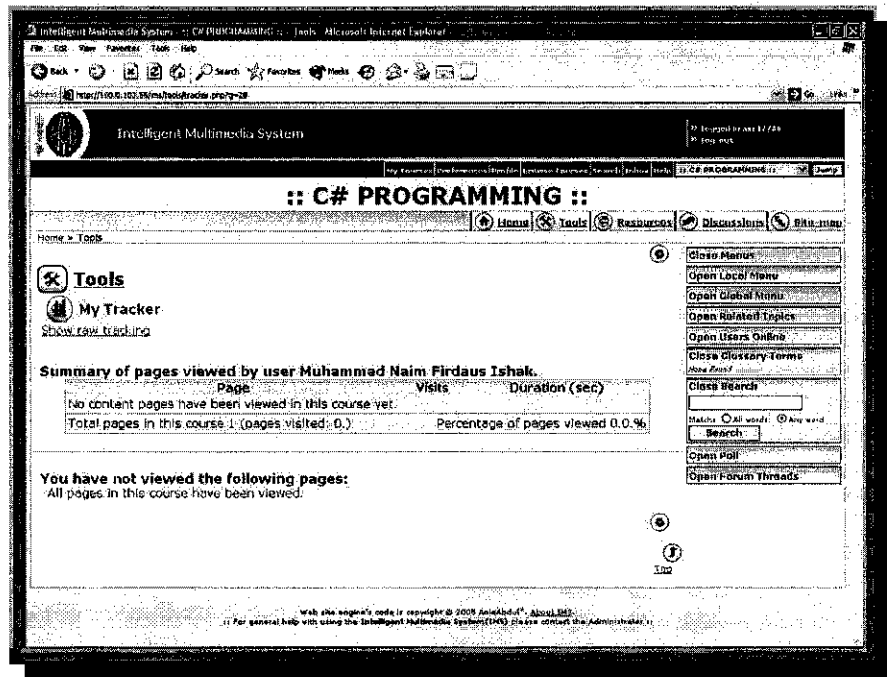
Figure 4.9: Tracking Summary Interface

Raw Tracking

Click on "show raw tracking" to view a breakdown of the tools that have been used and the path through the content of the course.

Navigation Strategies

The Navigation Strategies table provides a summary of the tools a student has used while moving through a course. Navigation tools used, such as the Global Menu, Sequence Links, or perhaps the sitemap, display tendencies students have for structuring content in different ways.

Navigation tendencies are also displayed for various learning tools. Learning tools used, such as MyTracker, Search, or perhaps the Glossary, display tendencies learners have for interacting with the content of a course.

Figure 4.10: Navigation Tendencies

Navigation Path

The Navigation Path lists the sequence of pages students has visited as they moved through a course, as well as the navigation tool they used to access a page, and the time spent on each page (i.e. Duration).

Access Method	Page Viewed	Duration	Date
			Page: 1
Home		N/A	Nov-18-00 3:27:50 am
Sequence	Consciousness	10.49	Feb-8-03 7:45:04 pm
Global Menu	Goals	03.48	Feb-8-03 7:55:52 pm
Global Menu	Suggested Discussions	00.12	Feb-8-03 7:59:40 pm
Global Menu	Learning to Concentrate	00.03	Feb-8-03 7:59:52 pm
Global Menu	Relaxation	00.03	Feb-8-03 7:59:55 pm
Global Menu	The Two Minute Mind	01.38	Feb-8-03 7:59:58 pm
Global Menu	Focusing	01.47	Feb-8-03 8:01:36 pm
Local Menu	Visualizing	00.41	Feb-8-03 8:03:23 pm
Local Menu	Calculation and Concentration	00.07	Feb-8-03 8:04:04 pm
Sequence	Emotion and Learning	00.40	Feb-8-03 8:04:11 pm
Table of contents	Suggested Discussions	00.06	Feb-8-03 8:04:51 pm
Headings	Targets: Outline	00.53	Feb-8-03 8:04:57 pm
Home		00.17	Feb-8-03 8:05:50 pm
Print Compiler		00.12	Feb-8-03 8:06:07 pm
Local Menu	Targets: Outline	00.02	Feb-8-03 8:06:19 pm

Session above started on February 8, 2003 7:45 pm and ended on February 8, 2003 8:07 pm (Duration 22 min 14 sec)

Figure 4.11: Navigation Path

✸ **Resources**

The Links Database can be accessed through the Resources tab in the Course Navigation bar, or through a link icon, which student may find embedded throughout the content of a course. Clicking on the icon will take student to the Links Database for this course.

The Links Database includes links to Web sites related to the content of the course currently being viewed. Both Lecturers and Students can *submit links* to a course's link database. To submit a link, users need to choose the category into which the link best fits (or suggest a new one), then, select *"Suggest a new link"* In the *"Add a New Resource"* form that appears, enter the title of the site, use meaningful *keywords* in the description to help others find the link when searching. When users are satisfied with the information, click "Submit Resource" to post the link for Lecturer approval.

51

Figure 4.12: Add a New Resources Form

The TILE Learning Object Repository is fully searchable and delivers accessible and personalized learning objects. This environment supports the delivery of web-based media including hypertext, various xml formats, video, animation, simulations, Java and others. VLES's TILE search interface allows students to access this repository of content packages and *download instructional materials*. They may view content online through the repository's own viewer, or download packages and read them offline with the accompanying viewer, or in another VLES/SCORM conformant content package reader.
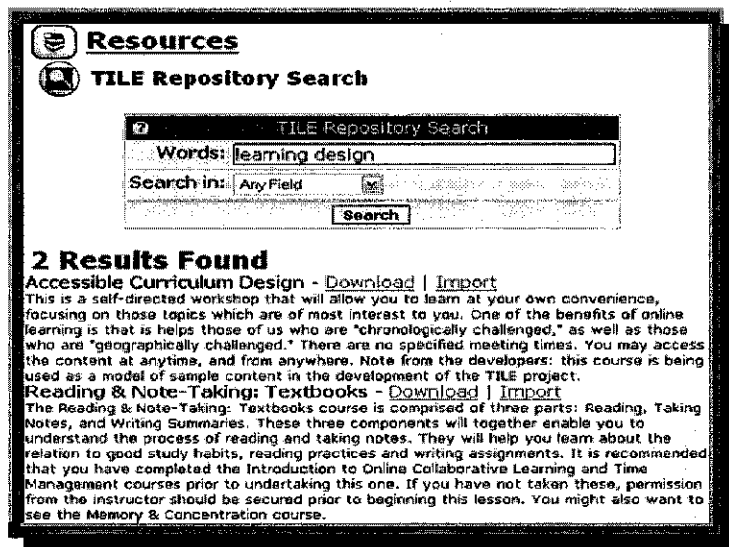
Figure 4.13: TILE Repository Search

✸ Discussion

The Discussions page contains a list of communication related utilities. Students may interact with course members in a variety of ways, privately or in a group setting, through the *Forums, AChat, and Users Onlin.*

To access the forums, select the *Discussions* link in the Course Navigation bar, then select the *Forums* link. Select the name of the forum student wish to enter.
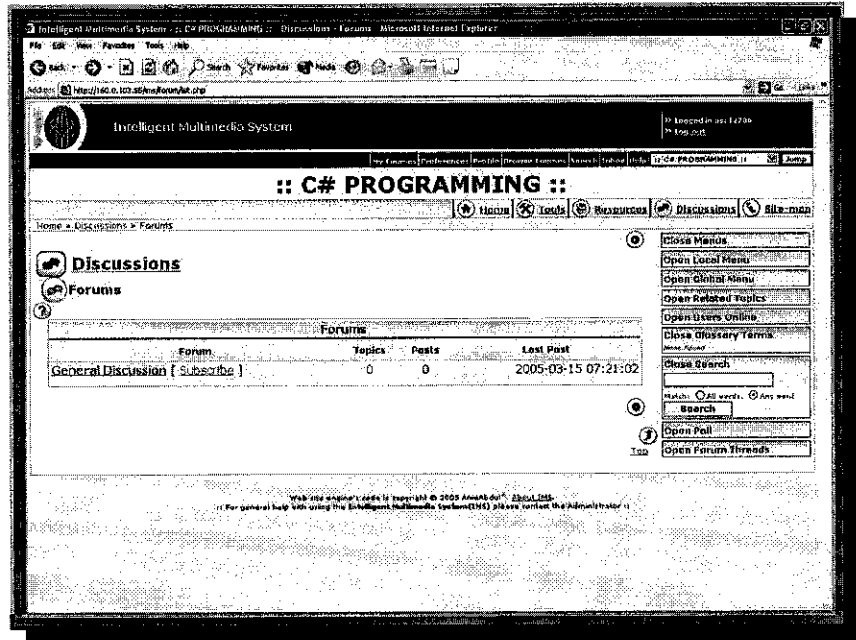
Figure 4.14: Discussions Interface

Message threads are displayed 10 threads at a time. When more than 10 threads exist, *Page menu* links appear at the top and bottom of the thread so that users are able to flip through all the pages in the thread list. The most recent message is posted to the end of the thread, on the final page if multiple pages exist.

Where new messages have been posted that users have not yet read, the word **New!** appears beside the thread. If a thread has more than 10 replies, the word "**Hot!**" appears beside it. Users may respond privately to others by selecting their login name either from the *Started By* column of the thread list, or at the top of each message. To add a *reply to a thread*, use the Add Post form that appears below the thread. To post a reply to a specific message within a thread, select *Reply* next to the message.

54

AChat is an access compliant chat that allows assistive technology users to participate in synchronous online discussions. On the opening screen of the chat, select *Enter the Chat* to be moved into the chat room. Student may wish to open AChat in a new window alongside VLES, as they navigate through a course. Most browsers allow to right click on the link then choose to open it in another window. Click on *Edit Preferences* or *Help* within AChat for more about configuring and using the chat.
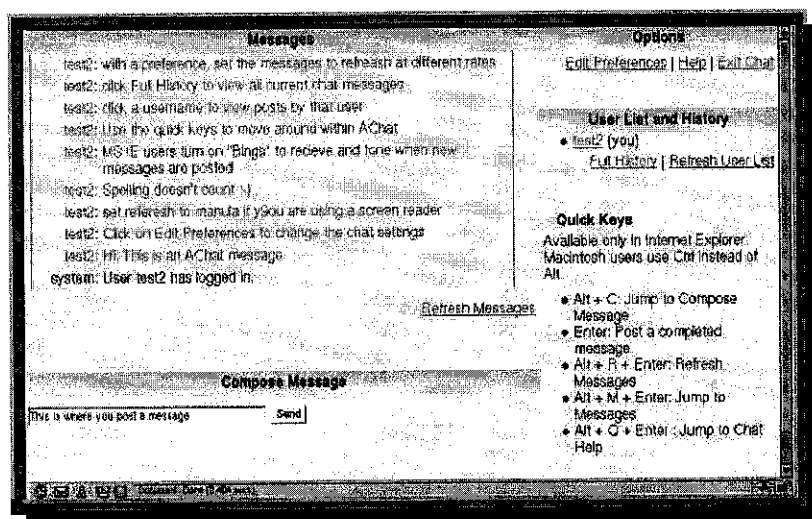

Figure 4.15: AChat Interface

The **Users Online Menu** displays a list of those currently logged into a course.

Figure 4.16: User Online Box

Clicking on a user's name will open the VLES **Inbox** through which they can send the selected user messages. The Users Online list can be found on the *Discussions page*, or attached to the Menu.

Student may wish to send a message asking other classmates to join them in the Chat Room, where they can communicate with them in real time. *Click on user own name* in the Users Online menu to see how it works.

4.3.2.1.2   Lecturer User Interfaces

Lecturer are also same as a learner, as you go about creating your first course. Lecturer should *read through the Students section* of this course first. It applies as a novice VLES lecturer, as much as it applies to the students who will take the course. Lecturer should know how the students will be using the course, and know how to configure VLES to work best for them and their students.

Get an Lecturer Account

To create courses, the lecture must have *VLES System Account* upgraded to *lecturer status*. Administrators have the option of making lecturer requests available or not. They may also automate approval of lecturer requests, so new lecturer may begin creating courses immediately after making their request. When lecturer requests is enabled, a link will appear next the a user's "Status" on their **Profile** screen. Select the link, if it is there, to *Request an Lecturer Account.* Enter a brief description of the proposed course and use the *Request*

56

Lecturer Account button. Once the Administrator has done so, the respective lecturer will receive an email notifying them of the approval or rejection. If this feature is not enabled, contact the System Administrator directly to have an lecturer account created.

Lecturer have a variety of tools available to them for managing their online courses and instructional content. These include a course Content Editor, a Web resources database, forum manager, course statistics, an enrollment manager, and other tools. This section describes how these tools are used.

✸ File Manager

VLES's *File Manager* enables lecturers to upload files to a course directory, and embed or create links to those files within course content. Lecturers can open the File Manager from the Tools page, or open it into a new window from the Content Editor or the Test Question editors, to manage files while editing content.
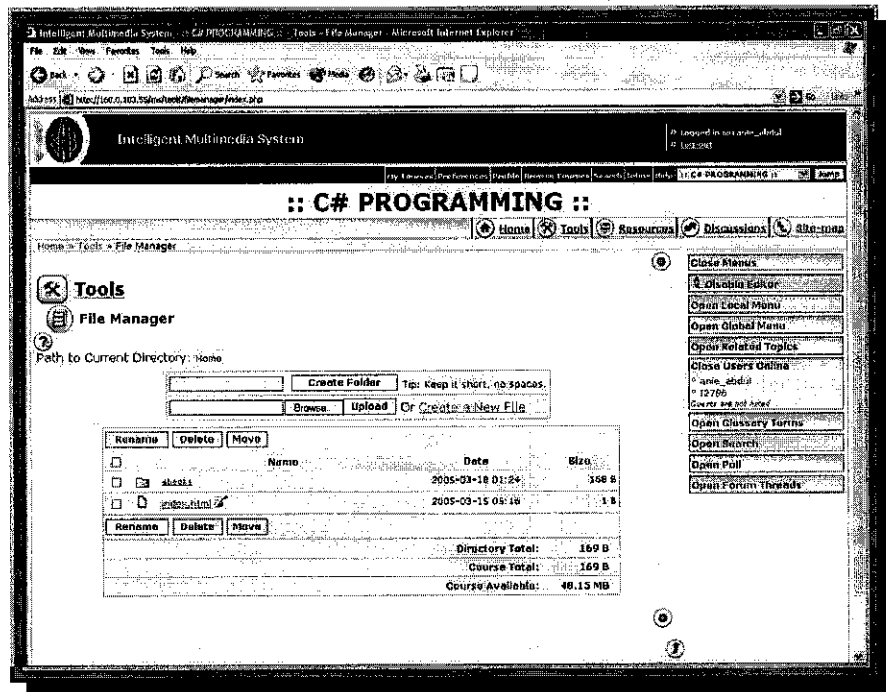
Figure 4.17: File Manager Tools

Create Folders

To organize the course files, create Folders in which to sort them. While they are not required to create folders, instead uploading all files into the root directory of the file manager, files will be easier to manage if a folder structure is planned during the early stages of creating a course.

Uploading Files

Move into the folder where the uploaded file should be located. Then select the *Browse* button to locate a specific file on the local hard drive, followed by the *Upload* button to save the file to the course directory.

58

Creating and Editing Files

Select the Create a New File link to open the file editor. Files must be plain text, or HTML formatted text. In most cases, lecturer will want to enter their text or HTML into a content page using the Content Editor rather than the File Editor, though in some cases opening content outside of VLES may be desirable. In such cases, the File Editor can be used.

Rename, Delete, and Move

The *Rename and Delete* functions in the File Manager do not require much explanation, except perhaps that they can select, then delete, multiple files, while they may only rename files one at a time.
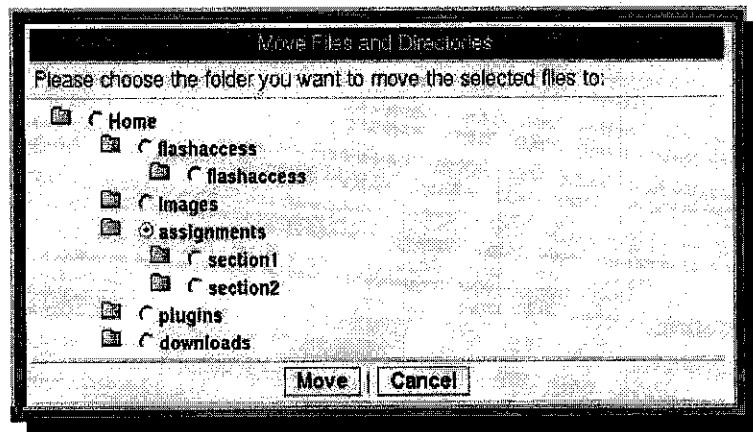

Figure 4.18: Rename, Delete, and Move Files form

❋ **Enrollment Manager**

The Enrollment Manager is used to manage people in a course. Lecturer may manually create a course list, adding students by entering names and email addresses into the *Create Course List* tool. They might choose to

Import or Export a Course List in a comma separated values (CSV) format, or *Create Groups* for selectively assigning test or quizzes to different students. Students who have completed the course can be added to the *Alumi list*, so they can access the course and participate in forums after finishing the course. *Roles & Privileges* can be assigned to enrolled course members to create assistants or multiple lecturers.
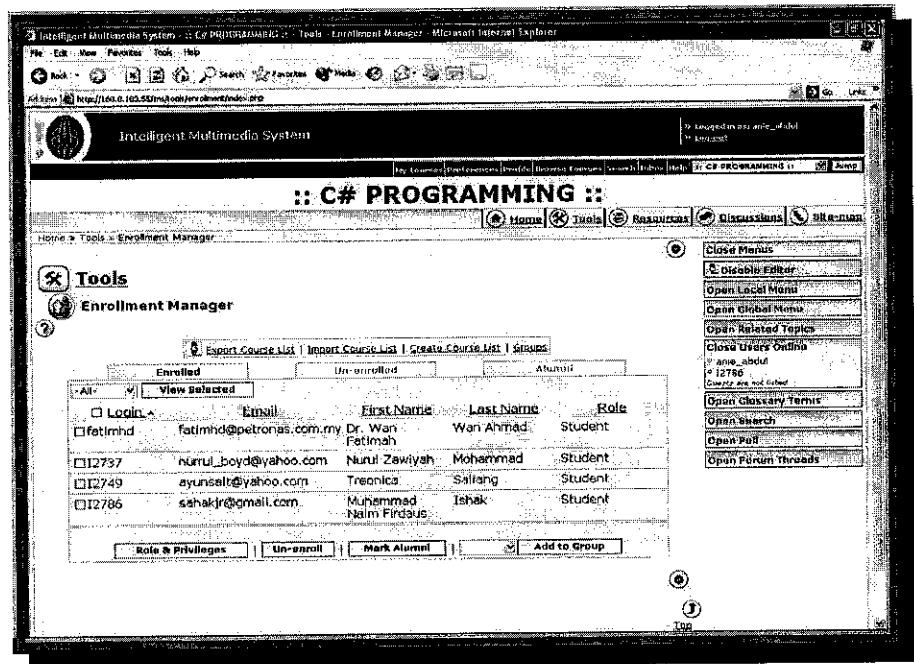

Figure 4.19: Enrollment Manager Tools

❋ Test & Servey Manager

The Test & Survey Manager allows lecturers to create online quizzes and surveys. In the first screen shot below there are two assessments *On Going*, indicated in the Status column. An assessment might also be *Pending*, not yet made available to students, or be *Expired*, or no longer available to students. Tests or Surveys become expired automatically when the End Date is reached.

Use the *Question Database* link in the Editor Tool Bar to create questions. Then click *Create Test/Survey* to setup the initial properties for an assessment, after which questions can be added to it from the Questions Database. Use the Questions links in the Questions column to *add or edit questions* in an assessment. Use the Preview link to see what the assessment would look like when presented to students.

After students have taken a quiz or test use the *Submissions* link in the Results column to view or mark submitted tests. Use the *Statistics* link to view test statistics.
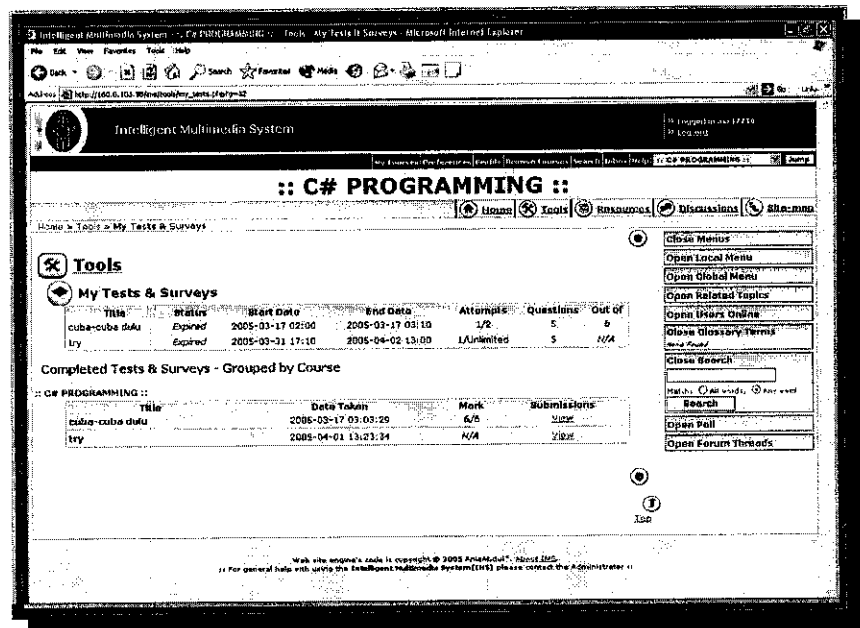

Figure 4.20: Test & Servey Tools

Linking to Tests

When a test or survey becomes available to students, the Take Test link becomes active on their My Tests page, found under Student Tools on the Tools Page. The link is also displayed at the top of the home/announcements page, so students know about the test when they login. You may wish to link to tests from other locations throughout your content, to make them easier to find or to associate tests with a particular topic within your course. Once your test is established, and the test window is open, go to My Tests, and right click on the Take Test link next to a specific test. From the menu that appears select "*copy link location*", then paste that link into your content at the point where students should take the test.

Viewing and Marking Submissions

Results must be generated by the lecturer when a test has open ended questions in it. The number of tests that require marking is shown in the main Test & Survey Manager table beside the *Submissions*. Clicking this link will display the Completed Tests table, which lists all submitted tests. Selecting *View & Mark Test* will allow an lecturer to view the test submission, and submit marks for each question. *Multiple choice* and *true or false* questions will be *marked automatically* when you open a students results. Review them to ensure they are marked correctly. Lecturers must manually enter a mark for open ended questions by enter the mark in the appropriate text field next to each question. When you have completed reviewing multiple choice, and true or false results, and entering marks for the open ended questions, submit the results to update that student's scores. If Release Result is set to display marks Once quiz has been submitted, students will be able to view their scores and

feedback in their "My Tests" section using the newly active "View Results" link.

Statistics

By selecting the "Statistics" link for a test in the Test & Survey Manager, two types of result statistics can be viewed.

> ➢ Question Statistics - Overall results for each individual question can be viewed here. Beside each question, the number of blank answers, and the number/percentage for each response is given. A red asterisk marks a correct response.
> ➢ Submission Statistics - Marks for each student's submission can be viewed here. Also displays overall question and submission averages.

❋ **Course Tracker**

The *Course Tracker* can be used to show course usage and content coverage statistics. Tracking data can be downloaded for offline viewing or study. The course tracker must be turned on by the VLES system administrator.

- *Show page statistics*
  Review content and feature usage patterns as a source of information from which to improve the effectiveness of the course.

- *Show member statistics*

  Understand how individual students are using the course, and provide them with content that will match their learning preferences. Also see which pages individual students have viewed, and the time spent on those pages, to track students' content coverage.

- *Download tracking data in a CSV file.*

  Import the raw tracking data into a spreadsheet application, or database table, to do the study of usage patterns. Or, archive CSV files after each session to create a course usage history.

- *Reset tracker for new session.*

  Empty the tracking data after each course session. You might choose to first archive data from previous sessions using the CSV tool above.

❉ Course Default Preferences

To set the course preferences to the configuration the lecturer are currently viewing, click on *Course Default Preferences* in the Lecturer Tools, then click *Save current settings as course default preferences*. For more about configuring VLES preferences, see Preferences in the Learners section of this course. Lecturers set course preferences the same way they would set their own personal preferences, and the same way students would set their's.

✶ Course Copyright

Enter a custom copyright notice to display at the bottom of all the content pages. Simply enter it into the text area, and select the 'Save Changes' button. By default, no copyright is displayed.
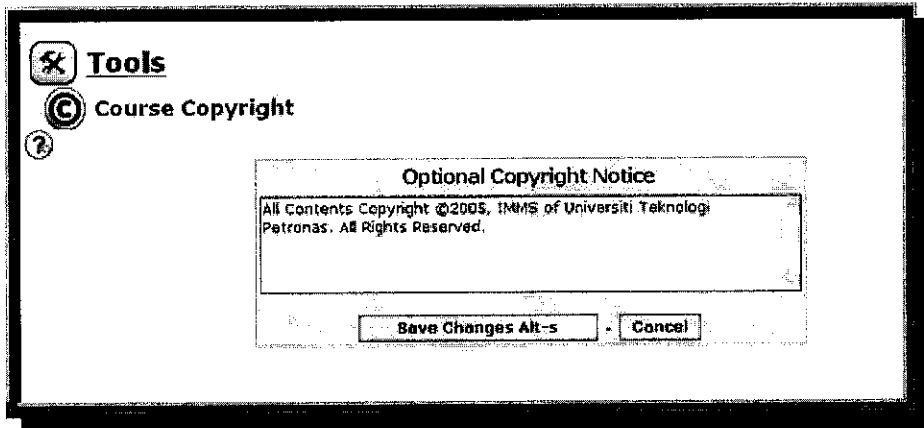

Figure 4.21: Course Copyright

✶ Links Database

The Links Database is a place where lecturers can gather links to web resources used to support the course content. Links can be sorted into categories, perhaps categories matching the main topics in a course.
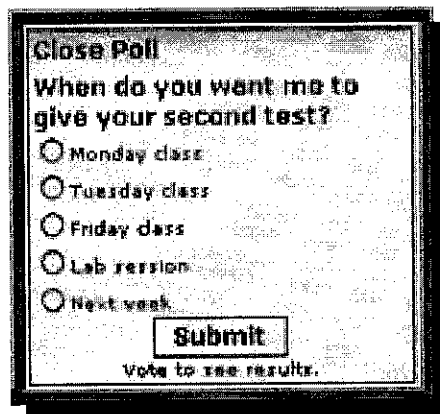
✶ Managing Forums

Lecturers can create forums to allow communication between course members. *Course Forums* are private to a specific course. Lecturers can create as many Course Forums as they need for a course. *Shared Forums* are common among a number of courses on an VLES system, so course

members can communicate with members of other courses. A Shared Forum may be available to all courses, or to a smaller number of courses. In the screen shot below you will see one course forum, and two shared forums.

✳ Polls

Polls can be used to obtain student input/perspective on various topics. A poll is essentially a single multiple choice survey question. Students can each vote for their choice.

Figure 4.22: Polls window

A lecturer may create and edit polls by going to the *Discussions* area and selecting the link for *Polls*. The most recent poll can be displayed in the side menu by going to *Preferences*, and adding it under the Menus table.
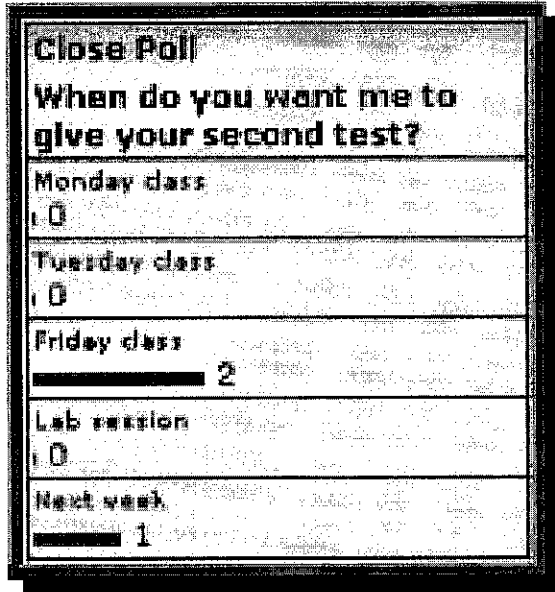
Figure 4.23: Polls Result window

4.3.3 C Programming e-Book

This e-Book is specially designed for the Virtual Learning Environment System users that enrolled with the C Programming course. Users can view the e-Book by browsing the Course materials folder in the VLES. The e-Book is designed interactively using a combination of Macromedia DreamweaverMX and SwishMax application in order to develop an eye catching e-Book. It is important to design a simple and yet attractive application so that the users feel interested to use it. The e-Book covers all the topics learn in C Programming. Users can read the information by just clicking the desired topics displayed at the left side of the e-Book. Navigation buttons are provided in every page of the e-Book such as home, back, and etc. The example figures of the C Programming e-Book can be found at the appendices page.

4.3.4 C Programming e-Tutorial

The C Programming e-Tutorial is also one of the applications included in the Course Material folder. It is designed for each topic taught in the C Programming e-Book. It is provided for the students to check their level of understanding based on each topic learnt. Each question comprises 4 choices of answer and students need to choose one of them before they can proceed to the next question. If the answer chosen is correct, then only they can go to the next question or else they stuck on that question and need to go back and start the exercise from the beginning once more. Every answer chosen by the users will display a respond page which includes together the correct answer (if the answer chosen is wrong), short notes regarding to the question and their current marks. The print screens of the e-Tutorial application can be seen at the appendices page.

CHAPTER 5

CONCLUSION

5   CONCLUSION

If a close watch and detailed review are made on the existing educational systems and the population of students, quite a wide ranges and variations among students can be detected. With the recent technological advancements in the fields of computer technology, artificial intelligence technology, education and instructional technology, web-based instruction has been becoming somewhat common place in what would be considered traditional higher educational settings. One of the weaknesses on the traditional teaching and learning system is that it offers no facility in tracking student's progress or keeping courseware up-to-date. It is not enough just to supply materials to the learners. It has to monitor whether they have accessed the pertinent information, or whether they need to be chassed at times. It could be monitored and recorded.

VLES provides the opportunity for interaction from the learners, thus allowing them to tailor learning experiences to meet their specific needs or abilities. It supports an active and dynamic learning environment and offers the possibility of providing a stimulating learning environment to engage learners in meaningful learning reflection, application, and interaction. The most important are handling of multimedia documents, easy to use Graphic User Interface, On-line Help, easy to integrate multimedia documents, easy to manage course material, online discussion. The online discussions and email provide opportunities for all the learners to participate in the discussion of key issues as well as contributing to flexible learning approach. The other features are online quiz template which, helps putting quizzes online, learners and teachers registration interface.

REFERENCES

[1]   Kearsley, G., "The World Wide Web: Global Access to Education", Educational Technology Review 5, 1996.

[2]   Maddux, C.D., "The World Wide Web: Some Simple Solution to Common Design Problems", Educational Technology, 1998.

[3]   Berge, Z.L, "Guiding Principles in Web-Based Instructional Design of Gravity-type Vertical Seawalls", Engineering Applications of Artificial Intelligence, 1992.

[4]   Chau, K.W., Ng, V., "An Expert System for the Design of Gravity Type Vertical Seawalls. Engineering Applications of Artificial Intelligence", 1992.

[5]   Chau, K.W., Albermani, F., "Expert System Application on Preliminary Design of Liquid Retaining Structures, Expert Systems with Applications", 2002.

[6]   Chau, K.W., Chen, W., "An Example of Expert Systems on Numerical Modeling Systems in Coastal Processes", Advances in Engineering Software, 2001.

[7]   Chau, K.W., Ng, V., "A Knowledge-Based Expert Systems for Design of Thrust Blocks for Water Pipelines in Hong Kong", Journal of Water Supply Research and Technology – Aqua, 1996.

[8]   Chau, K.W., Yang, W.W., "Development of An Integrated Expert System for Fluvial Hydrodynamics", Advances in Engineering Software, 1993.

[9]   Reggie Kwan, Weijia Jia, Jimmy Chan, Anthony Fong, Ronny Cheung, "Web-based Learning: Men and Machines", The First International Conference on Web-based Learning in China, 2002.

[10]   BRITAIN, S., LIBER, O., 1999. *"A Framework for Pedagogical Evaluation of Virtual Learning Environments"*. Report 41, JTAP. http://www.jisc.ac.uk/uploaded_documents/jtap-041.doc

[11]   MUSGROVE, A., 2001. Using WebCT3 to create web-based learning for multiple learning styles. *"Proceedings from the Society for Information Technology & Teacher International Conference"*. pp. 53–58.

[12]   MORSS, D.A., FLEMING, P.A., 1998. WebCT in the classroom: A student view. North American Web Developers conference, October. http://www.unb.ca/naweb/98/proceedings/morss/

[13]   GOLDBERG, M.W., 1997. *"Students' Use of a Web-based Resource in Computer Science"*. University of British Columbia. http://www.cc.ubc.ca/campus-computing/may-aug97/cs-study.html

[14]   FOLLOWS, S.B., 1999. Virtual learning environments. *"The Journal, November"*. http://www.thejournal.com/magazine/vault/A2374.cfm

[15]   JACOBSEN, D.M., KREMER, R., 2000. Online testing and grading using WebCT in computer science. WebNet 2000: World conference on the WWW and the Internet, San Antonio, Texas, USA, 30 October–4 November. http://www.ucalgary.ca/~dmjacobs/seng/webnet2000_paper.html

[16]   ROBERTSON, T., KLOTZ, J., 2001. Confronting design problems in developing on-line courses in higher education. Mid-South Educational Research Association (MSERA) Conference.

[17]   CURTIS, D.D., LAWSON, M.J., 2001. Exploring collaborative online learning. *JALN,* 5 (1).
http://www.aln.org/publications/jaln/v5n1/pdf/v5n1_curtis.pdf

[18]   GOLDBERG, M.W., 1997. *Students' Use of a Web-based Resource in Computer Science.* University of British Columbia.
http://www.cc.ubc.ca/campus-computing/may-aug97/cs-study.html

APPENDICES

APPENDIX A   :   FINAL YEAR PROJECT GANTT CHART

APPENDIX B   :   C PROGRAMMING E-BOOK

APPENDIX C   :   C PROGRAMMING E-TUTORIAL

APPENDIX A

FINAL YEAR PROJECT GANTT CHART

APPENDIX A : GANTT CHART

No	Detail / Week	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	Topic Confirmation	▒													
	Requirement Definition														
	• Introduction	▒													
	• Objective	▒													
	• Literature Review / references	▒	▒												
2	System and Software Design		▒	▒											
	• Project Planning		▒	▒											
	• Choose Tools		▒	▒											
	• Schematic Diagram		▒	▒											
	• Develop Expert System Rules		▒	▒											
3	Submission of Preliminary Report			■											
4	Develop Prototype				▒	▒	▒	▒	▒						
	• Convert Rules				▒	▒	▒	▒	▒						
	• Design System Interface				▒	▒	▒	▒	▒						
	• Review with Supervisor				▒	▒	▒	▒	▒						
5	Submission of Progress Report									■					
6	Enhance Prototype									▒	▒	▒	▒	▒	
	• Pre EDX											▒			
7	Submission of Dissertation Final												■		
8	Oral Presentation													■	
9	Submission of Project Dissertation														■

▒ Development phases   ■ Dateline for report and presentations
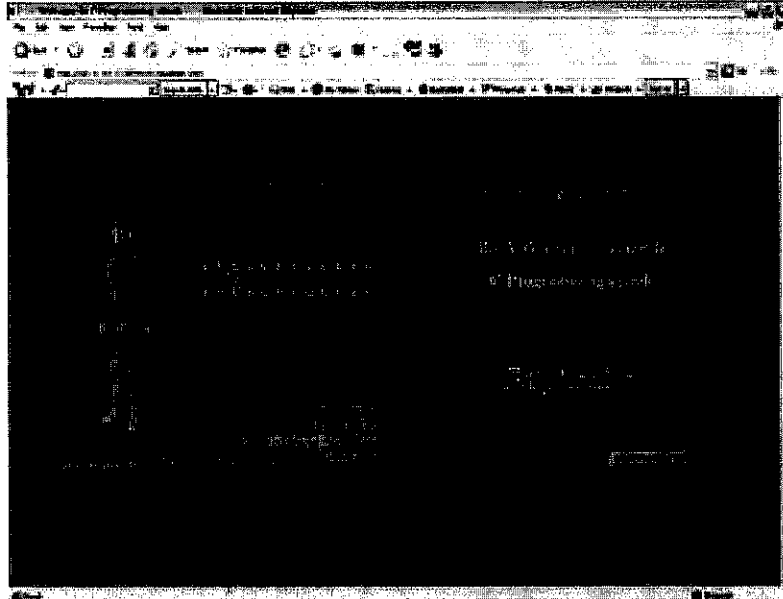
APPENDIX B

C PROGRAMMING E-BOOK

Figure B.1: The C Programming e-Book Main Page.

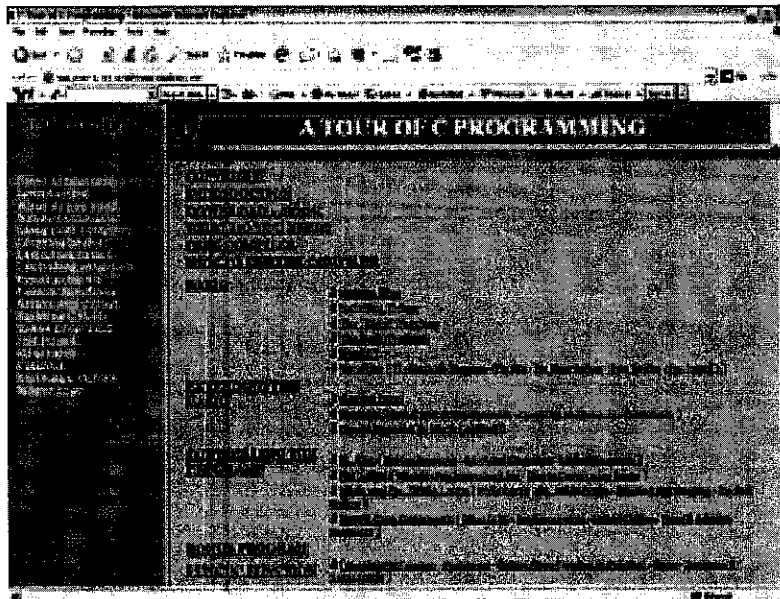It does have an animated Introduction page represented before the users were brought to the Main page.

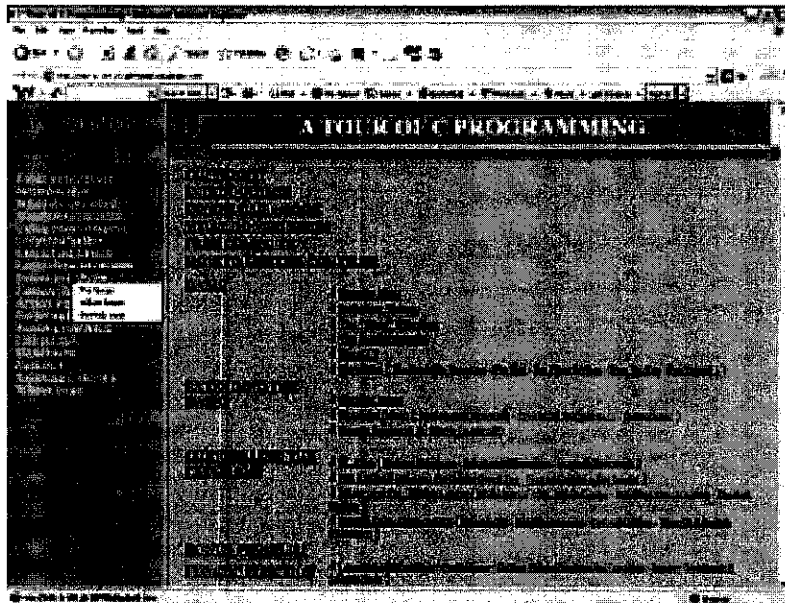**Figure B.2**: The C Programming e-Book Menu Page

75

Figure B.3: The C Programming e-Book Menu Page.

When the user point on the certain topics, the sub topics will be popped up
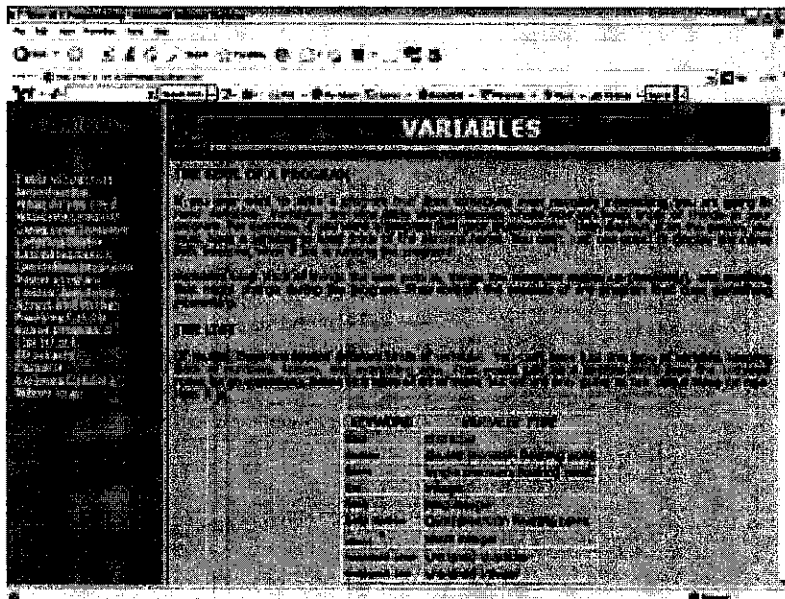
as the figure shown above (left side)

Figure B.4: The example of the content in the topics chosen by the user

APPENDIX C

C PROGRAMMING E-TUTORIAL
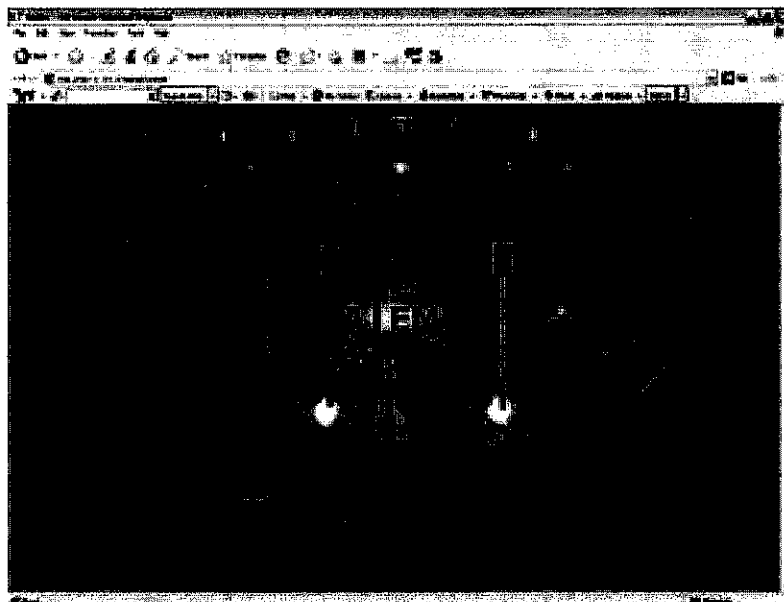
Figure C.1: The C Programming e-Tutorial Introduction page

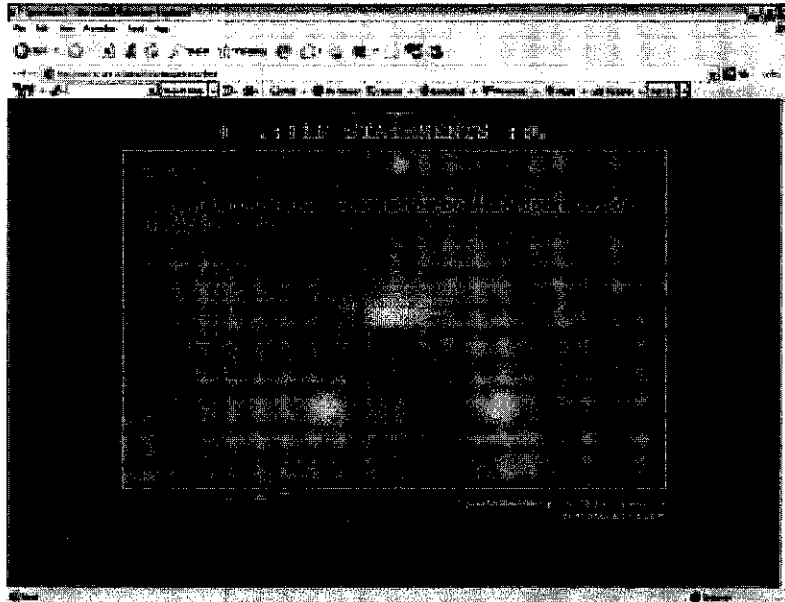**Figure C.2**: The C Programming e-Tutorial Main page

Figure C.3: The example of the question included in the e-Tutorial.
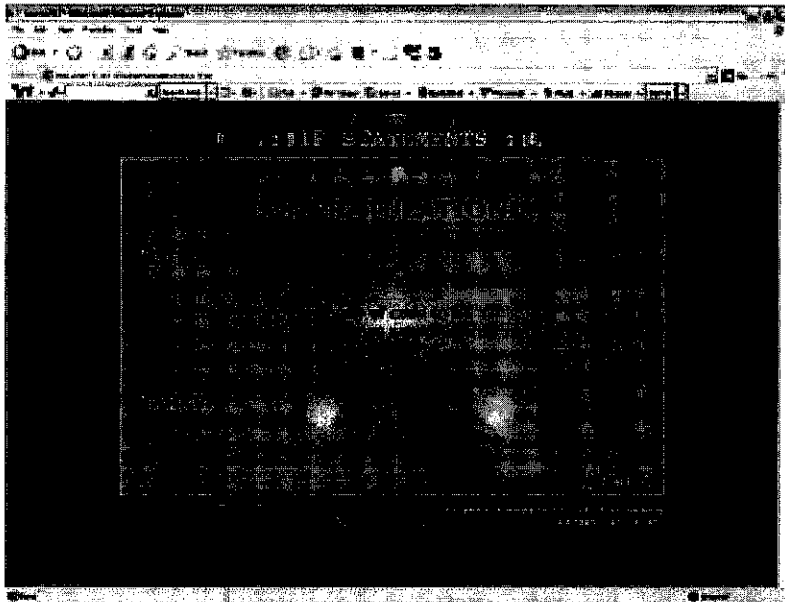Each question will provide 4 choices of answer.

Figure C.4: If the answer chosen by the user is correct, this page will appear.
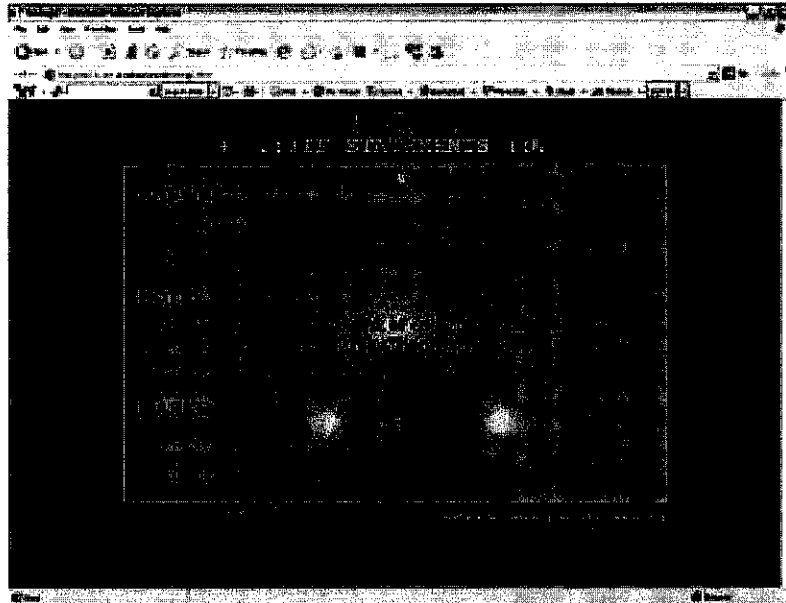The page consists of some notes regarding to the question and the current marks.

Figure C.5: If the answer chosen by the user is wrong, this page will appear. The page consists of the correct answer, some notes regarding to the question and the current marks.