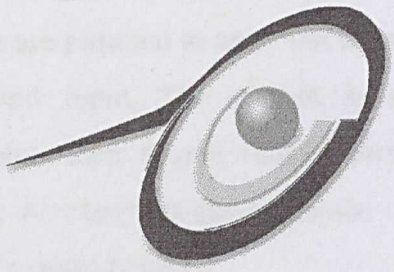
Faculty of Computer Science
and Information Technology



OTMS

Online Title Management System

Prepared by:

Wong Sze Khai

WET990039

Session 2001/2002

Under supervision of:
Puan Miss Laiha Mat Kiah

Moderate by:
Encik Rosli Salleh

Abstract

All undergraduates of the Faculty of Computer Science and Information Technology, University of Malaya are required to carry out a final year project in computer or information technology related topic. The project is a compulsory subject for undergraduates in order to complete their degree studies during their three years in the university. This final year project attaches two semesters and brings out nine credit hours out of the total hundred and twenty credit hours.

My thesis was named Online Title Management System (**OTMS**). I was doing the project under the supervision of Puan Miss Laiha Mat Kiah and moderated by Encik Rosli Salleh.

The main objective of the project is to implement a computerize system for managing the thesis titles for WXES/T 3181: Projek Ilmiah I. The main target users for the system are lecturers and undergraduates. This system will be keep track by the administrators assigned by FSKTM.

Online Title Management System is divided into three different modules, which are the Administrator module, Lecturer module and Student/visitor module. Generally, the Administrator module will be responsible for security control, which is keeping track of the user accounts and performing thesis title management. In the meantime, the Lecturer module allows users to access the database and performing thesis title management (compare, add, delete, update and search). Last but not least, the Student/visitor module lets the users to search and browse for related information.

This project is very important to all the undergraduates because it gives us the opportunity to apply whatever knowledge we have learnt during the pass two years. Beside that, we can also gain knowledge from the literature review done and produce a complete and formal report.

Acknowledgement

First of all, I would like to thank the Faculty of Computer Science and Information Technology for offering this opportunity to carry out a small-scale final year project. I would also like to extend my sincere appreciation to my supervisor, Puan Miss Laiha Mat Kiah. Her guidance, valuable advises and instructions, opinions and thoughtful contributions have helped me a lot throughout the entire project.

Beside that, I would also like to take this opportunity to express my gratitude to Encik Rosli Salleh for sparing his precious time to be my moderator. For special mention, I would like to thank the administrator, Puan S.N. Koh for providing me with some related information about the previous year's thesis titles.

Then, I would like to thank the staffs in FSKTM upon their kind cooperation and help. They have provided me with sufficient information and some sample forms (such as Borang Pemilihan Tajuk Projek WXES/T 3181) to carry out the project.

Last but not least, I would like to express my gratitude to all my friends and course mates who gave me some useful opinions and feedback. They also supplied me with some of the materials such as references books, CDs, etc. to accomplish my project.

Table of Contents

Chapter 3: System Analysis

Chapter 4: System Design

Chapter 5: System Implementation

Chapter 6: System Testing

List of Figures

List of Tables

Chapter 1:

Introduction

Chapter 1: Introduction

1.1 Project Overview

Nowadays, a lot of online information management system is available throughout the World Wide Web (WWW). Actually, the system is incorporated into many Web sites such as Yahoo, MSN, etc. and we are using it daily without realizing their existence.

Online Title Management System **(OTMS)** is designed to manage all the thesis titles for WXES/T 3181: Projek Ilmiah I in the Faculty of Computer Science and Information Technology, University of Malaya. This online information management system can help lecturers from the faculty to manage their titles for that particular subject via Internet. Any authorized lecturers can logon to the system and manage their titles properly, at anywhere and anytime they want. The main purpose of **OTMS** is to avoid titles offered by any lecturer to be exactly the same with each other. Beside that, all undergraduates who take this particular subject can view a complete list of titles offered to them by the lecturers.

OTMS also includes a search engine to search through the system for any specific thesis title. Browsing and searching for the titles can be made by all the undergraduates or any visitors without login. However, for security purposes, all the lecturers will need to logon to the system before they can make any changes to their own titles. **OTMS** is designed to be a very user friendly and easy-to-use system for adding, deleting, modifying and searching for titles. Anyway, what good is a system if it is very difficult to use and cannot make our works more efficient, precise and faster.

As usual, **OTMS** includes an administrator role to maintain the integrity of the system. Administrators can register the lecturers to become authorized users of the system. They can also make modification to any of the contents in the Web site and all the thesis titles stored in the database. Administrators will have the most powerful rights to access the system, follow by the individual lecturer. All undergraduates and visitors will have no rights to modify the contents of the Web site or the titles stored in the database. They can only view a list of the titles for their selection.

OTMS will emphasize that all the data and information can be retrieved from the Web based database easily to be display directly to the users. On the other hand, all the thesis titles and modification made to it (either by the lecturers or administrators) will be captured and saved into the database correctly, efficiently and immediately. Thus, this system is a solution to the current system, which is keeping the data and information manually in Microsoft Excel Worksheet/Workbook format or Word documents. This will be a huge leap in helping the lecturers to manage their titles for WXES/T 3181 upon the completion and successfully implementation of **OTMS**.

1.2 Problems Definition

The current thesis management system is not efficient. It is very troublesome and not handy for all lecturers, administrators and even students. The current thesis management problems can be categories into three sections, which are the administrator section, lecturer section and student section.

1. ***Administrator:***
   - Need to manually check all the thesis registration form (Borang Pemilihan Tajuk Projek WXES/T 3181)
   - Highly responsible on entry of the correct information for each student
   - Troublesome data entry process using Microsoft Word or Excel format
   - Must avoid the purposely wrong thesis registration form from students
   - Must avoid submission of more than one form by any student
   - Difficult to contact students if the thesis registration form has errors

2. ***Lecturers:***
   - Must submit all their thesis titles to the Head of Department or Projek Ilmiah committee members, who will then pass it to the administrator to put on the Web site
   - Difficult to make changes to their titles, lecturers must go through the Head of Department or committee members
   - Lack of facilities to help lecturers to check their student's record (which student registered for which title under them)
   - Might be approached by a lot of students even if they don't have any title left
   - Difficult to avoid students from making more than one registration

3. ***Students:***
   - Have to queue up to get the thesis registration form
   - Have to go to the office just to confirm the title by submitting the thesis registration form
   - Need to rush to meet the lecturers to register or ask information regarding the titles
   - Sometime have to skip class just to get or summit the form and to meet the lecturers
   - Does not know the status of each title (either available or not)

1.3 Project Objectives

This project is aimed to develop a brand new Web application for lecturers to manage their thesis titles. Through the Web, the relevant and available information can be gained easily for any modifications and comparisons. Objectives of the proposed system are as below:

i. ***Compare the thesis titles***

Lecturers can compare their suggested thesis titles with other lecturer's titles before they put it on the website through **OTMS**. This is very important to avoid titles that are exactly the same (and uses the same development tools) coming out from two different lecturers. Comparison will be done automatically when the lecturer uses the compare function provided by **OTMS**.

ii. ***Manage the thesis titles***

Lecturers can add, delete, search or modify their own thesis titles via Internet. All they need is a browser (IE 5 or compatible) and connection to the Internet. Using this online method, the thesis titles can be managed easily no matter where and when. All the records will be stored in a Web based database for easy and direct access all the time.

iii. ***View the thesis titles***

All the undergraduates can view the thesis titles from the website before they choose the appropriate title for themselves. Using **OTMS**, the status of a certain title, which is available or not available, will be display out. This is very important to the students before they approach the lecturers.

iv. ***Avoid students from registering with more than a lecturer***

With **OTMS**, any student can only register a thesis title once, with any of the lecturers. Students can no longer register with more than one lecturer because the

system will record the student's matric number upon his/her first registration. When the same student approaches another lecturer and try to register again, the system will abort the registration process and notify the lecturer. This is very important to ensure fairness competition among the students.

v. ***Provide better information to meet the students' need***

Students can get the full detailed information about the thesis titles offered for a particular session. For every title, information such as descriptions, suggested development tools, lecturer in charge, status, session, etc. can be preview by students just on their monitor screen. All this information is vital for students before they make their decision on which title is more suitable for them.

vi. ***Create a paperless environment***

Currently, most of the information or records are kept in a paper form. Students need to fill in a form (Borang Pemilihan Tajuk Projek WXES/T 3181) after they get the lecturers' approval for certain title. With the implementation of **OTMS**, all this information can be shifted to computer storage. The lecturer just need to key in the student's name and matric number into the system for instant storage. Therefore, students will no longer need to fill in any form and a paperless environment can be created.

vii. ***Save students' time***

Students will no longer need to meet the individual lecturer to know whether a specific title is still available or not. They can always check the status of the titles through the system at anytime, anywhere for the latest update by the lecturers. Therefore, students can go straight to meet the other lecturers that still have titles available and this can save their valuable time. They do not need to go door by door and meet every lecturer in the faculty. Students also do not need to queue up to get the form from the office once this system is implemented.

1.4 Project Scope

The implementation of the Online Title Management System is to enhance the process of managing the thesis titles for WXES/T 3181 by using computer and Internet based solutions. Some of the technologies related to Web development will also be added into **OTMS**. Generally, this system will integrate 3 different modules, which are:

- Administrator module
- Lecturer module
- Student/visitor module

1.4.1 Administrator module

On the system administrators' side, all information related to the thesis titles are presented. This module not only provide a full view of all the titles, but also includes some user friendly interfaces for system administrators to add, delete, modify or search for all records.

Beside that, administrators will also be in charge of the user registration for the system. All lecturers must register with the system administrators to become authorized users of **OTMS**.

This module gives the system administrators the privileges to access all the titles and records stored in the database. System administrators can perform different tasks on all the titles without any limitations. They just need to login to this module to access their privileges.

Overall, this module is about creating a role, which is the administrator to maintain the user registrations and all the records. With this module, authorized users for the system can be easily trace and control.

1.4.2 Lecturer module

The lecturer module provides end-user (lecturers of FSKTM) with the abilities to maintain their own thesis titles. Although this module has some similarities with the Administrator module, it is not totally the same.

First of all, lecturers must register with the system administrators before they can actually access this module through the login page. Lecturer module does not provide the registration functionality.

Beside that, lecturers will have fewer privileges compared to the administrators. Although lecturers can also add, delete, modify and search for titles, they can only perform the first three tasks (add, delete and modify) on their own titles. This mean a lecturer cannot and does not has the rights to modify other lecturers' titles. However, they can still search for all the titles in the database.

The only extra feature of this module is the compare function. This function will compare the titles that a lecturer wants to add into the database with the existing titles to check whether there is duplication. In other words, this can help to prevent two titles that are exactly the same (even with the same department, session and semester) from being added into the database.

1.4.3 Student/visitor module

The last module of the system is also the end-user (students or other visitors) module. This module has the least functionalities compared to the other two modules. For security reason, students can only view a list of titles offered to them by the lecturers. This module will list out the status for every title, whether it is still available or vice versa, and other information about the title. They can also search the database to find out any titles that they have in mind or interested in. For this module, no logon is required, any students and visitors can easily access this module to get the information they desire.

1.5 Project Significance

This project is significant since the numbers of students enrolled in FSKTM are consistently increasing year by year. This means the numbers of students who are going to take the WXES/T 3181 course will also follow the same trend as that particular course is required to complete the degree studies. These make the titles' management becomes more and more difficult. Things will tend to get complicated and lecturers will need to spend a lot of time in managing their titles. It would be a very difficult task to solve this problem manually.

With this system, lecturers can expect time saving features in **OTMS** to help them a lot. This will probably make the process functioning more effective and efficient. Furthermore, students will also benefits from the system. They can look for suitable titles online and then start hunting for it.

1.6 Project Expected Outcomes

The outcome that is expected from the completion of the Web based **OTMS** is an online title management system for all the lecturers in FSKTM who are involved with the WXES/T 3181 course. The end users for this project will mainly consist of two categories, which are lecturers and students.

Lecturers can easily manage their titles anytime and anywhere through the Internet. Functionalities provided by the system can help lecturers to avoid from setting exactly the same title (and even the same department, session and semester) for a particular semester and session. Beside that, the system can also prevent students from registering with more than one lecturer.

In the meantime, students can access the system to hunt for the titles available. The status of every title, either available or not, will be display clearly through the computer

monitors. To make life easier, what you need to access the system is just a browser with connection to the Internet. This means that students can use the system even at their home.

Beside the two main categories of end users, another group of users, which are system administrators, will help to maintain the system integrity. Administrators' main task is to register the lecturers to become authorized users of the system. This is very important to prevent unauthorized personnel from tampering with the system. The other task of the administrators is to maintain all the records whenever it is necessary.

1.7 Project Expected Strengths

This project is expected to possess a few strengths that enable us to improve or overcome our current problems. Its expected strengths can be categorized and are listed below.

i. ***Friendly user interface***

This system is expected to possess a friendly user interface that is easy to use. GUI components such as command buttons, check boxes, drop down lists, and radio buttons are used to attract the users to navigate through the system. Authorized users are able to easily locate all the system's functions through a well-designed menu.

ii. ***Easy accessibility***

This system could be accessed easily using a web browser where it could be downloaded free from any authorized web site such as Internet Explorer from http://www.microsoft.com. Moreover, access to the system is made available at all the time, 24 hours a day and 7 days a week.

iii. ***Relatively fast response time***

This system will be designed in such a manner that its web pages will be loaded in a reasonable amount of time. This is to ensure that users do not need to wait too long to view the pages. All usage of unnecessary components such as graphic files, multimedia files and others will be kept at the most minimum level to ensure the relatively fast response time.

iv. ***Enforce different access levels***

This system is expected to enable administrators to have a different access level from the other end users. This is to provide a better and secure environment among administrators. Beside that, lecturers will also reside in another access level. The purpose of this is to enable different users perform different tasks in the system. However, administrators will have the highest access level, followed by the lecturers and finally the students.

v. ***System validation***

This system is expected to provide a validation system for its users. It will validate the user to be either authorized or unauthorized. Certain modules in the system can only be access by the authorized users. This validation system also makes sure that only the creator of the thesis titles will have the rights to modify the records (except for under certain circumstances, administrators will need to take over the task).

1.8 Project Expected Limitations

The target users for the system are lecturers and students from FSKTM as the main users. Administrators and any other visitors will be considered as minority users. Due to the limitations of time and human resources, this system will be used in FSKTM only with no integration or link with other faculties. This means any other faculties in University of Malaya that provide similar thesis paper for their students will not be able to use the system.

Beside that, **OTMS** will only be use for the subject WXES/T 3181: Projek Ilmiah I. The system cannot be used for other subjects included in the Master or PhD levels offered by FSKTM. However, expansion can be made in the future if necessary.

Last but not least, advance booking or reservation for thesis titles cannot be made through the system. This feature will not be considered to be implement into the system to ensure that all students will have the same opportunity to get their titles respectively and thus the unfairness competition among students can be avoided.

1.9 Project Schedule

Project scheduling is a particular demanding task for software managers. Managers estimate the time and resources required to complete all the activities and organize them in a coherent sequence. Project scheduling involves separating the total work involved in a project into separate activities and judging the time required to complete these activities. Usually, some of these activities are carried out in parallel to shorten the total project duration.

In estimating schedule, it should not assume that every state of the project would be problem-free. If the project is new and technically advanced, certain part of it may turn out to be more difficult and take longer than originally anticipated.

A guideline for estimating is to estimate as if nothing will go wrong then increases that estimate to cover anticipated problem. A further contingency factor to cover unanticipated problems may also be added to the estimate. This extra contingency factor depends on the type of project, the process parameters (deadline, standards, and so on) and the quality and experience of the software engineers working on the project.

This project schedule is usually represented as a set of charts showing the work breakdown, activity dependencies and staff allocations. Bar charts and activity networks are graphical notations, which are used to illustrate the project schedule.

Task Name	Jun'01				Jul'01				Aug'01				Sep'01				Nov'01				Dec'01				Jan'02			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
Project Definition																												
Literature Review																												
System Analysis																												
System Design																												
System Development																												
System Testing																												
Documentation																												

Table 1.1: Project Schedule for OTMS

Chapter 2:

Literature Review

Chapter 2: Literature Review

2.1 Literature Review

What is a Literature Review? A literature review is an account of what has been published on a topic by accredited scholars and researchers. Occasionally it is required to write one as a separate assignment (sometimes in the form of an annotated bibliography), but more often it is part of the introduction to an essay, research report or thesis.

2.1.1 Purposes of the Literature Review

The main purpose of the literature review is to guide the students or researchers using the best way to access and analysis information regarding their research topic. It also helps the students to develop their information seeking and critical appraisal skill. Students can also recognize relevant information, synthesize and evaluate it according to the guiding concepts. Besides, writing the literature review is also to convey to the reader what knowledge and ideas have been established on a topic, and what their strengths and weaknesses are. As a piece of writing, the literature review must be defined by a guiding concept (e.g., the research objective, the problem or issue that are discussing, or some argumentative thesis). It is not just a descriptive list of the material available, or a set of summaries.

Besides enlarging the knowledge about the topic, writing a literature review can gain and demonstrate skills in two areas:

1. **Information seeking**: ability to scan the literature efficiently, using manual or computerized methods and also to identify a set of useful articles and books
2. **Critical appraisal**: ability to apply principles of analysis to identify unbiased and valid studies

The following steps are required in a literature review:

a. Be organized around and related directly to the thesis or research question that are under developing
b. Synthesize results into a summary of what is and is not known
c. Identify areas of controversy in the literature
d. Formulate questions that need further research

2.1.2 Techniques on conducting Literature Review

Information is essential to do a good research or analysis. For this project, several techniques have been taken to seek information. These techniques are listed as follow.

i. ***Refer to reference books from the library***

A lot of reference on conference, workshop, journal and symposium can be widely found from the library. Reference books on methodology and system design also can be found from the library. With the helps of these reference books, students can easily access any information that they need to complete the project.

ii. ***Search for information from the Internet***

Nowadays, Internet is the main source of information. Relevant information on Web applications, client-server technology and programming tools can be found through the Web-based search engines. Detailed analysis of some of the technologies is provided for the public. Current systems that are similar with the proposed system can also be viewed and compared.

iii. ***Do analysis on the pass year thesis***

As FSKTM provides students with a Documentation Room, several pass year thesis documentations have been studied in order to identify any potential mistakes and to gain some skills on software development. The format on how to do a complete

report and documentation can be found from the previous thesis documentations. It is a very useful resource for students to access.

iv. ***Refer to newspapers and magazines***

Reference on the latest newspapers such as INTECH (The Star), Computimes (New Traits Times) and magazines such as PC World is one of the techniques to gain the knowledge on latest technologies available. However, the technologies available must be choose wisely for the project.

v. ***Have discussions with friends, staffs and lecturers***

Useful advises have been given for each session meeting conduct with my supervisor and other lecturers or related staffs. It is a useful method for error correctness and acted as reminder when carried out the system development process.

vi. ***Conducting interviews and surveys***

Interviews and surveys have been conducted with the administrator, lecturers, related staffs and students to find out the lacks of the current system. After that, any potential improvements can be pointed out to implement in the suggested system.

2.2 Types of Items Reviewed

In this specific thesis, I have decided to do a literature review on some of the items involved in the project. Below is a list of the items:

- Existing system
- System architectures
- Internet
- Web browsers
- Database
- Programming languages
- Programming tools
- System platforms
- Web server software
- Methodology

Because of time factor, I only have the chance to review a few examples for each of the items mentioned above.

2.3 Existing System Review

2.3.1 Current system in FSKTM

Currently, there is no database to record the thesis titles in Faculty of Science Computer and Information Technology, University of Malaya. Although the thesis titles for each semester in every session are not that many, some or all of these information might be lost without proper handling and documentation.

Let us look at the processes concerning the current thesis titles management. Every lecturer who is involved with WXES/T 3181 (Projek Ilmiah I) will come out with his or her own suggested thesis titles. After that, he or she will submit the thesis titles to the Head of Department or the Projek Ilmiah committee members, who will then pass it to the administrator to put on the FSKTM website, which is located at this URL: http://www.fsktm.edu.my/~ilmiah/main.htm

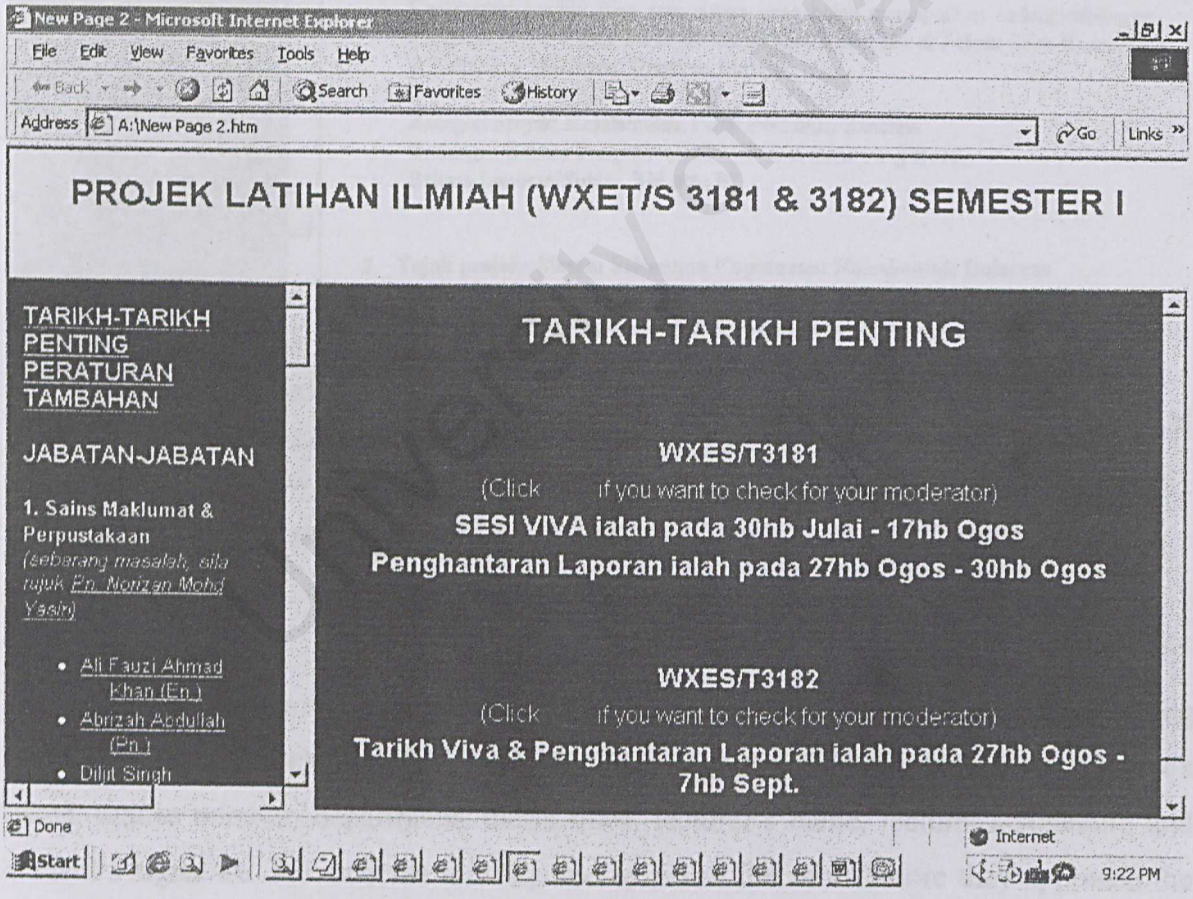


Figure 2.1: The Main Page of Projek Ilmiah (WXES/T 3181) Web Site

From the above URL, students can browse all the titles available for that particular semester before they approach the lecturers to get their approval and signature. Students can browse the titles suggested by a particular lecturer just by clicking on the link provided (the link is the lecturer's name). Lecturers are grouped according to their department so that every student can choose the lecturers from their own department.

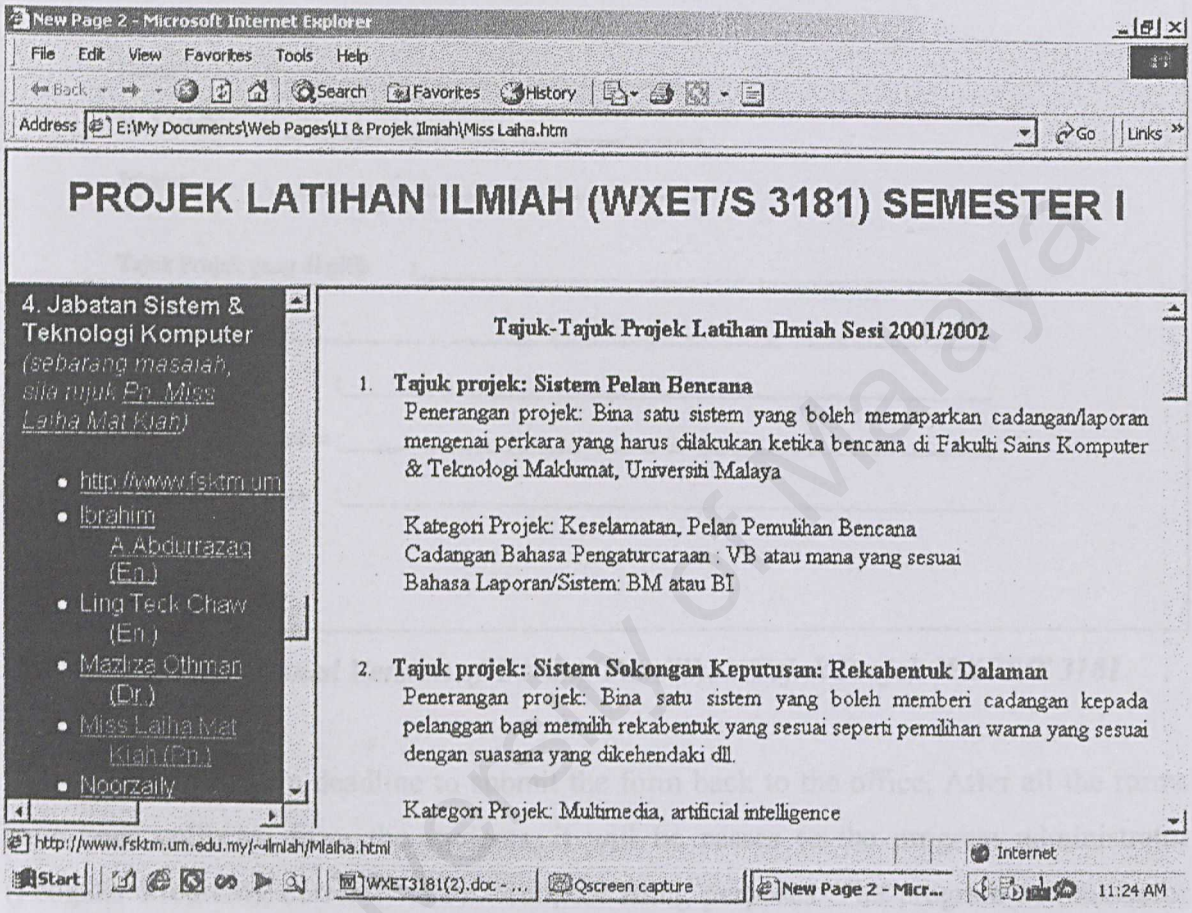
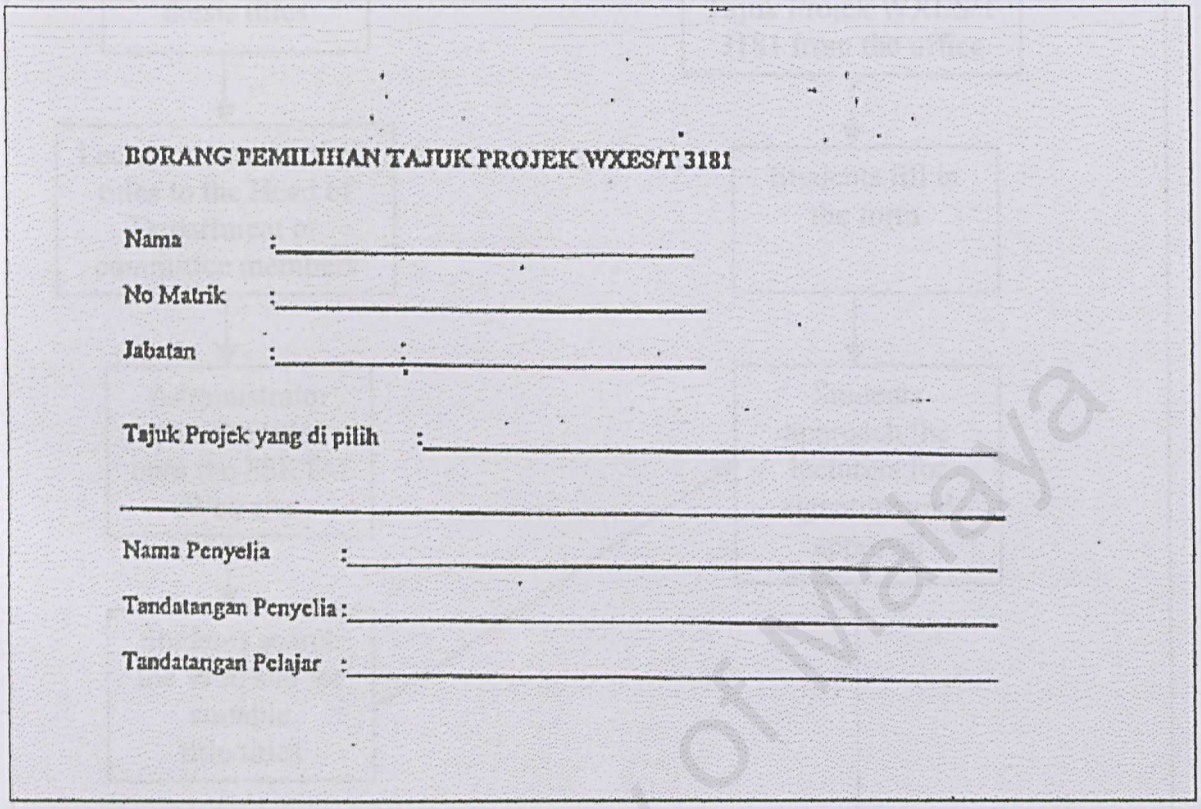


Figure 2.2: Web Page Displaying The Titles Offered By Pn. Miss Laiha From Jabatan Sistem & Teknologi Maklumat

After that, the students will need to get a form from the office, which is Borang Pemilihan Tajuk Projek WXES/T 3181. This form contained details such as the student's name, matric number, department, thesis titles, lecturer's name, lecturer's signature and student's signature. All students are required to fill in the form before they approach the

lecturers. If a lecturer agreed to put a particular student under his or her supervisions, then he or she must sign on the form as approval.



BORANG PEMILIHAN TAJUK PROJEK WXES/T 3181

Nama :_____

No Matrik :_____

Jabatan :_____

Tajuk Projek yang di pilih :_____

_____

Nama Penyelia :_____

Tandatangan Penyelia :_____

Tandatangan Pelajar :_____

Figure 2.3: Scanned Version of Borang Pemilihan Tajuk Projek WXES/T 3181

There will be a deadline to submit the form back to the office. After all the forms have been collected from the students, it will be passes to the program administrator (Pengatur Program), currently is Pn. Koh, for filing purposes. The program administrator will manually key in all the information from the forms into a Word Document, or sometimes in Excel Workbook or Worksheet format. This is a very tedious process and consumes a large amount of time.

The current processes of managing the thesis titles will be summarized in Figure 2.4 in the following page.

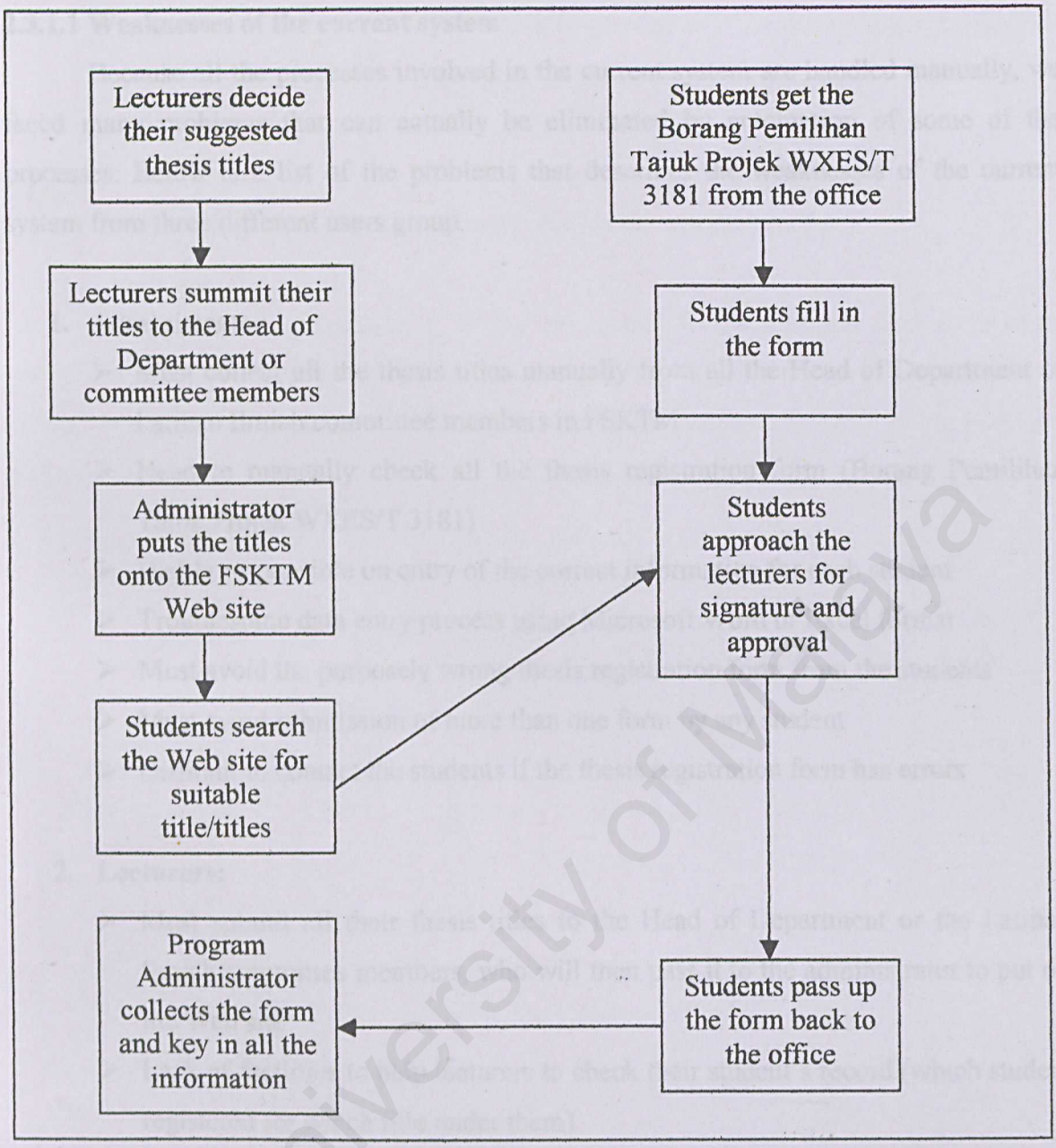


Figure 2.4: Processes Involved In the Current Thesis Titles Management System

2.3.1.1 Weaknesses of the current system

Because all the processes involved in the current system are handled manually, we faced many problems that can actually be eliminated by automation of some of the processes. Below is a list of the problems that describes the weaknesses of the current system from three different users group.

1. **Administrator**
   - Must collect all the thesis titles manually from all the Head of Department or Latihan Ilmiah committee members in FSKTM
   - Need to manually check all the thesis registration form (Borang Pemilihan Tajuk Projek WXES/T 3181)
   - Highly responsible on entry of the correct information for each student
   - Troublesome data entry process using Microsoft Word or Excel format
   - Must avoid the purposely wrong thesis registration form from the students
   - Must avoid submission of more than one form by any student
   - Difficult to contact the students if the thesis registration form has errors

2. **Lecturers:**
   - Must submit all their thesis titles to the Head of Department or the Latihan Ilmiah committee members, who will then pass it to the administrator to put on the Web site
   - Lack of facilities to help lecturers to check their student's record (which student registered for which title under them)
   - Might be approached by a lot of students even if they don't have any title left
   - Difficult to avoid students from making more than one registration

3. **Students:**
   - Have to queue up to get the thesis registration form
   - Have to go to the office just to confirm the title by submitting the thesis registration form

- Need to rush to meet the lecturers to register or ask information regarding the titles
- Sometime have to skip class just to get or summit the form and to meet the lecturers
- Does not know the status of each title (either available or not)

Because of the weaknesses in the current manual system, automation of some of the processes and the introduction of a database to keep all the thesis information is an inevitable step. Implementation of **OTMS** will try to solve most, if not all of the problems mentioned above.

2.3.2 Currently existing system on the Internet

The literature review has also been carried out to study and review the currently existing systems on the Internet. Although these systems are not exactly the same with the proposed **OTMS**, they do have some similarities in general (which is the browsing and searching function).

2.3.2.1 MIT Theses Online

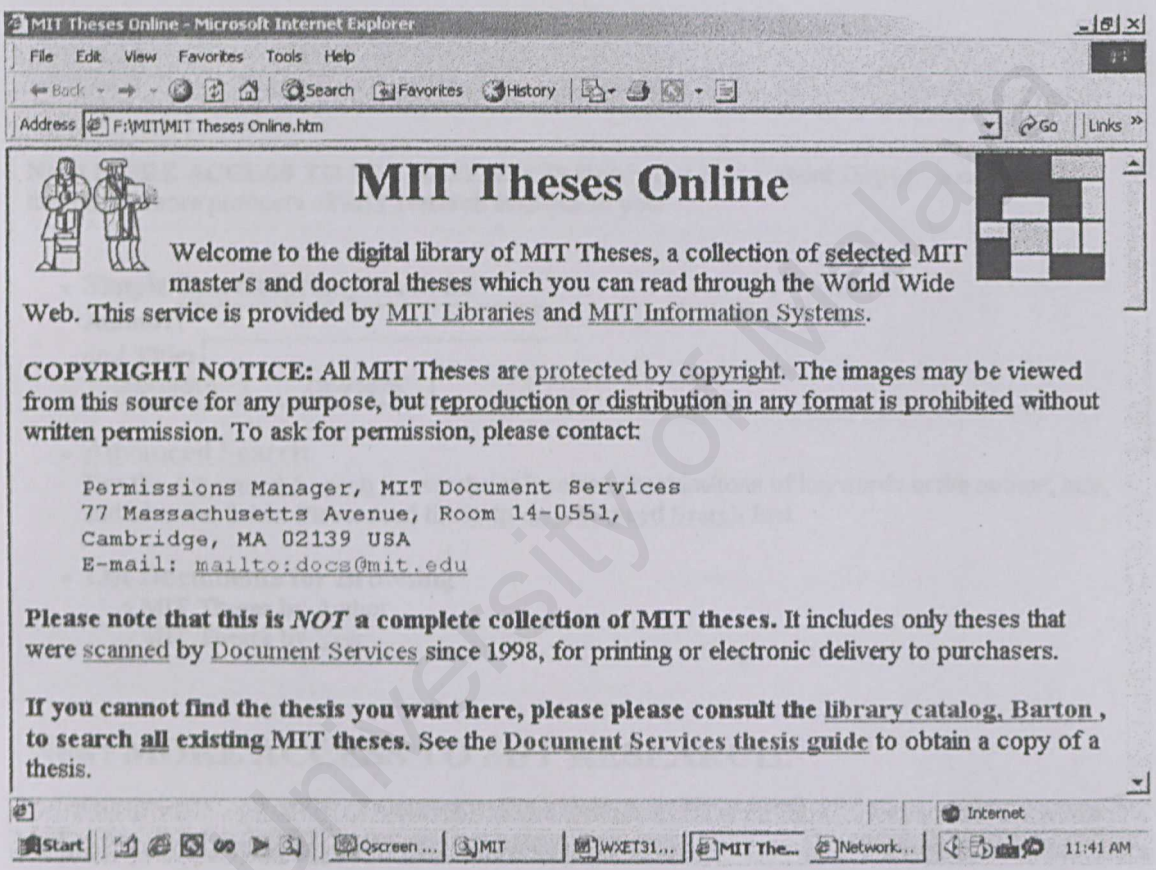


Figure 2.5: MIT Theses Online Home Page

The figure above showed the home page of MIT Theses Online, which is located at http://thesis.mit.edu. It is a digital library that keeps the MIT Theses, a collection of selected MIT masters and doctoral theses from all MIT departments, which we can read through the World Wide Web.

For this particular project, I will concentrate only on the browsing and searching function of the MIT Theses Online. This web site provides users with two types of searching options, which are simple searching and fielded (advanced) searching. For simple searching, users can input two fields – Author and Title. Fielded searching offers some extra fields such as Abstract, Organizations Selection, Document Identifier, etc., and the use of Boolean expressions (AND / OR). The search results will be display on a blank page with the number of theses found clearly stated.

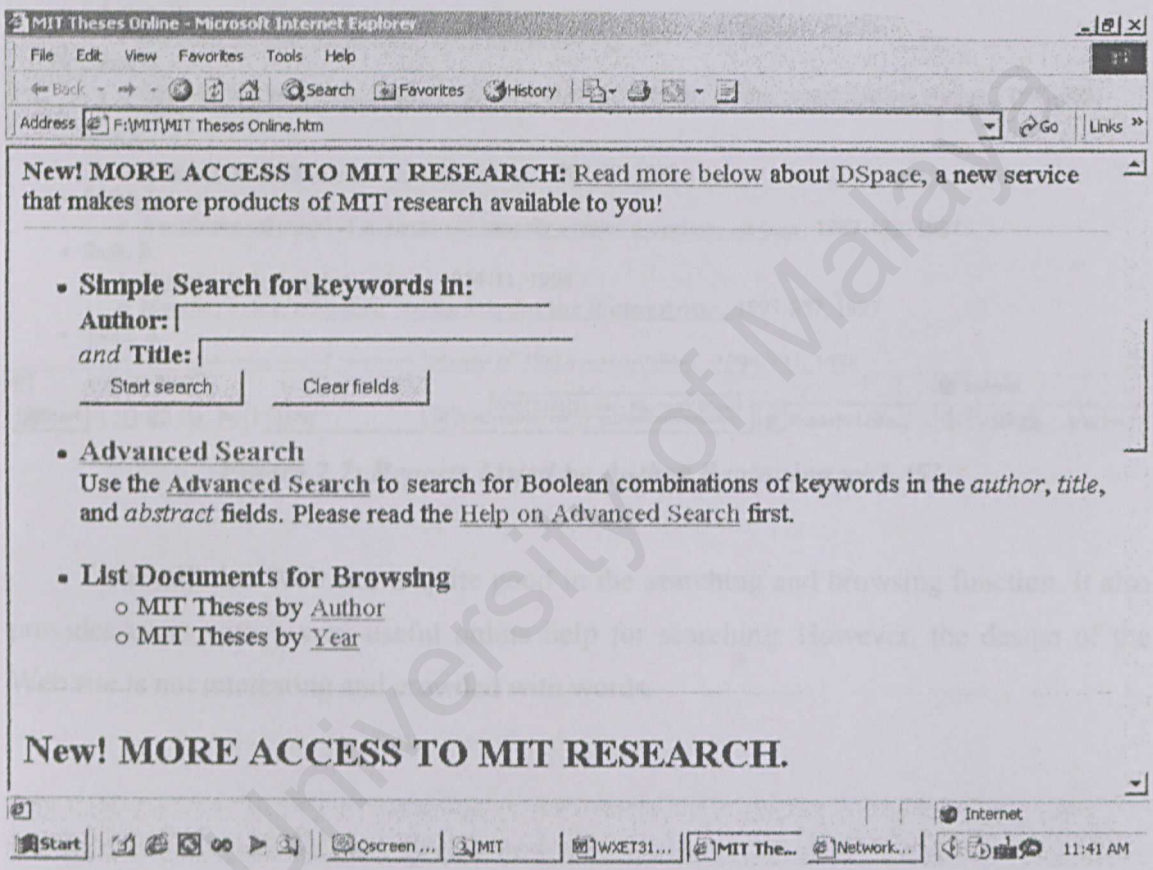


Figure 2.6: Simple Search Option of MIT Theses Online

Other than that, MIT Theses Online also enables users to list out all the theses for browsing purposes. Users can either list the theses by Author or by Year. The system will lists the reports on a blank Web page.

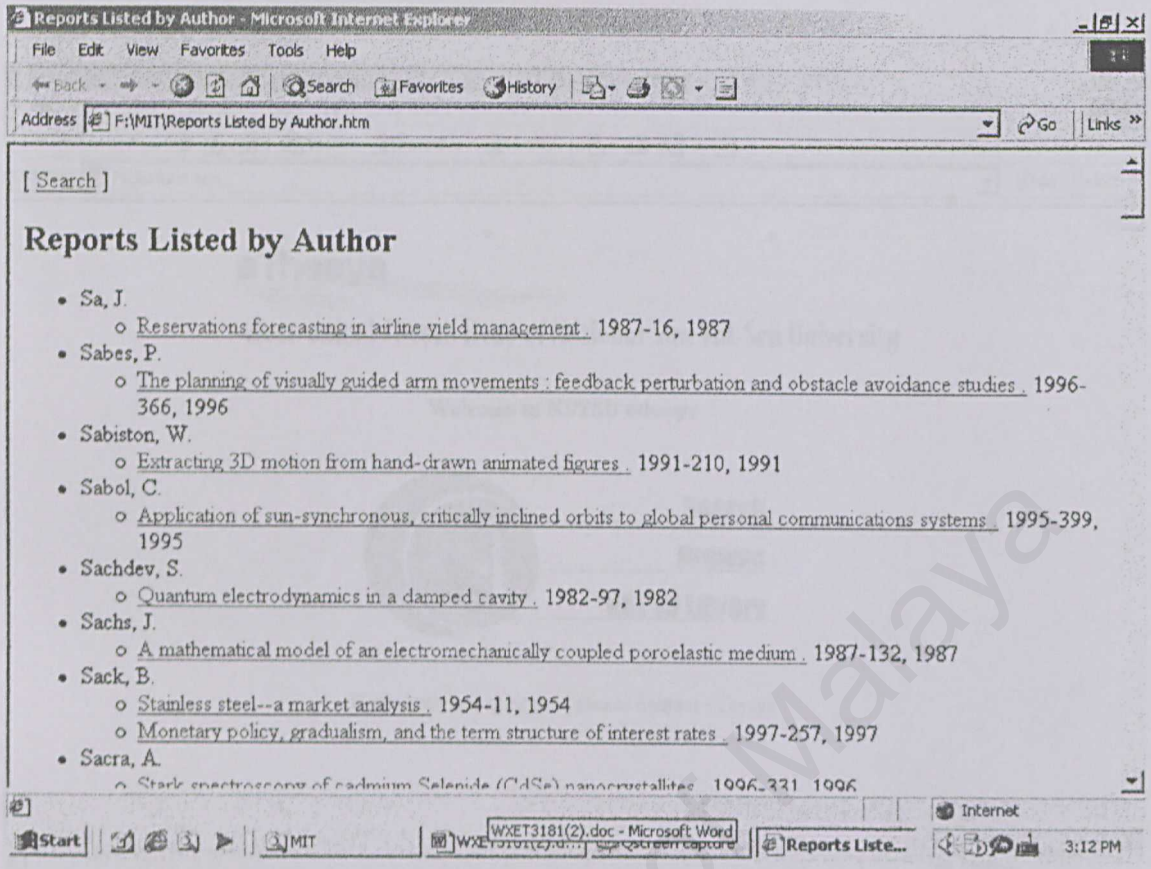


Figure 2.7: Reports Listed by Author Beginning with 'S'

Overall, this Web site is quite good in the searching and browsing function. It also provides users with a very useful online help for searching. However, the design of the Web site is not interesting and crowded with words.

2.3.2.2 NSYSU eThesys

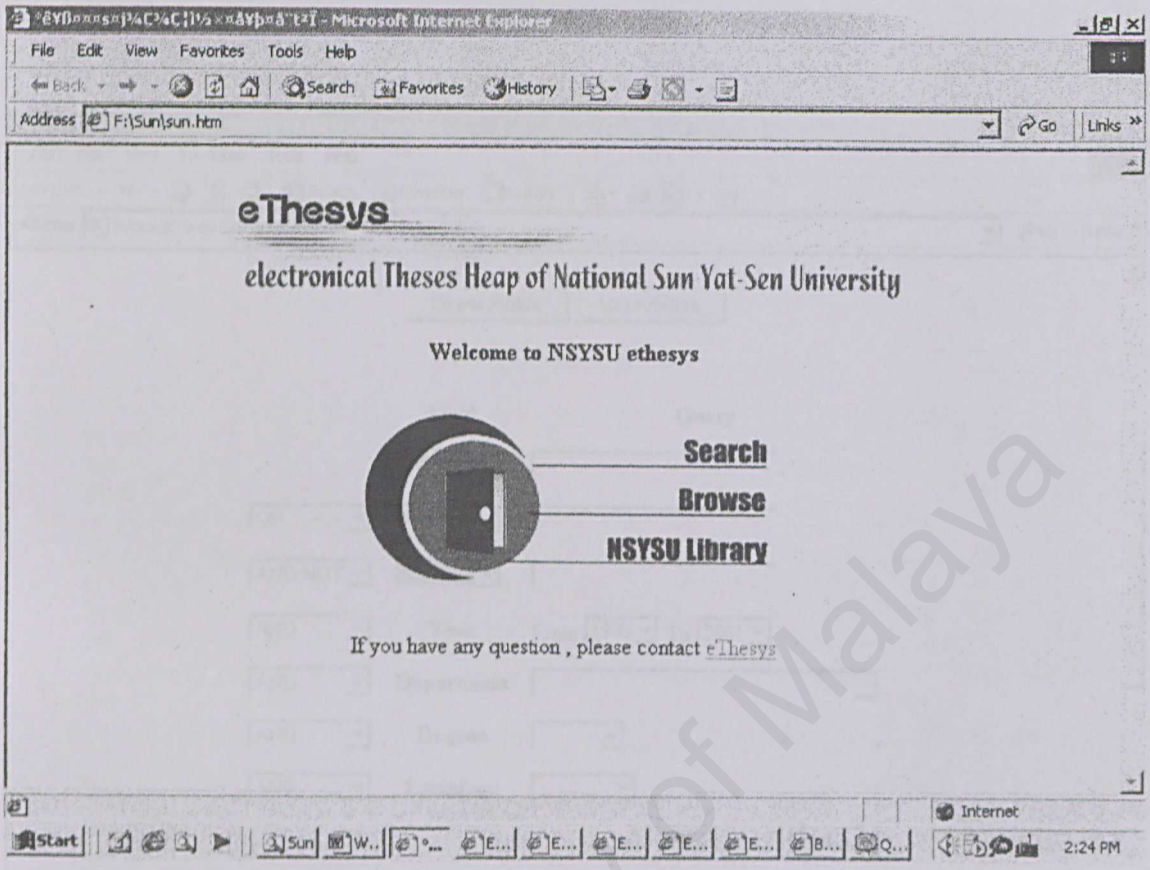


Figure 2.8: Home Page of NSYSU eThesys

The above Web site is prepared by the electronical Theses Heap of National Sun Yat-Sen University, located in Taiwan. The URL for the above page is http://www.lib.nsysu.edu.tw/eThesys/default_e.htm.

The functionality of this Web site is almost the same compared to MIT Theses Online, where both of them also provide the search and browse function. However, eThesys does provide a better search functions. It allows the users of the system to change the number of text boxes available for their queries besides providing fields like Title, Author, Abstract, Year, Department, Degree and Language. Users can also search by committee member name or by keyword. An extra Boolean expression, AND NOT also can be use together with AND/OR. Some fields are equipped with drop down list to simplify the user's

input process. The results of the search function are also displayed in a more organize manner compared to MIT Theses Online.

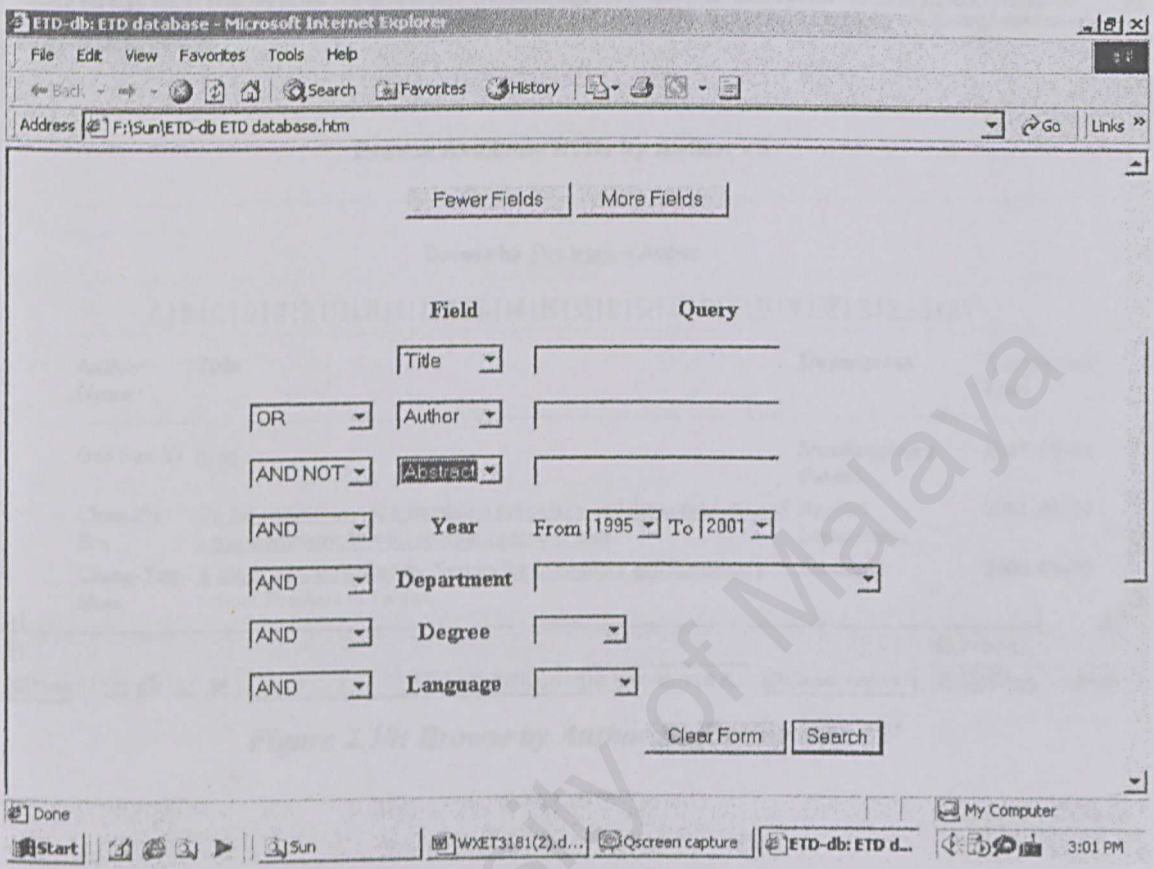


Figure 2.9: Advanced Search Option for eThesys

As mentioned earlier, eThesys also provides the browse function. Users can browse the available electronic thesis according to its department or author. The reports are displayed in a table-like format, which makes it look tidier and not too crowded.

Overall, eThesys is a better system compared to MIT Theses Online. eThesys provides a better advanced search function with more options for users to choose from and also more user-friendly with feature such as the drop down lists. The user interfaces of eThesys are also better because it is kept simple, clean and tidy. For the browse function, eThesys again outperformed MIT Theses Online in the way it displays the reports.

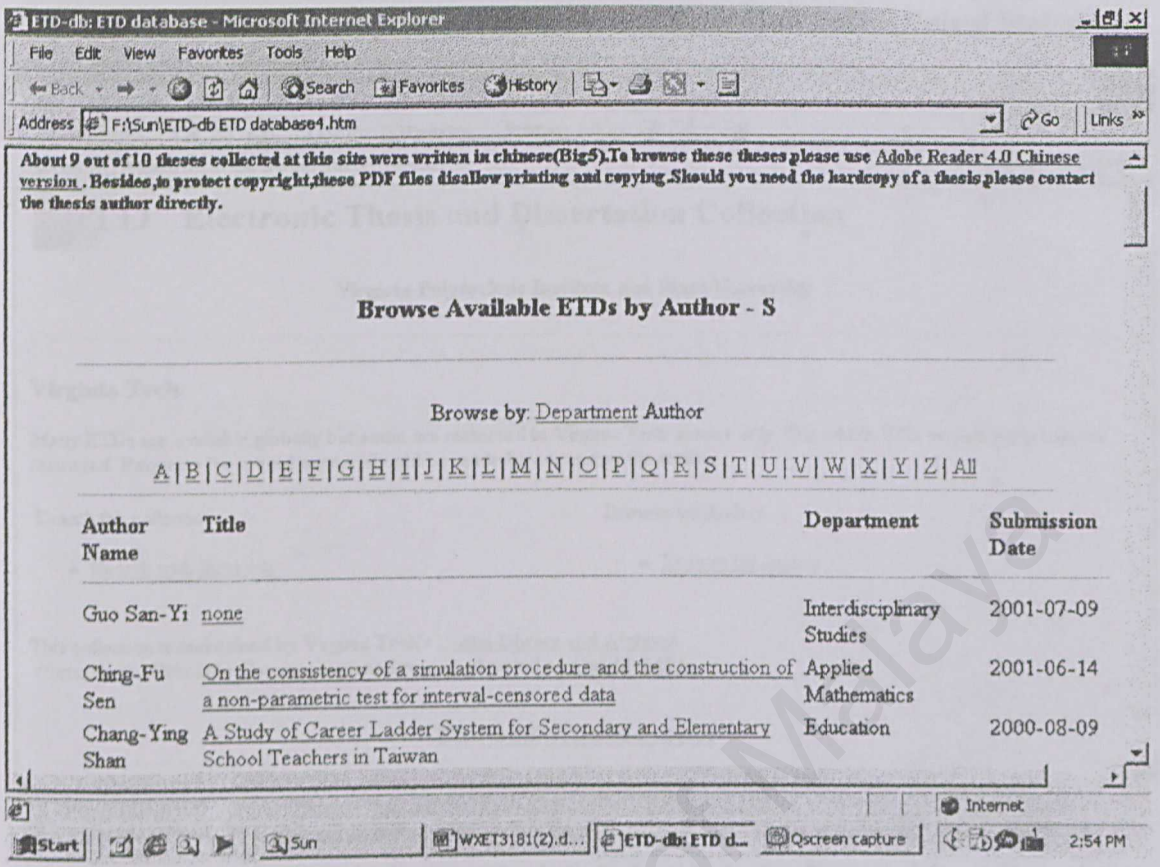


Figure 2.10: Browse by Author Beginning with 'S'

2.3.2.3 ETD of Virginia Polytechnic Institute and State University (United States)

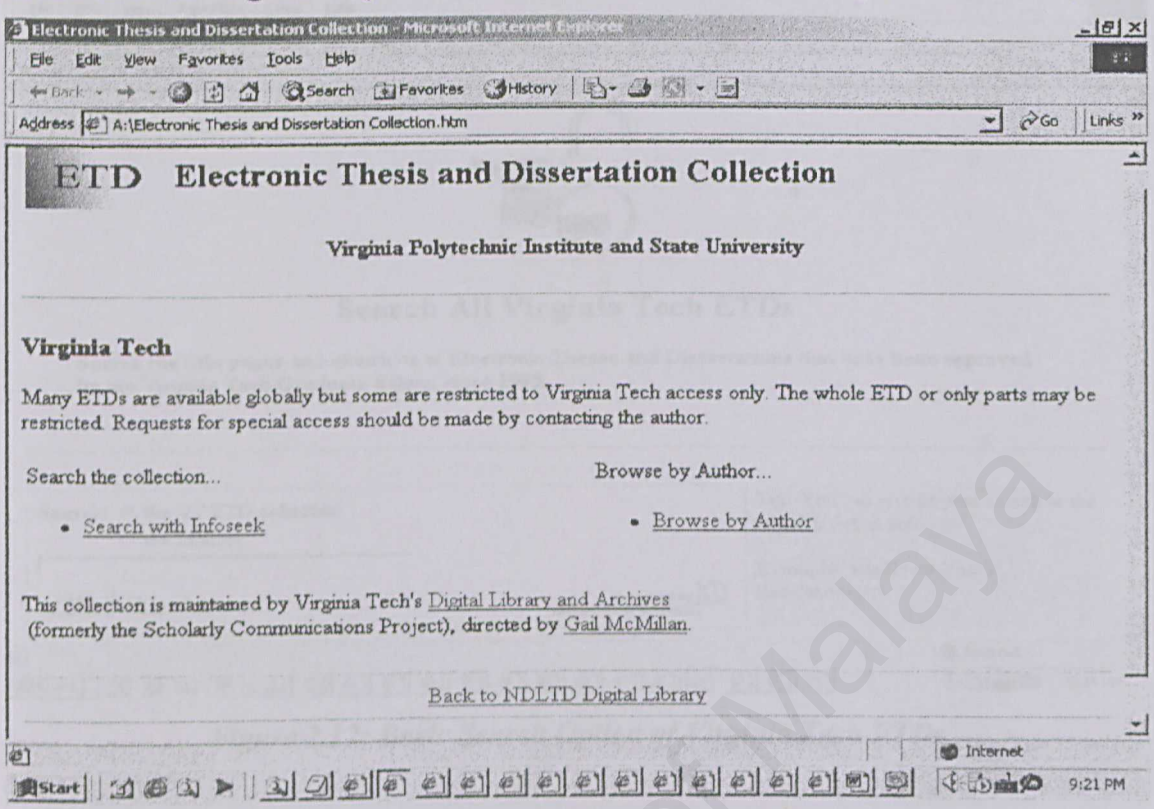


Figure 2.11: Home Page of ETD (Virginia Polytechnic Institute and State University)

The above Web site is from Virginia Polytechnic Institute and State University. The URL for the above page is: http://www.theses.org/vt.htm. This Web site also provides two services, which are search and browse. The search function consists of two options – basic and advanced search. The basic search option was almost the same with other Web sites and there is nothing special about it. However, the advanced search option is quite impressing with the use of drop down lists, radio buttons, and check boxes. These types of features enable users to perform their search more efficiently.

For the browse function, the main page itself will gives the users an idea of how many electronic theses and dissertations are unrestricted, restricted or mixed. If the users click on any alphabet, a page displaying the author's name, title and department will be shown. However, this result page is easier to read compare to the other system with the use of color bars separating one title from another (refer to ***Figure 2.15***).

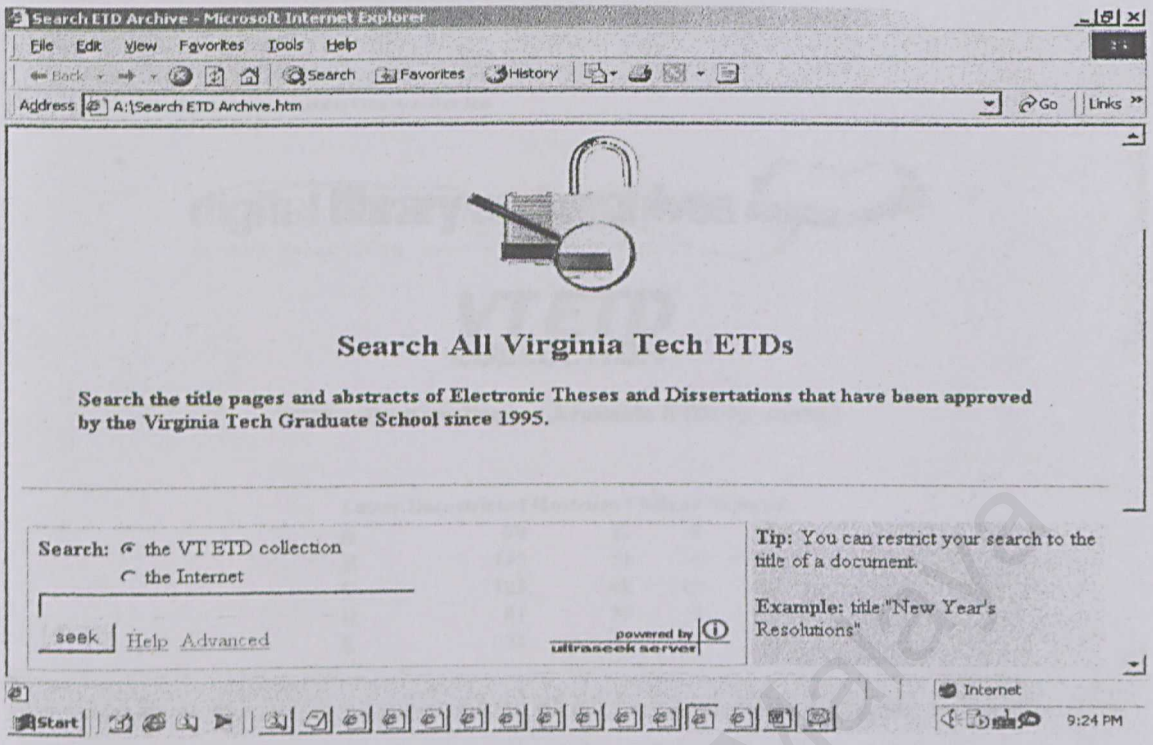


Figure 2.12: Basic Search Option of Virginia Tech ETDs

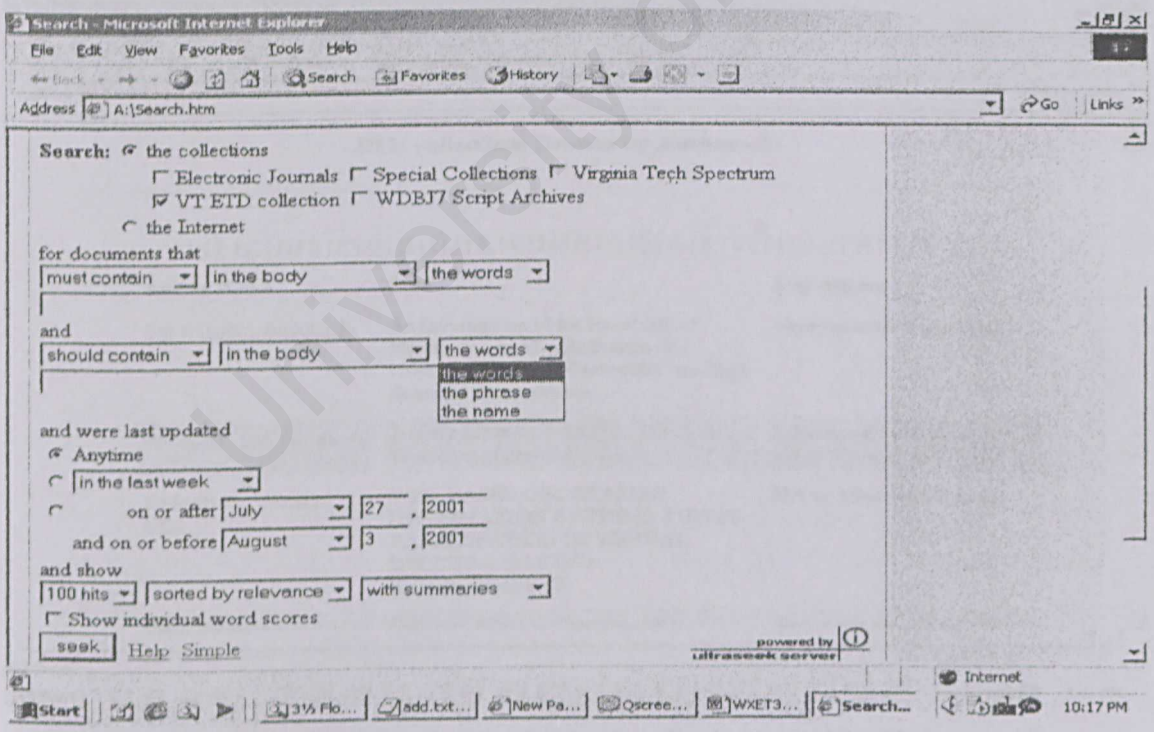


Figure 2.13: Advanced Search Option of Virginia Tech ETD

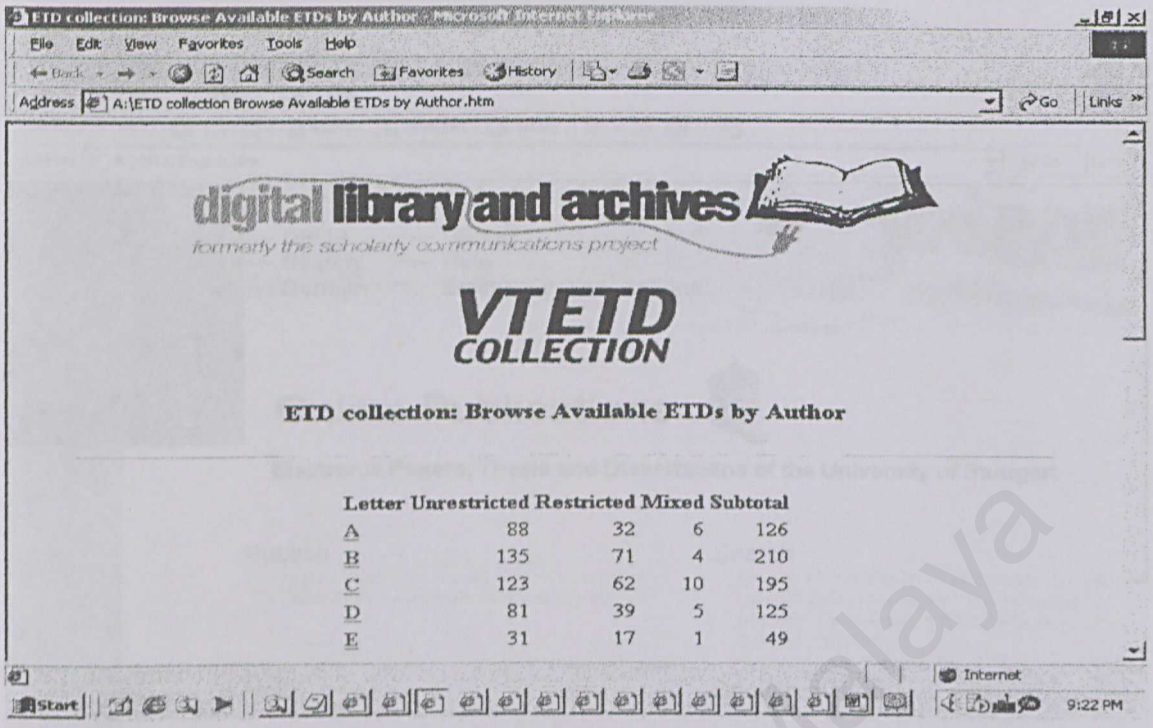


Figure 2.14: Browse Function of Virginia Tech ETDs

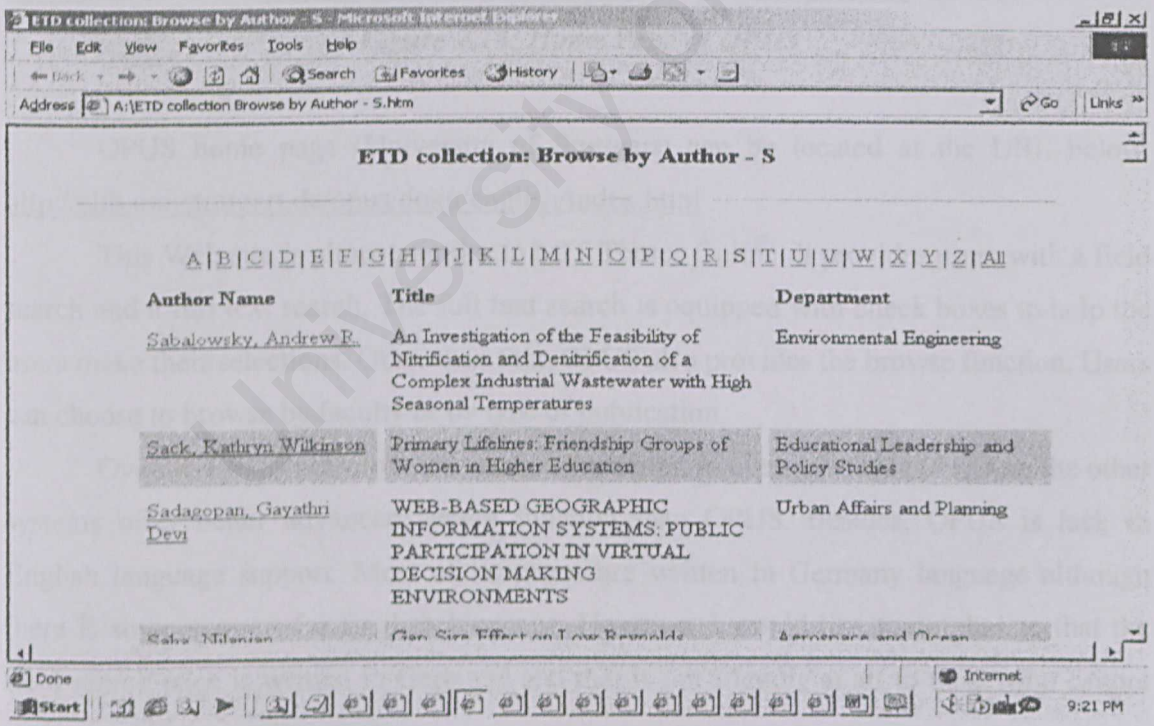


Figure 2.15: Browse by Author Beginning with 'S'

2.3.2.4 OPUS of University of Stuttgart (Germany)

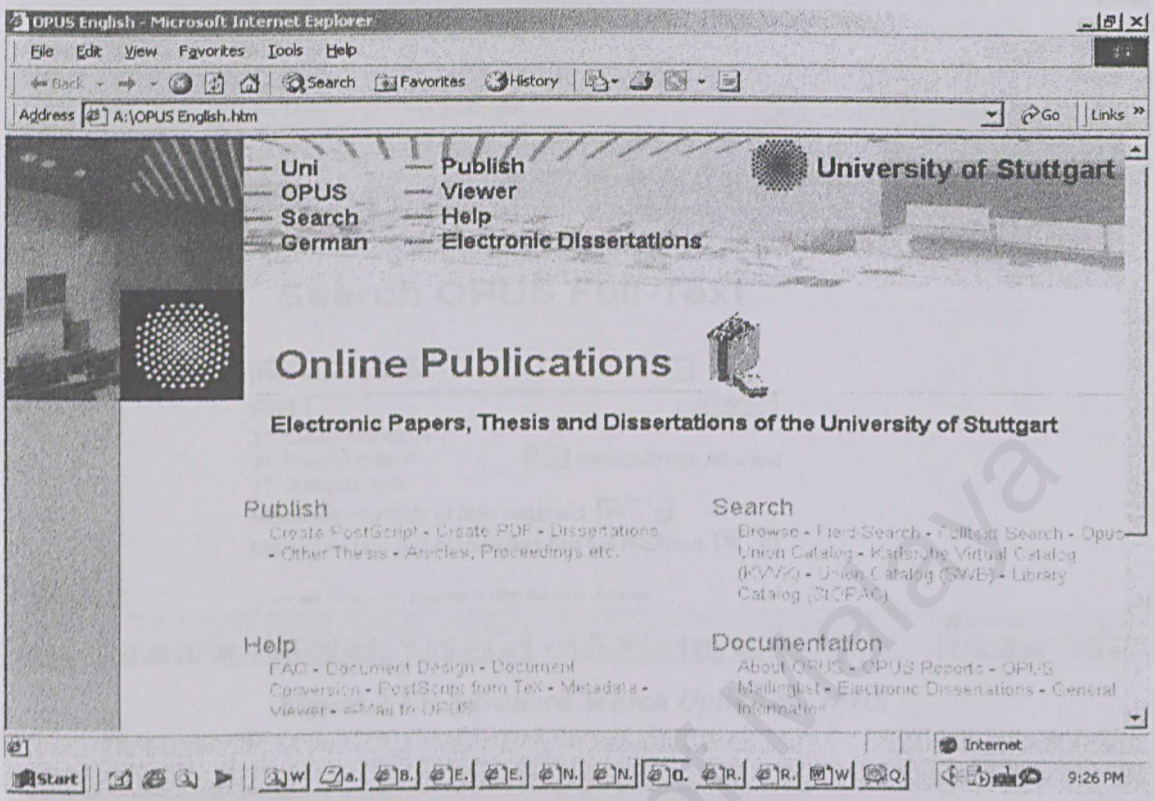


Figure 2.16: Home Page of OPUS

OPUS home page (University of Stuttgart) can be located at the URL below: http://elib.uni-stuttgart.de/opus/doku/english/index.html

This Web site is almost similar to MIT Theses Online. It provides users with a field search and a full text search. The full text search is equipped with check boxes to help the users make their selections. Other than that, OPUS also provides the browse function. Users can choose to browse by faculty or by type of publication.

Overall, OPUS performance can be considered as average. This is because the other systems offer better advanced search function than OPUS. Besides, OPUS is lack in English language support. Most of its pages are written in Germany language although there is some support for English language. However, I would like to emphasize that the field search page is written in Germany and that is not friendly at all to those who cannot understand Germany language.

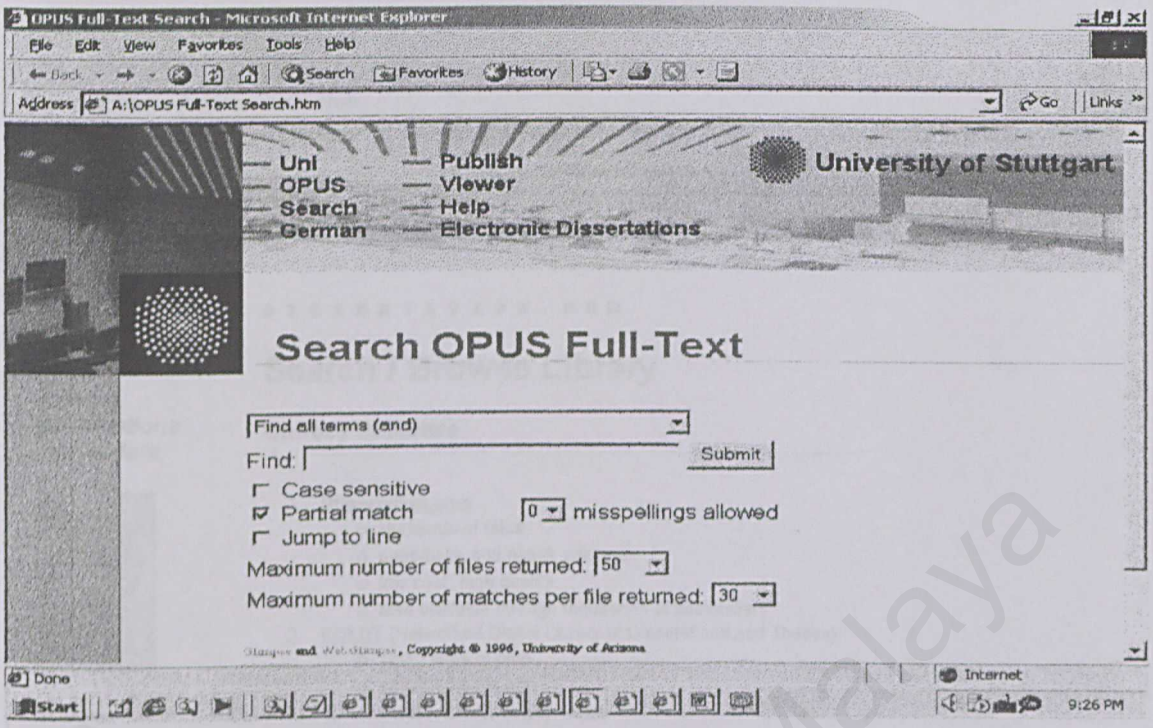


Figure 2.17: Advanced Search Option of OPUS

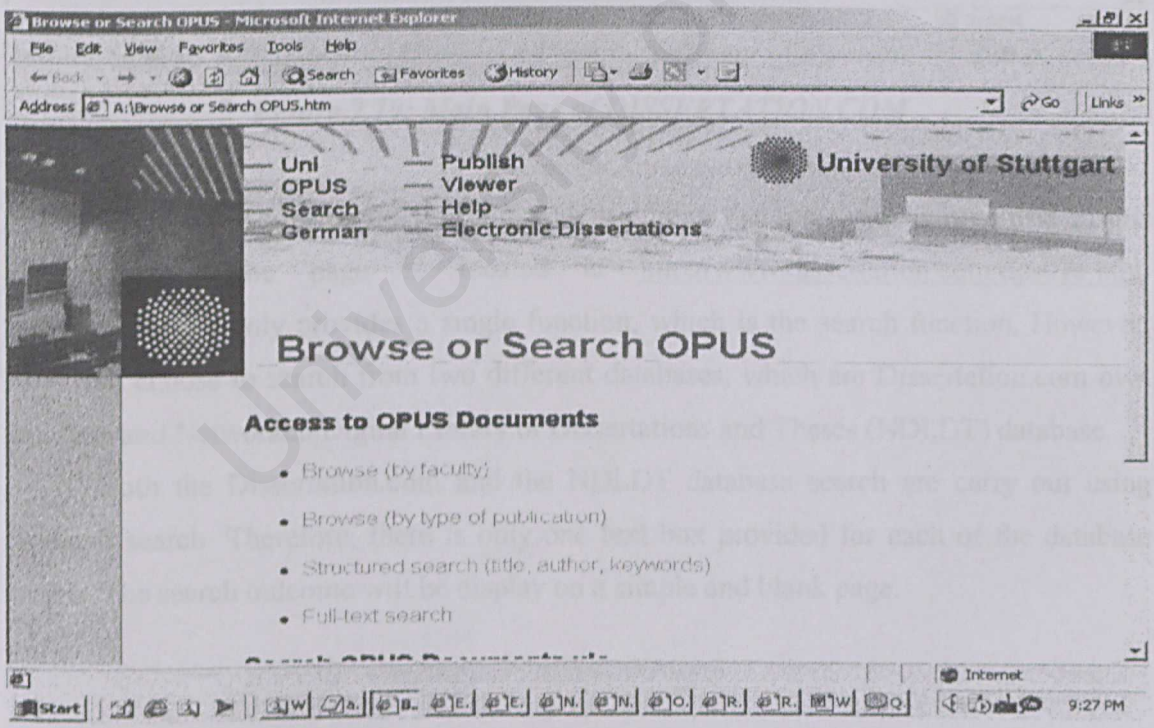


Figure 2.18: Browse Function for OPUS

2.3.2.5 Dissertation.com

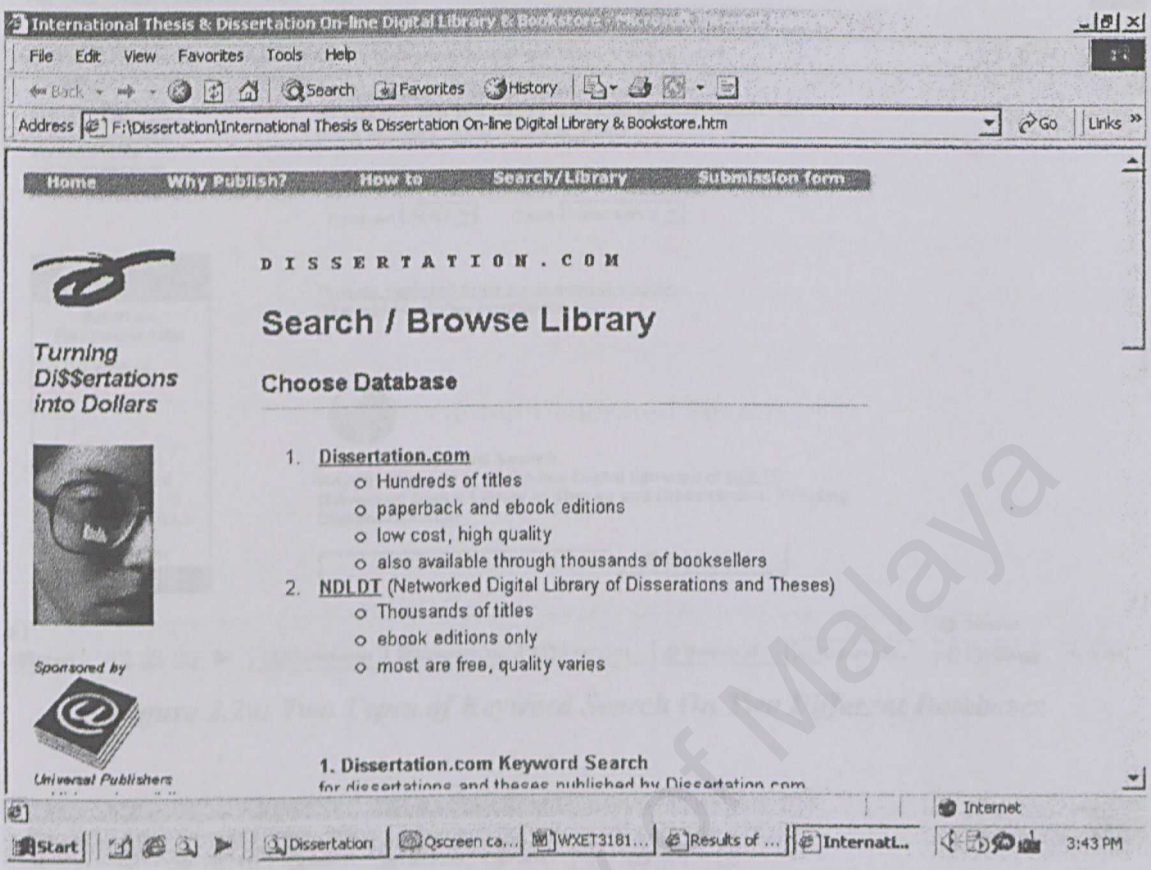


Figure 2.19: Main Page of DISSERTATION.COM

Dissertation.com is the simplest system among the five systems reviewed in this chapter. Its home page is located at http://www.dissertation.com/dps07a.htm. Dissertation.com only provides a single function, which is the search function. However, users can choose to search from two different databases, which are Dissertation.com own database and Networked Digital Library of Dissertations and Theses (NDLDT) database.

Both the Dissertation.com and the NDLDT database search are carry out using keyword search. Therefore, there is only one text box provided for each of the database search. The search outcome will be display on a simple and blank page.

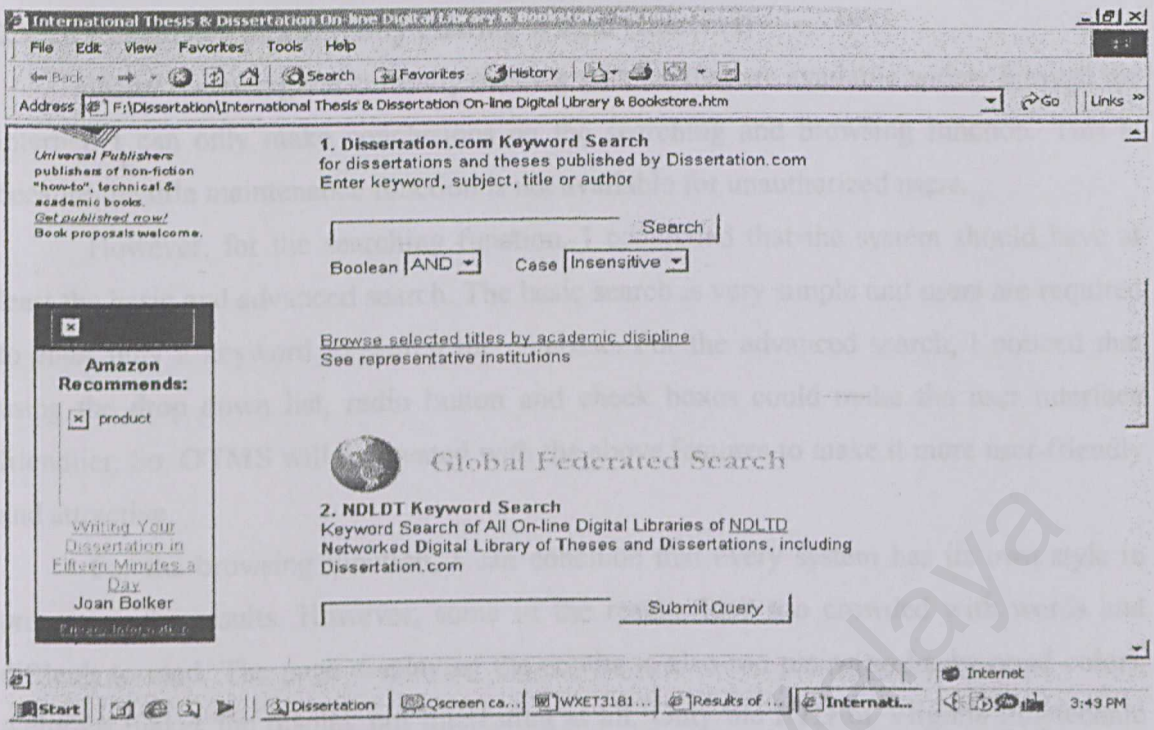


Figure 2.20: Two Types of Keyword Search On Two Different Databases

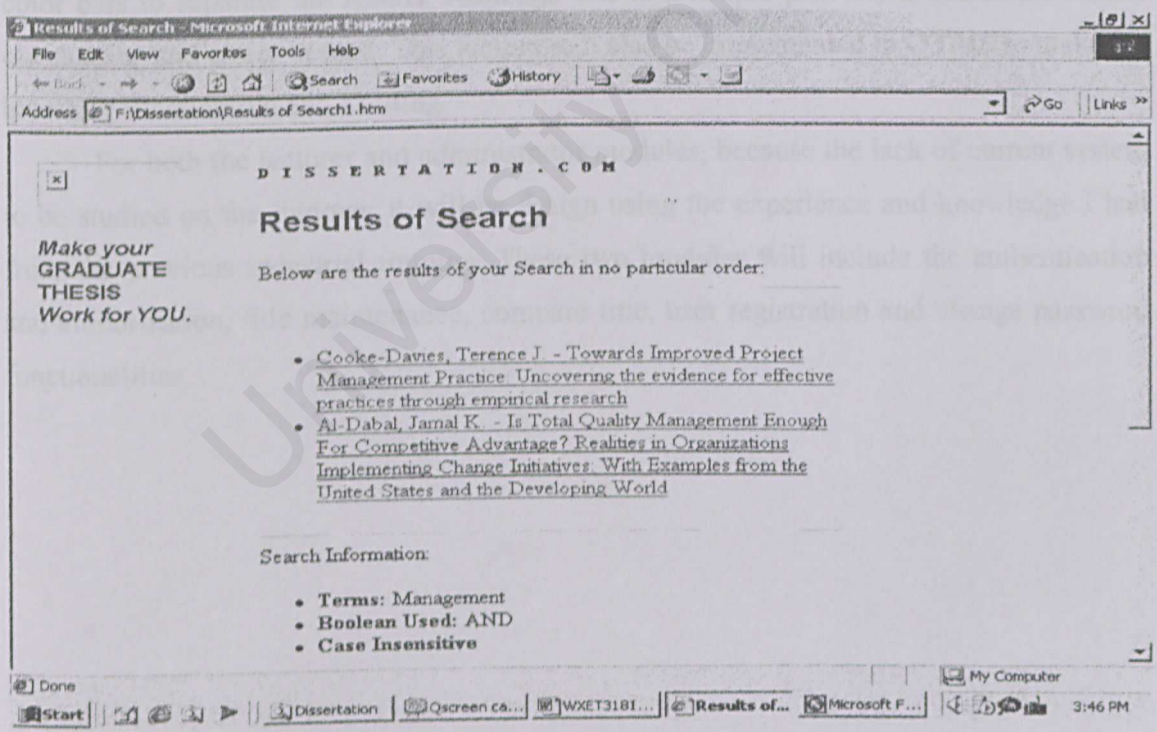


Figure 2.21: Results of Search from DISSERTATION.COM

2.3.2.6 Conclusion on currently existing system (Internet)

After reviewing the currently existing systems that are available widely through the Internet, I can only make conclusions on the searching and browsing function. This is because the title maintenance function is not available for unauthorized users.

However, for the searching function, I concluded that the system should have at least the basic and advanced search. The basic search is very simple and users are required to enter only a keyword to search the database. For the advanced search, I noticed that using the drop down list, radio button and check boxes could make the user interface friendlier. So, **OTMS** will be created with the above features to make it more user-friendly and attractive.

For the browsing function, I can conclude that every system has its own style in presenting the results. However, some of the results look too crowded with words and difficult to read. The page displaying the results is also too plain (no background color), therefore makes the display not interesting at all. Only the ETD of Virginia Polytechnic Institute and State University (United States) is quite impressive in the way it uses light color bars to separate the results. Although this is a simple process, it makes the results outstanding and easier to read. This feature will also be implemented in **OTMS** to make the browsing function fun and exciting.

For both the lecturer and administrator modules, because the lack of current system to be studied on the Internet, it will be design using the experience and knowledge I had from the previous industrial training. These two modules will include the authentication and authorization, title maintenance, compare title, user registration and change password functionalities.

2.4 System Architectures

2.4.1 Client-Server architectures

Client-Server Architectures is a class of software architectures where processing is distributed among one or more information requesters (clients) and one or more information providers (servers), as well as in the interfaces (network, protocols, middleware) between them. Both clients and servers can reside in the same computer or in different computers connected by a network. The figure below illustrates the Client-Server Architecture.

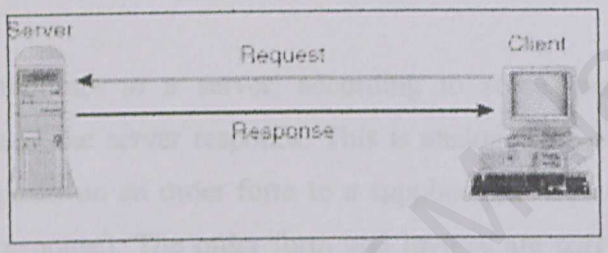


Figure 2.22: Client-Server Architecture

Client-Server architectures include both two-tier, where clients talk directly to servers, and three-tier, where a third software component provides interface services between the clients and the servers. ***Figure 2.23*** shows the three-tier client-server architecture. Three-tier architectures are further categorized by the type of middleware employed. Multi-tier is used to describe architectures with multiple layers or types of middleware. Client-Server architectures are in contrast to a mainframe architecture where the processing is self-contained.

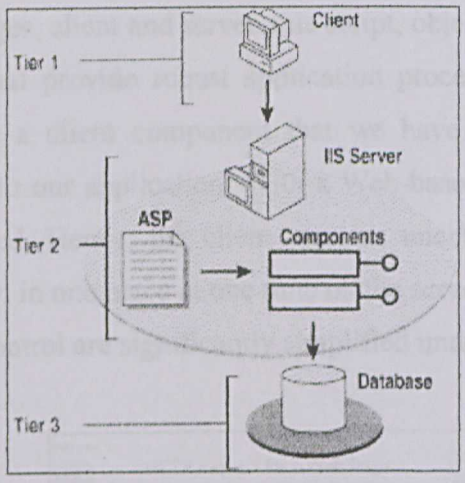


Figure 2.23: The 3-tier Client-Server Architecture

A client sends requests to a server, according to some protocol, asking for information and action, and the server responds. This is analogous to a customer (client) who sends an order (request) on an order form to a supplier (server) who dispatches the goods and the invoice (response). The order form and invoice are part of the "protocol" used to communicate in this case. There may be either one centralized server or several distributed ones. This model allows clients and servers to be placed independently on nodes in a network, possibly on different hardware and operating systems appropriate to their function, for example fast server/cheap client.

Client-server architecture can consists of two modes of application, which are the client-server application and Web application. Essentially, these two modes of computing were the same in that they both used the client-server model. That is, client-server and Web applications both facilitate the client making a request of the server. The main difference is the degree to which they distribute the application logic between the client and the server.

While a client-server application distributes a mix of presentation, application, and database logic between the client and server, Web applications typically parcel almost all of the logic to the server. Using this mix, Web applications can provide the thinnest client of all, where the only component that's installed on the client machine is a browser.

The benefit of Web-based applications hinges on providing a universal client to all users that make requests of a centralized application stored on the server. This application

is comprised of HTML pages, client and server-side script, object-based controls, and other server-side components that provide robust application processing. With a client-server application, we still have a client component that we have to install with every new enhancement and change to our application. With a Web-based application, we just need the browser to be installed. Hence, the client remains unchanged, and the application changes are made centrally, in one place at one time on the server. Administration, software distribution, and version control are significantly simplified under this model.

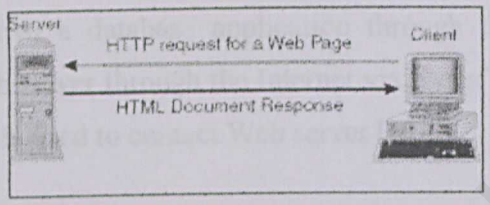


Figure 2.24: Web Applications Using the Client-Server Architecture

Besides Web client and Web server, the Database server is one of the elements that are present in a Web-based application. The architecture of the Web-based system can be seen as below:

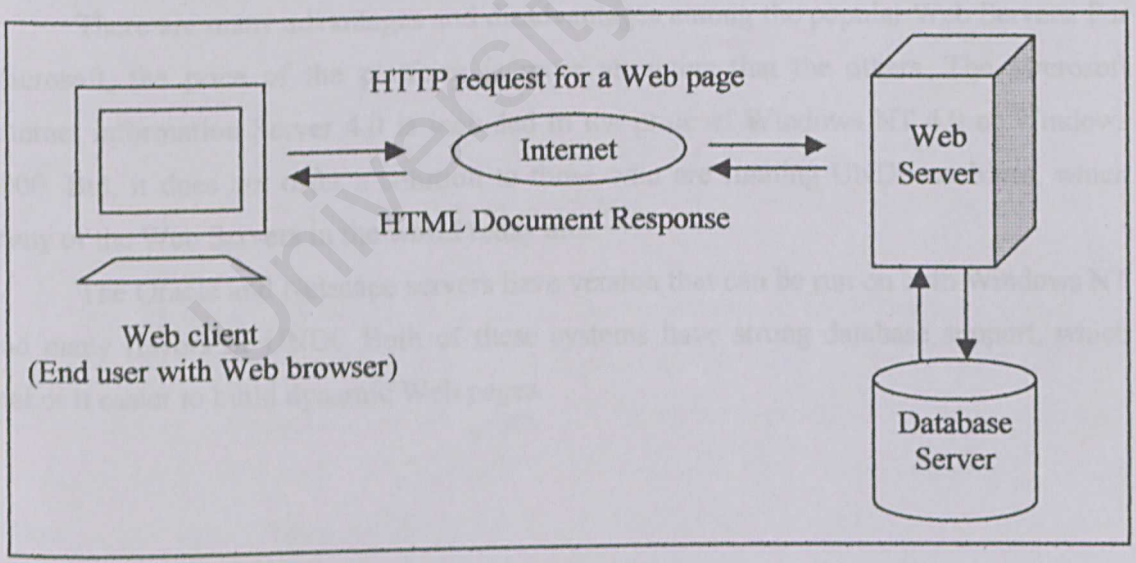


Figure 2.25: Components of Web-based Database Application

2.4.1.1 Web Clients

Currently, there are two giants fighting over market shares, which are Netscape Navigator and Microsoft Internet Explorer (IE) [12]. The job of the Web browser is to contact Web servers, receive HTML pages and then interpret and display those pages. Home pages on an Intranet like Internet are built using a page markup language called HTML (HyperText Markup Language). Once the URL is typed, the Web browser will looks at the URL and determines which server it should contact, which directory to ask for and what is the specific document is wanted in that directory.

An end user can spawn a database application through the Web browser that communications with the Web server through the Internet via HyperText Transfer Protocol (HTTP). In the simple, HTTP is used to contact Web server [17].

2.4.1.2 Web Servers

Web Servers [2, 6] is the software that sends out HTML pages (or whatever pages are called by the HTML code) to the clients. The Web consists of a great many Web Servers scattered all around the net. Each server has different capabilities and capacity. There are 3 majors players in the Web Server field are Microsoft, Oracle and Netscape.

There are many advantages and disadvantages among the popular Web Servers. For Microsoft, the price of the platform is more attractive that the others. The Microsoft Internet information Server 4.0 is included in the price of Windows NT 4.0 or Windows 2000. But, it does not offer a solution to those who are running UNIX machines, which many of the Web Servers in the world today are.

The Oracle and Netscape servers have version that can be run on both Windows NT and many flavors of UNIX. Both of these systems have strong database support, which makes it easier to build dynamic Web pages.

2.5 Internet

The Internet is the largest network in the world. It can become a larger system if it can interconnect all the computer networks that span the globe [25]. It is a three level hierarchy composed of backbone networks (e.g. ARPANET, NSFNET, MILNET), mid-level networks, and stub networks. These include commercial (.com or .co), university (.ac or .edu) and other research networks (.org, .net) and military (.mil) networks and span many different physical networks around the world with various protocols including the Internet Protocol.

Using the Internet, we are allow to communicate with other people throughout the world via electronic mail (email); read online versions of newspapers, magazines, academic journals, and books; join discussion groups on almost any conceivable topic; participate in games and simulations; and obtain free computer software [25].

2.5.1 World Wide Web (WWW)

World Wide Web is a subset of the computers on the Internet that are connected to each other in a specific way that makes those computers and their contents easily accessible to each other [25]. WWW is a system for exploring the Internet by using hyperlinks. It includes an easy-to-use standard interface [25]. This interface makes it possible for people who are not computer experts to use the Web to access a variety of Internet resources [25]. When we use a Web browser, the Web appears as a collection of text, pictures, sounds and digital movies.

The basic idea of WWW is to merge the techniques of computer networking and hypertext into a powerful and easy to use global information system. Hypertext is text with links to further information, on the model of references in a scientific paper or cross-references in a dictionary. With electronic documents, these cross-references can be followed by a mouse-click, and with the World Wide Web, they can be anywhere in the world.

All the pages on the Internet that are familiar through radio, TV and magazines are Web. It is based on Hypertext. Hypertext is a way to organize documents so that our computer can help us find items of interest.

In the illustration, the blue rectangles are electronic documents, similar to pages of paper documents. Inside them, sensitive spots are indicated (the little red rectangles). A computer is used to display the pages on its screen. The sensitive spots are exploited by the computer to switch automatically from one page to another when the user clicks on any of the sensitive spot.

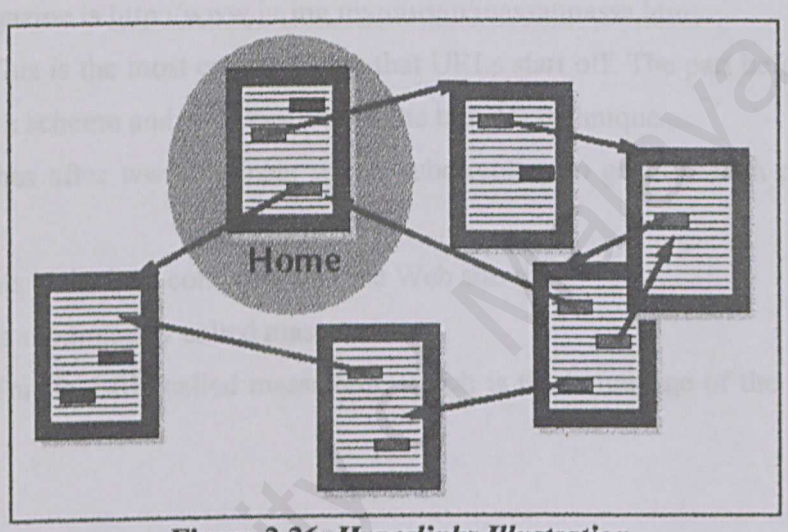


Figure 2.26: Hyperlinks Illustration

This navigation by wandering from one page to another is called "browsing". WWW is "seamless" in the sense that a user can see the whole Web of information as one vast hypertext document. There is no need to know where information is stored, or any details of its format or organization.

The Web is not identical to the Internet. It is one of many Internet-based communication systems. It is important to identify three levels: the physical connections (the cables), the common behavior (the Internet) and the services.

2.5.2 Uniform Resource Locator (URL)

URL is an Internet Web site address. It is a compact representation of the location and access method for a resource available via the Internet. It used to specify Web locations by providing an abstract identification of the resource location. The locations are either departmental Web pages or company wide Web pages. The URL tells the Web browser where to find the information that want to display. To visit a location, simply type in the Intranet location to visit or click on a link to the location.

A URL is broken up into various parts [12]. Example: The Web site of Utusan Malaysia's Massa magazine is http://www.jaring.my/utusan/massa/massa.html

1. http://www - This is the most common way that URLs start off. The part before the colon is called a scheme and specifies http as the transfer technique.
   The back slashes after www indicate which subdirectory to go to (which path to take).
2. /jaring.my - This is the host computer that the Web site is on.
3. /massa - This is subdirectory called massa.
4. massa.html - This is a file called massa.htm, which is the homepage of the *Massa* magazine.

2.5.3 HyperText Transfer Protocol (HTTP)

HyperText Transfer Protocol (HTTP) is a network protocol built on top of transfer control protocol/internet protocol (TCP/IP) that allows for the transfer of data formatted for the Web. It receives a lot of attention today because it is the heart and soul of the Web. This protocol allows end user to easily jump from page to page simply by clicking on the hypertext or link (enter the URL) that tell the browser which page on which server to use next. Beside that, multiple links can exist on a simple document [2].

2.6 Web Browsers

2.6.1 Introduction

Web browsers, also called Internet browsers, are the client application programs that run on almost every available graphical user interface (GUI) computer platforms, including Microsoft Windows 3.X, 95/98/ME, NT, and Apple Macintosh operating systems. Web browsers are multimedia enabled. They can process text, graphics, audio and video, which marked up or embedded in HTML documents. The powerful feature of HTML documents is the function of hyperlinks. The appearance of the document varies from browser to browser depending on the capabilities of each browser, system and preference.

In 1993, a team of student researchers at NCSA released a browser program called Mosaic. This easy to use Web client program, available on PC and Macintosh platforms popularized use of the Internet and the Web. Since Netscape Communications Corporation distributed its commercial Web browsers – Netscape Navigator freely on the Internet, the huge wave of the Internet growth has changed the way we gathered information and the way we do business. Netscape Web browser – Navigators and Communicators were the most popular Web browsers on the Internet but Microsoft Web browser – Internet Explorer is gaining major market share nowadays.

Currently, the most popular Web browsers are Netscape Navigator/Communicator and Microsoft Internet Explorer. The features of browser software have expanded to encompass the ability to access other Internet services in addition to the WWW. The Netscape Navigator and Communicator give a user the ability to read and send email, transfer files (FTP), and read and post Usenet news. Netscape Navigator and Communicator can also be configured to access new media types, such as video, or run other applications within the browser. This is done through Helper Applications or plug-ins.

2.6.2 Comparison between Navigator and IE

Both Netscape Navigator/Communicator and Internet Explorer (IE) are the most popular Web browsers nowadays. Nevertheless they are different Web browsers. Both Netscape and IE can be compared with 10eatures, which are speed, security, platform support, multimedia, mail and news, look and feel, Java, intranet, HTML and groupware.

i. Speed

Speed is one of the criteria that determine the performance of the browser. There are a large number of factors affecting browser speed. Some things, such as response time from the Web server, are beyond the control of the browser. However, the browser can control how efficiently it moves graphics around and how it caches Web pages. Loading pages with both text and graphics into the browser when the browser disk and memory caches were nearly full, Navigator 3.0 was about 35 percent faster than IE 3.0 as shown in the figure below (CNET Networks Inc. 2000). Navigator is even faster when its' cache is emptied.

Navigator loads pages faster than IE	
IE 3.0	
Navigator 3.0	
CNET performance index (longer bars are better).	

Table 2.1: Speed Comparison between Navigator and IE

IE is significantly faster at loading large graphics files. It also appears to be more efficient at managing large images that are already in the memory.

Navigator is noticeably faster at loading most Web pages in a high speed Internet connection. Meanwhile IE showed more speed in displaying pages cached in memory.

ii. Security

Both browsers now provide essentially the same level of security for online transactions, supporting secure sockets layer (SSL) versions 2 and 3 and support site certificates. Both also support personal certificates and digital signatures so that other users can verify the user identity. Perhaps more significantly, they can both now support 128-bit keys for their security codes, providing much stronger security than was possible with the old 40-bit keys. Both products also can warn users about attempts by a Web server to leave cookies on their systems, which can potentially provide personal information to a Web server.

Microsoft has introduced a further level of security with Authenticode, a technology to protect users from potentially malicious downloads of ActiveX controls or Java applets. IE also support for parental controls and rating systems (PICS and RSAC) that Navigator does not have.

iii. Platform support

Netscape can justly lay claim to the cross platform crown. On the day it was released, Netscape Navigator 3.0 supported all flavors of Windows, the Macintosh, and UNIX variants such as Solaris and AIX. More importantly, the browser is essentially identical across all these platforms. However, some features such as the Java just in time compiler are currently only available for Windows 95/98 and Windows NT. Meanwhile Microsoft IE only available on specific platforms such as Macintosh, UNIX and Windows.

iv. Multimedia

Both products support the most popular multimedia data types, such as AVI movies, AU and AIFF sound files. IE has a seamless ActiveX control called Active Movie that handles all of its multimedia files. It also supports MPEG, a video format that allows for much smaller files than AVI or QuickTime. And with its support for streaming, Active Movie can start playback of a file even before the entire file has been downloaded. Navigator has equally broad multimedia support,

but it requires separate plug-ins for each format. Navigator 3.0 comes pre-configured with a couple of the more popular plug-ins, including QuickTime and Live3D VRML.

Both products provide basic support for VRML, which creates 3D worlds through which users can navigate. Netscape's implementation was able to handle a wider variety of files than IE. Netscape also has the further advantage that plug-ins are already available from companies such as Silicon Graphics to support recent drafts of the VRML 2.0 standard.

v. Mail and news

Netscape Navigator 3.0 has mail and news clients in every version. Microsoft's mail and news clients come with every IE 3.0 configuration, except for the minimal install.

The biggest complaint about Netscape's mail and news is that the initial configuration of the clients is not a simple task, and when there is something wrong, all Navigator does to help is pop up an alert box telling that the configuration is in error. Microsoft has a wizard that walks through the process of setting the mail and news servers, online identity, and other chores. Also, Microsoft's mail client offers basic rules and filtering for incoming messages, while Netscape's does no.

While Netscape's mail send and view HTML pages directly in a mail message, the mail editor doesn't support HTML formatting for the body of the message. Microsoft mail has HTML editing build in, but it only allows simple formatting such as bold facing or centering the text. If the mail reader on the receiving side doesn't handle HTML formatting, the receiver will get a plain text representation of the message, without annoying HTML tags embedded inside.

The big news in newsreaders is better offline support. Both Navigator and IE allow download of a group of news messages for later offline reading. More importantly, both can let post replies offline, which can be sent the next time when connect to the Internet. It is easier in IE to download a newsgroup, but Netscape's 3-pane newsreader makes it easier to browse and manage the newsgroup.

vi. Look and feel

IE 3.0 lets users add their favorite links as buttons, and even add an icon to the toolbar that automatically copies the current pages into users' favorite HTML editor. Beside that, IE scrolls pages better than Navigator – IE smoothly slides Web pages up and down, instead of simply redrawing the pages as most applications do. And the icons go from gray to color when the mouse moves over them, which provides great feedbacks.

Consistency is the strong suit of Navigator 3.0. Whether users are in the mail program, the newsreaders, or the browser itself, menus and options remain consistent across the applications. Also, all the 3 components (mail, news and browser) lets users viewed fully formatted HTML pages, something still missing in IE. Navigator lets users turn off the toolbars directly from the menu and whereas IE requires users to first open up an option dialog box and then turn off the toolbar.

IE has more modern and customizable interface, but Navigator is more consistent.

vii. Java

Programming browsers over the Web is the waking dream of the community of software developers. It provides easily distributed computing where the Web server just delivers the code, and the Web client does the heavy lifting of actually executing the program. Thus, a Web site can deliver powerful content without requiring heavy duty CPU time on the server side- the clients will do the work.

That's what has made Java such a success. It enables this kind of Web programming – both the language and the bytecodes (the program instructions generated by the Java compiler) are platform independent. The same Java program can run unchanged on UNIX, Windows or Macintosh machines.

Both Navigator 3.0 and IE 3.0 provide just in time (JIT) Java compilers, which turn the platform independent Java bytecodes into machine specific instructions. For compute-intensive operations such as mathematical calculations, this can greatly enhance the speed of the Java applet. IE 3.0 is up to 15 percent

faster than Navigator for compute-intensive applets. And even on pages where a JIT compiler provides almost no benefits (simple Java applets such as animations and tickers), IE 3.0 still runs Java faster.

Browsers access Java performance tests		
Complex applets		Simple applets
	IE 3.0	
	Navigator 3.0	
CNET performance index (longer bars are better).		

Note: IE is faster on our benchmark tests than Navigator. Our testing was based on the Caffeine Mark from Pendragon Systems

Table 2.2: Java Comparison between Navigator and IE

Naturally, Microsoft had to introduce theirs own scripting language. Luckily, instead of creating some new language, they based this new scripting language on their popular Visual Basic language. Called VBScript, it allows much of the same programmability as JavaScript but leverages the large community of developers familiar with the VB incarnations.

Both JScript and VBScript are essentially plug-in scripting languages. They interact with IE's underlying ActiveX scripting architecture to drive the Web browser.

As a conclusion, IE runs Java better and has a broader scripting support that helps Web developers as well. But Microsoft's IE uneven JavaScript implementation is vexing.

viii. Intranet

The IE 3.0 Web client has a number of advantages for companies that use the Windows 95/98 and Windows NT platforms. For one thing it's free, while shipping versions of Navigator aren't. The administration kit, which lets the IS

department configure virtually every feature of the browser is also free for IE, but the similar kit from Netscape costs nearly RM 2000.

More importantly, IE allows users to create intranets with hook into their desktop business applications. For example, when a user clicks on a link to a word document, IE will open word inside the browser window. It's an example of in-place activation, and it's a powerful technology. Add to the dynamic nature of ActiveX controls, and users intranet becomes the medium for automatically keeping components consistent and up to date across the network.

Netscape Navigator 3.0 simply doesn't support this sort of tight operating system integration, although users can get part of the way there with specific file-viewer plug-ins as well as with Ncompass's ActiveX plug-in. but the ability to merge Web browsing and application functions is still missing. However, better integration is due in Navigator 4.0.

Integration of users desktop applications with the intranet isn't quite seamless yet, but it is much better in IE than it is in Navigator.

ix. HTML

HTML is the basic page description language for the Internet. HTML should render essentially the same way from one browser to the next. Unfortunately, both Netscape and Microsoft have in the past, pushed their own proprietary extensions to the HTML specifications. This has made for tough choices for Web designers to support one browser's set of tags, and users page might be unreadable in the other browser.

In their latest browsers, Navigator and IE continue pushing HTML. IE 3.0 supports a few new tags, such as ones that let designers create floating and borderless frames. Version 3.0 of Navigator has a few new tags too, such as Spacer, which allows designers to specify exact horizontal and vertical spaces.

x. Groupware

Currently, the only collaboration tools that ships with the two browsers are essentially Web phones: Netscape's CoolTalk and Microsoft's NetMeeting. Both let users use the Internet as a telephone, as long as the contact person has the same software. The recently ratified H.323 standard, which both companies have committed to support, will eventually allow Web phones from different companies to inter-operate. Both products support chat, where users can write text messages to each other in real time. And both support a whiteboard feature where users can collaborate and brainstorm.

However, Microsoft's NetMeeting takes the next logical step by letting users share entire Windows applications via the whiteboard; multiple users can watch someone editing a word document or a spreadsheet. And NetMeeting allows each user in a conference to take over the shared application and add their contributions. It's a neat collaboration trod, and since it involves sharing just the user interface of an application, it doesn't require that all the users install the application being shared. Of course, features such as applications sharing work best over fast Internet connections. Powerful application sharing makes NetMeeting a more superior groupware tool than CoolTalk.

xi. Extensibility

Netscape's plug-in architecture was one of the crowning achievements of Navigator 2.0. It allowed developers to extend the file and media types that navigator could handle, bringing streaming audio, VRML, and a host of other capabilities to the Web. Several of the most popular plug-ins, including QuickTime and Live3D VRML, now come bundled with Navigator 3.0.

ActiveX controls are Microsoft's answer to plug-ins. The main thing going for ActiveX controls is that they are Web based on Microsoft's mature Object Linking and Embedding (OLE) technology, which means there is a wide variety of Windows programming tools that already, or will soon, support ActiveX development. The other key feature is that ActiveX controls will work in any

Windows program that support ActiveX architecture. Thus, users can install one ActiveX control that would provide similar functionality in their Web browser, their word processor or their graphics editor.

From the user's point of view, the most compelling feature of ActiveX controls is that they can be downloaded and installed automatically. In contrast, a Netscape plug-in has to be downloaded and the browser shut down before the plug-in can be install. Furthermore, while adding ActiveX capability, Microsoft has not forgotten the installed plug-in base. Most existing plug-ins work just fine in IE 3.0. If users installed a plug-in for Navigator, IE will detect and use the plug-in as if that plug-in had been installed in IE. However, Microsoft does not support Netscape's Live Connect architecture, so any Live Connect plug-ins will not work in IE.

This blade cuts two ways: Navigator users can also use ActiveX controls via free Ncompass plug-ins. Navigator's ActiveX compatibility is not quite as good as Microsoft's plug-in support, but it does demonstrate that ActiveX controls can run in Navigator.

Both products have great support for browser extensions: Navigator through plug-ins and IE through ActiveX. Both also offer partial support for competing extension architectures.

2.7 Database

2.7.1 Why use Web database applications

In most Web-based applications, database serve as the basic building block for information services. Organizations might want to use this database in their Web applications for the following benefits:

i. Consolidation of data

Users may want to unlock the potential unused information in organization database. Information from database in various part of an organization (for example finance department, human resource department, project management department and so on) can be consolidated using Web-based applications and served to the users as though it were from a simple source. Databases do not have to be physically located in each department of an organization. On the other hand, one database can be shared for the whole organization.

ii. Better manageability

The evolution from document based information repositories to paperless and efficient data storage in Web database has save a lot of energy, time and cost in managing records. Records can now be trace easily without any problems and maintaining records will be hassle free job.

iii. Extending the functionality

The functionality of the Web server can be extended so that the information contained within can be made available to the general public or other internal users. This is a task that is currently undertaken by many organizations and government agencies where primary product is information.

2.7.2 Design considerations via Web database applications

Designing a Web-based database is very important prior to the advent of the Internet. However, there are several issues that must be put into consideration when designing the Web-based databases. The following is a brief list of issues relating to that:

i. Security

Secure communications and user identities are critically important for protecting the data in the Web database from loss, corrupt and unauthorized use. Therefore, we must consider various Web technologies and develop a security system based on some specific scenarios.

ii. Performance

Internet users will not want to wait for their queries to be processed. Any queries that take more than a couple of seconds will probably frustrate the users. That's too bad because he or she might not even want to use the Web site again in the future. Therefore, we may need to use a denormalized design to achieve acceptable performance.

iii. Backup

The Internet is available 24 hours a day and 7 days a week. The Web site must be continuously available all the time. This can complicate our backup strategy, which can impact our database design. If we choose to denormalize our design, we might increase the physical size of our database and this can increase the time frame required to back up the database. The longer it takes to back up the system, the longer the database may be unavailable or unresponsive and the users will not be able to access it.

iv. Language

The Internet is worldwide, which means that our Web site may be viewed in foreign countries. Do we need to support multiple languages? If so, this can impact our

database design because we need to track data in different languages. Beside that, our database physical size will also increase and become more costly and difficult to maintain.

2.7.3 Microsoft Access 2000

Microsoft Access 2000 is the relational database management system used to create the relational database. Together with the ODBC driver for Access, data can be retrieve from the database in client-server based system. It contain facilities to import, export and link tables to HTML files, to process hyperlinks into or out of database applications and to publish tables, forms and reports in HTML format. In addition, forms and datasheets can be saved as Active Server Pages.

Microsoft Access is a powerful database management system, which collects information related to a particular subject or purpose, such as tracking customers' orders or maintaining a book collection. By using Microsoft Access, we can manage all the information from a single database file. And within the file, the data can be divided into separate storage containers called tables. Besides storing information, Microsoft Access provides extensive new features designed in development of Web Enabled application.

Microsoft Access 2000 provides powerful new tools for sharing, managing and manipulating data. It shares our database with coworkers over an Intranet, find and retrieve information quickly, and take advantage of automated, pre-packaged solutions to quickly create databases.

However, Microsoft Access is not so suitable for Web-based application since Access do not have the speed to support more than a few hundred users over the network. Besides, it is limited in the size of each record and each table.

2.7.3.1 Advantages of Microsoft Access

Microsoft Access is widely distributed, securely tested and proven in the market place and in use by many users. Data integrity, speed, Y2K compatibility, integration with Windows and other programs, these have already been built and tested by Microsoft. With Microsoft Access software, reliability is guaranteed from the start.

Every couple of years, Microsoft upgrades its Office suite of business software, including Access. Following Access 2.0, Access 95 and Access 97, 1999 saw the release of Access 2000. With each upgrade, powerful new features become available, ensuring that software developed in Microsoft Access stays at the leading edge of technical development. Here are some of the advantages of Access 2000:

i. Integration with other software

Microsoft Access is closely integrated with the other members of the Office suite of business software. As far as accounting software is concerned, the link with the suite's spreadsheet program, Excel, is particularly useful.

ii. Microsoft Access is fully networkable

Microsoft Access is designed for both stand-alone and multi-user applications. The program is fully networkable.

iii. Microsoft Access is Y2K compliant

As might be expected, both Microsoft Access 97 and Microsoft Access 2000 are fully year 2000 compliant.

2.7.4 Microsoft SQL Server

Microsoft SQL Server (MSSQL) is a multi-user relational database management system (DBMS) that runs on the Microsoft Windows NT operating system. The SQL Server driver enables applications to access data in Microsoft SQL Server databases through the Open Database Connectivity (ODBC) interface. Structured Query Language (SQL) is used to access data in a SQL Server database. All the client workstations communicate with SQL Server across a network, such as a Window NT Server, Novell, TCP/IP network and etc. [15]

2.7.4.1 Benefits of MSSQL

- ***Ease of installation, deployment and use***

  SQL Server includes a set of administrative and development tools that improve our ability to install, deploy, manage, and use SQL Server across several sites.

- ***Scalability***

  The same database engine can be used across platform ranging from laptop computers running Microsoft Windows 95/98 to large, multiprocessor servers running Microsoft Windows NT or Windows 2000.

- ***Data warehousing***

  SQL Server includes tools for extracting and analyzing summary data for online analytical processing (OLAP). It also includes tools for visually designing databases and analyzing data using English-based questions.

- ***System integration with other server software***

  SQL Server integrates with email, the Internet, and Windows.

2.7.5 Oracle DBMS

2.7.5.1 Introduction

Oracle Corporation's reputation as a database company is firmly established in its full featured, high performance RDBMS server. With the database as the cornerstone of its product line, Oracle has evolved into more than just a database company, complementing its RDBMS server with a rich offering of well-integrated products that are designed specifically for distributed processing and client-server applications.

As Oracle's database server has evolved to support large scale enterprise system for transaction processing and decision support, so too have its other products, to the extent that Oracle can provide a complete solution for client-server application development and deployment. An overview of client-server database system and the Oracle product architectures that support their implementation are listed below.

2.7.5.2 Oracle 7 RDBMS Server

The Oracle 7 server is a full-featured RDBMS that is ideally suited to support sophisticated client-server environment. Many features of the Oracle 7 internal architecture are designed to provide high availability, maximum throughput, security and efficient use of its host's resources. Although all these features are important architecturally for a database server, Oracle 7 also contains the following language-based features that accelerate development and improve the performance of server side application components.

i. PL/SQL language

A major component of the Oracle 7 server is its PL/SQL (Procedural Language/Structured Query Language) processing. PL/SQL is Oracle's fourth generation language (4GL) that incorporates structured procedural language elements with the structured query language. PL/SQL is designed specifically for client-server processing in that it enables a PL/SQL program block containing

application logic as well as SQL statements to be submitted to the server with a single request. PL/SQL significantly reduce the amount of processing required by the client portion of an application and the network traffic required to execute the logic. PL/SQL is extremely flexible because the language used by client is interchangeable with that used by the server.

ii. Stored procedures

Although version 6 of Oracle supported server based PL/SQL, Oracle 7 provides the capability to store PL/SQL blocks as database objects in the form of stored procedures, functions and database packages. Calling a stored procedure from a client application generates minimal network traffic. Stored procedures provide a convenient and effective security mechanism.

iii. Database triggers

Database triggers resemble stored procedures in that they are database resident PL/SQL blocks. The difference between he two is that triggers are fired automatically by the RDBMS kernel in response to a commit time event (such as an insert, update or delete operation). Database triggers greatly reduce the amount of codes and processing that is necessary in the client portion of an application.

iv. Declarative integrity

Constraints are enforced by the server whenever records are inserted, updated or deleted. In addition to using referential integrity constraints that enforce primary and foreign key relationships, defined constraints can be used to control the value domains of individual columns within a table. Server enforced integrity reduces some of the codes required for validation by the client and also increases the robustness of the business model defined within the database. With constraints, we can often improve performance and provide the flexibility to support multiple front-end interfaces.

v. User defined functions

User defined functions are stored as PL/SQL blocks. User defined functions are similar to stored procedures and also reduce the amount of application codes in the client portion of an application. Not only these functions from PL/SQL, but it can also extend the set of standard Oracle SQL functions.

2.8 Programming Languages

2.8.1 CGI and PERL

CGI stands for Common Gateway Interface and it refers to an HTML standard for defining forms and sending or receiving form's data. PERL, which is Practical Extraction and Report Language, is a programming language invented by Larry Wall to create application programs that run on network servers. It is used not only because it is an effective programming language but also it is free and portable for UNIX, Windows NT and Macintosh operating systems. Libraries are also available for processing ODBC from PERL.

Any programming language can use CGI. The server programs can be written in Visual Basic, C++ and Java. It is simple to learn, use and standardize. CGI is essential for processing forms, looking up records in the database and generating dynamic HTML documents. Without CGI, a Web server can offer only static documents and links to other pages or servers.

2.8.2 VBScript, JavaScript and JScript

VBScript and JavaScript or Jscript are two lightweight programming languages that can be used to place programming logics into HTML pages. Both are interpreted and both reside within the page. Using script, Web page developers can control or script both ActiveX controls and generic Java applets. Script compatible browser is an essential need to request a scripting file from the Web servers. The ActiveX-scripting interface let users plug in other scripting engines.

2.8.2.1 JavaScript or JScript

JavaScript or JScript is Netscape's cross-platform, object-based scripting language for client and server applications. JavaScript allows the applications that run over the Internet been created, which the client applications run in a browser and server applications run in a server.

JavaScript is lightweight in that there isn't a great deal to learn. So, we can be productive with it very quickly, in contrast to much more complex languages such as Java. As a scripting language, JavaScript is meant to tell an application what to do. Unlike languages used to create applications, it cannot do anything without the application.

Using JavaScript, we can create dynamic HTML pages that process user input and maintain persistent data using special objects, files, and relational databases. A JavaScript page can validate the data entered before it is sent to the server. If the data is invalid, JavaScript can block transmission to the server. Because all of this work is performed on the client side, JavaScript does not waste bandwidth transmitting bad data and then receiving an error page from the server.

Beside that, JavaScript can provide a high degree of user interaction like some other systems, including CGI and Java.

2.8.2.2 VBScript

VBScript [4, 13], as the name implies, is another scripting language that was designed by Microsoft based on Visual Basic programming language. VBScript is a fast, portable, lightweight interpreter for the use in World Wide Web browser and other applications that are Microsoft ActiveX controls, automation servers, and Java applets. VBScript can be embedded in the HTML pages to build the Web applications.

VBScript is designed for use with Microsoft's Internet Explorer browser together with other programming that can be run at the client site, including ActiveX controls, automation servers, and Java applets. Client-side scripting refers to scripts that can be executed in the users' Web browser, the Web client, rather that the Web server.

Unlike Java, VBScript and JavaScript code is represented as regular ASCII text within the HTML document. The VBScript code is interpreted and compiled while the browser downloading it from a Web server. Besides, the VBScript code is extremely easy to learn and use compare to Java and also JavaScript.

2.8.3 ActiveX

ActiveX is architecture of programs elements promulgated by Microsoft. It is an architecture that can be implemented with any number of different languages. For example, it can be written in Java, C++ or Visual Basic. The goal of all these technologies is to enable programs to interact with one another in a dynamic fashion. They do it by using an interface called IUnknown. Query interface can determine what objects, methods and properties a program support.

2.8.4 HyperText Markup Language (HTML)

HyperText Markup Language (HTML) is a language that includes a set of codes (or tags) attached to text. These codes describe the relationship among text element. For example, HTML includes tags that indicate which text is part of a header element, which text is part of a paragraph element, and which is a part of a numbered list element. One important type of tag is the hypertext link tag. A hypertext link, or hyperlink, points to another location in the same or another HTML document [25].

HTML [4] is known as a scripting language. The language doesn't compile and become executable as Visual Basic programs do. Instead, HTML formats Web pages, specifies where graphics and dividing frames go, and allow for embedded activated application such as ActiveX documents and Java programs. In the simple, HTML is the formatting language behind Web pages that formats Web pages to look the way Web pages do.

2.8.5 Active Server Pages (ASP)

ASP [27] stands for Active Server Pages, that is a program runs inside Internet Information Server (IIS), which is a part of the Windows NT 4.0 Option Pack. ASP provides an extensive server-side platform supporting compile-free, language-independent scripts and ActiveX components.

ASP file is just the same as an HTML file. It contains text, HTML tags and scripts. The script in an ASP file is executed on the server and the ASP file has the file extension ".asp".

When the browser requests an ASP file, the server calls ASP. ASP reads the ASP file and executes the scripts in the file. Finally, the ASP file is returned to the browser as a plain HTML file.

2.8.5.1 Advantages of ASP

- Dynamically edit, change or add any content of a Web page
- Response to user queries or data submitted from HTML forms
- Access any databases and return the results to a browser
- Customize a Web page to make it more useful for individual users
- Simplicity and speed
- Provides security since the ASP code cannot be viewed from the browser
- Since ASP files are returned as plain HTML, they can be viewed in any browser
- Clever ASP programming can minimize the network traffic

2.8.5.2 Comparison between ASP and other programming languages

2.8.5.2.1 ASP compare to CGI application

ASP provides all of the functionality of CGI applications in an easier-to-use and more robust environment. ASP is an easier way for server to access information in a form not readable by client (such as on SQL database) and then act as a gateway between the two to produce information that the client can view and use.

With CGI, the server creates as many processes as the number of client requests received. The more concurrent requests there are, the more concurrent processes created by the server. However, creating a process for every request is time consuming and requires large amount of server RAM. In addition, this can restrict the resources available for sharing from the server application itself; slowing down performance, and increasing wait times on the Web. ASP instead runs in the same process as the web server, more handling

client request faster and more efficient. It is much easier to develop dynamic content and Web application with ASP.

2.8.5.2.2 ASP compare to ISAPI applications

ISAPI applications require all of the programming and layout to contain in a DLL file written in C++. ISAPI applications are thus more difficult to create and maintain. With ASP files, and HTML Writer can script an external component and format the output. ASP separates the layout and design from the business logic.

2.8.5.2.3 ASP compare to PERL

PERL is the most popular programming language for writing CGI scripts. Due to its popularity, power and flexibility, as well as the fact that scripts written in PERL can run easily on most Web servers. PERL is the language that we teach for CGI scripting. Anyway, PERL and other scripting languages are not robust development tools by themselves. ASP provides a familiar framework and objects for building complex applications that require data from relational database and legacy sources. ASP supports virtually any scripting language to build these applications. Third parties are currently developing additional scripting engines, such as PERL, which will be announced when they are ready.

2.9 Programming Tools

2.9.1 Microsoft Visual InterDev 6.0

The World Wide Web (WWW) has made the Internet come alive for many new users. In the initial stages, users realized that they could view numerous documents on various topics from all over the world. Through the WWW, people have access to a plethora of knowledge. Not only can we read about many interesting topics on the Web, but we also can find information about our favorite products and services.

Once we have gained this knowledge, we will invariably want to act on it. Some companies have built Web-based applications, enabling us to buy their products and services electronically over the Internet. Other companies enable us to fill out registration information to begin receiving certain services. The point is applications enable the user to act on the knowledge they have gained. Businesses can capitalize on opportunities sooner by becoming closer to the customer through a virtual marketplace. Visual InterDev provides all the necessary tools to build these vital applications for the Web.

Visual InterDev is a comprehensive, Web-based application development tool. Visual InterDev provides an integrated environment that brings together various technologies to work towards a common goal of building robust and dynamic applications for the Web. Visual InterDev achieves this integrated development environment through the use of the Developer Studio shell interface, first used in Microsoft's Visual C++. We can open and work on Visual C++ and Visual J++ projects while simultaneously creating our Visual InterDev project. This feature greatly enhances productivity, especially when we are building COM and DCOM components and incorporating these components into our Visual InterDev application.

Visual InterDev enables the developer to build applications that are dynamic and interactive through the use of client and server-side script. VBScript is the default scripting language, but JavaScript also can be used for the same purpose.

Database integration is vital to any application. Visual InterDev provides a rich and robust set of visual database tools to immediately enhance our productivity. Visual

InterDev supports the major ODBC-compliant databases, both on the desktop and the server.

Managing our Web site once it has been developed is a very crucial function. Visual InterDev provides a set of tools to view and maintain our site. These tools are similar and compatible with the site management tools found in Microsoft FrontPage.

Visual InterDev supports the major object-based technologies that exist for developing Web-based applications, including ActiveX controls and Java applets. Visual InterDev supports the use of third-party ActiveX controls and enables us to integrate our own custom ActiveX controls. Visual InterDev also provides Design-time Controls that enable us to set control properties when we are designing our application and then use this functionality at runtime without the overhead of a typical ActiveX control.

Visual InterDev provides a robust, integrated development environment to address the many capabilities of the Web. We can integrate various technologies, like ActiveX controls and Active Server Components, to create a powerful application. The integrated development environment enables us to use scripting languages like VBScript and JavaScript to create dynamic applications and Web pages. We can even work on simultaneous projects of different types all from within Visual InterDev's Developer Studio interface.

In a nutshell, Visual InterDev is an exciting new tool that significantly augments a Web developer's productivity.

2.9.1.1 Visual InterDev Features

Visual InterDev includes a wide range of visual tools to augment an application developer's productivity. HTML editing is significantly enhanced through the use of the HTML Layout Editor and a version of the FrontPage HTML editor. The HTML Layout Editor, introduced with the ActiveX Control Pad from Microsoft, enables us to precisely place our ActiveX controls onto our Web page. It also enables us to control the x and y coordinates to ensure that the ActiveX controls are displayed in the proper manner. By

using this tool, we are able to take more control of our user interface, and can ensure that the interface we build is an effective one.

Visual InterDev also provides "What You See Is What You Get" (WYSIWYG) editing through its own FrontPage HTML editor. FrontPage enables us to visually author our HTML page. Content authors can use FrontPage 2000 to create files that are completely compatible with Visual InterDev's version of the FrontPage Editor. Visual InterDev also provides site management tools that are very similar to those tools provided in FrontPage 2000.

The Script Wizard is another visual tool that greatly enhances a developer's productivity. The Script Wizard enables us to associate specific actions with associated ActiveX control events. By linking these events and actions, the Script Wizard generates all of the necessary script language for us. Once the language is generated, we have the ability to modify and add to the code. This process can save us considerable development time by generating the routine script and enabling us to focus on the more advanced code for our application.

Visual InterDev also includes two tools for spicing up our Web pages. These tools focus on multimedia creation and management. The Microsoft Image Composer and Microsoft Music Producer enable us to create graphical images, music, and sounds for our Web site. We can use the Microsoft Image Composer to create engaging images for our Web pages. The Image Composer supports the Adobe Photoshop file format as well as GIF and JPG formats. The Image Composer is simple to use, and we don't have to be a graphic artist to master it. The Music Producer enables us to create music and sound effects for our Web site. We can pick from over 100 pre-defined styles of music and can modify the arrangement of instruments as well as the tempo.

Again, the goal of both of these products is to provide a rich and rewarding experience for the user. The Media Manager enables us to manage all of our multimedia files through the use of specialized folders. By using Media Manager, we can properly organize our images, sounds, video clips, and other multimedia files.

Now consider some of the key features for building robust server applications. Visual InterDev enables us to create dynamic Web pages through the use of Active Server

Pages. The concept of Active Server Pages was touched on earlier in the day. As a refresher, Active Server Pages are HTML pages that contain server-side script. The script code can be either VBScript, JavaScript, or some other scripting language. The Active Server Page, or .asp file, resides on the server machine and executes before being downloaded to the browser.

Active Server Components are a significant part of building distributed and powerful applications. Active Server Components are programs, DLLs, or executables (EXEs), which are built using the Component Object Model (COM) specification. Visual Basic, Visual C++, and Visual J++ all support the development of COM-based components. These programs can be called from Active Server Pages to provide robust application processing on the server machine.

For example, we might want to build an Active Server Component that uses the strength of the C++ language to perform financial analysis and return the results to the Web browser. We also can distribute the application processing load through the use of Distributed COM (DCOM). Active Server Components provide a method for building high transaction processing capability into our application. Visual InterDev provides an environment that is conducive to incorporating these components into our applications.

2.9.1.2 Database Integration

Visual InterDev provides some very robust database tools. The Visual Data Tools included with Visual InterDev are easy to use and significantly reduce the time and effort for adding database capability into our application. Some of the features include the following:

- ***Data View:*** Visual InterDev project window that enables us to view our entire database objects including tables, views, stored procedures, and triggers.
- ***Query Designer:*** A tool that enables us to visually build our SQL database queries and test the results.

- ***Database Designer:*** A tool that enables us to design, creates, and maintains our SQL database.
- ***Stored Procedure/Trigger Editor:*** A tool for editing stored procedures and triggers for Microsoft SQL Server 7.x and Oracle 7.x.

The Query Designer and the Database Designer employ a user interface similar to Microsoft Access that's very easy to use. We can drag and drop objects into the workspace and automatically build our SQL queries. We also can use the Database Designer to populate the database and modify existing data.

Other database features include the following:

- ***ActiveX Data Objects:*** An object-based approach to accessing a database over the Web that uses ActiveX Scripting.
- ***Database Design Time Controls:*** Controls geared toward the automatic creation of database connectivity and generation of Active Server scripting.
- ***Database Wizards:*** Wizards that guide us through the creation of an HTML, data-bound form

Database connectivity and integration is one of the best features of Visual InterDev.

2.9.1.3 Browser Integration

Another feature of Visual InterDev is the integration of a browser within the development environment. Visual InterDev includes an implementation of the Microsoft Internet Explorer 4.0 browser. This implementation supports all of the same features of the commercial version of Internet Explorer, including Java applets, ActiveX controls, ActiveX documents, VBScript and JavaScript, style sheets, and HTML 3.x features such as frames and tables. By using the Preview in Browser feature, we don't have to use one tool to build the application and then open up our browser to view it. With Visual InterDev, we can build our Web-based application and view the results all from within one environment. This

feature adds to the speed with which a developer can create and update applications and Web sites.

We do have the capability to use the Browse With function to view our Web site from within the browser's native window. For example, we could configure Internet Explorer or Netscape Navigator and view the site from within the production browser environment. We probably will want to use this feature during final testing of our Web site. If we are deploying our site for use with the Netscape Navigator browser, we have to use this feature.

2.9.2 Microsoft Visual Basic 6.0

Microsoft Visual Basic (VB) 6.0 has been incorporated with a number of new and enhanced features that make it more powerful and easier to use than previous versions. The features are shown in the table below:

- ***Native code compile***

  VB 6.0 has the capability to compile a program to native code, much as C++. Therefore this will give a faster program. However, Visual Basic runtime library file is still needed to provide a fully functional program.

- ***New database features***

  Visual Data Manager is the new feature in VB 6. It eases the maintenance of database structure, as well as to input and edits the actual data. Additionally, the Visual Data Manager helps to create, test and save SQL statements in a program.

- ***Internet features***

  VB 6.0 includes ActiveX controls and Web browser control to help Internet developers or programmers.

- ***Others***

  Other features are such as the enhancement to Code Editor and Development Environment.

Productivity	Performance	Data Access	Internet
Enhanced IDE with IntelliSense and an extensive set of wizards	Native code with faster loading forms and controls	Microsoft SQL Server 7.0	Browser based application development and Migration Wizard
Visual ActiveX control creation	Microsoft Transaction Server	Enhanced RDO 2.0 with User Connection Designer	Visual ActiveX control creation
IDE extensibility for smarter add-ins and customizations	Application and database performance tools	Enhanced Oracle and Microsoft SQL Server drivers	Web browser, HTTP/FTP, and Winsock Internet controls
Enhanced controls and tools	Integrated DCOM support	Inline SQL Server stored procedure debugging	Setup Wizard for Internet deployment

Table 2.3: Features of Microsoft Visual Basic 6.0

2.10 System Platforms

2.10.1 Windows 2000

Windows 2000 is the operating system for server developed by Microsoft. The interface of the software is similar to that of Windows 95/98. It's gaining market share due to the ease of use.

2.10.1.1 Advantages of Windows 2000

Windows 2000 is developed and sold by Microsoft. The company continually provides supports and updates. Users who are familiar with Windows will find themselves comfortably using Windows 2000. Most people using Windows 2000 hosting utilize Active Server Pages (ASP) technology. This is the main reason why people are turning to Windows 2000 hosting. Users can develop Web site using familiar interface of Microsoft tools such as Microsoft FrontPage, Visual Interdev, and Microsoft Access. With ASP users can develop a database-driven Web site using Microsoft Access or Microsoft SQL as the database.

Besides, Windows 2000 is used for businesses because it has a high level of stability and security. Then, it uses full 32-bit nativity and removes the need for DOS. It also uses NTFS Secure file allocation table which is very robust. Although the system is a little bit slow, it is very good in network connectivity. It steeps the hardware resources needed to run it and has the ease of use of Windows 95/98. It also improved the multimedia support, interfaces and stability of the overall system. It has less system crashes and higher connectivity to plug and play hardware especially for Universal Serial Bus (USB) devices.

2.10.1.2 Disadvantages of Windows 2000

Unlike Unix, Windows 2000 requires more system resources. The users need a powerful machine to run Windows 2000. The Windows 2000 server also needs reboot more frequent than Unix. If we are using Windows 2000 hosting services, we may find ourselves investing a lot of money in the development tools. Most of them are Microsoft products. The costs of applications that can run on our Web site are usually higher than that of Unix. For example, we can find a lot of free scripts to run Web board, chat room, Web stats, email, and more for the Unix-based Web site, but we won't find many free applications in 2000 world.

2.10.2 Windows 98

Windows 98's predecessor, which is Window 95 has 32-bit and 16-bit code nativity. It has embedded MS-DOS within the operating system (OS). It has a graphical user interface and a satisfactory array of driver support for some hardware. Besides, it has better memory handling compared to Windows 3.1. Finally, Windows 95 comes with better looking and is simpler to install and use.

On the other hand, Windows 98 has an even better driver support than Windows 95. It includes many of the Service Packs that Windows 95 needed. It also has the improved multimedia and DirectX support. Eventually, it also integrates Internet Explorer into the graphical user interfaces (GUI). It has higher stability and includes USB functionality, which cannot be found on the Windows 95.

2.10.2.1 Advantages of Windows 98

Windows 98 is an excellent operating system. Although it was introduced in the spring of 1998, some papers made the comment "it is not worth the money" and "there is not much new compared to Windows 95", but it still has its' own advantages to make it outstanding from Windows 95. Here are some of the advantages:

i. Generally better performance

Generally Windows 98 utilizes the PC resources better:

- Memory management has been completely changed. It finally works in producing a stable operating system.
- The file system is better integrated into the operating system, which gives new functionalities.
- Program loading can be up to four times as fast compared to Windows 95.
- Hardware support is significantly improved with a new driver model, and all new chipsets etc. are supported.

These improvements are sufficiently significant to justify an upgrade.

ii. Better user interface

Superficially the Windows 95 user interface has not changed much. But we need not dig down very deep to see many novelties. Here is talking about small items with better adaptations in the Start menu, new tool bars, etc. But these small items are really very smart when we need to set the user interface. This is all thoroughly described in "Windows 98 - Short Course" at http://www.karbo.dk/pctutor/w98-lds.htm, and those are small modifications with significant effects.

iii. **Better system tools**

The system tools are significant for the more demanding user, who really wants to know and be in command of the PC. A number of new tools have been added to Windows 98. They improve surveillance facilities.

iv. **Stability**

Many will experience that previous instability in Windows 95 has just gone. The PC can be left on for weeks on end without going down a single time. Many may laugh at this – "Why should we pay to correct Windows 95 errors?" That may be true, but consider the wasted time with PC's, which fail and need to be restarted, etc.

2.11 Web Server Software

2.11.1 Personal Web Server

Microsoft Personal Web Server (PWS) is a desktop Web server for Windows NT Workstation, Macintosh, Windows 95 and Windows 98 client computers. It enables developers to publish their personal home page, share documents with certain workgroup or build Web applications for Windows NT Server's built in Web Server, which is Internet Information Server (IIS). Owner of the Web server can use the site administrator tools to read messages, browse and retrieve files from user's disk. Microsoft Personal Web Server will automatically convert simple text documents.

2.11.1.1 Advantages of Personal Web Server

1. ***Provides a good test drive for IIS Web site***

   Personal Web Server on Microsoft Windows NT Workstation is also ideal for developing Web applications for IIS. Personal Web Server on Windows NT Workstation includes support for features such as ASP, script debugging and the Internet Service Manager (the comprehensive administration tool for IIS integrated into Microsoft Management Console). Transactional Web applications for Microsoft Transaction Server (MTS), also part of Windows NT Server, can be developed. Personal Web Server is a great platform for testing before hosting the site on the company server, or on an Internet Service Provider (ISP). Links, forms, scripts and applications can be tested and checked to be sure they are displaying and functioning correctly. Microsoft FrontPage also can be used to easily copy a Web site developed on Personal Web Server to IIS.

2. ***Provides wizards***

   Personal Web Server includes a wizard that walks developer through setting up a home page and sharing files. The Personal Web Server administrator also reduces the complexity of running a Web server. Personal Web Manager enables the

developer to start and stop the server, view statistics, and easily share additional directories or the Windows Explorer.

2.11.1.2 Disadvantages of Personal Web Server

1. ***Does not support high volume Web sites***

   Because Personal Web Server was not design to support high volume Web sites, it does not include all the features found in IIS, such as Microsoft Site Server Express, Index Server and Certificate Server.

2.11.2 Apache 1.3

Apache 1.3 is UNIX Web Server and available free of charge. It also can run in Windows NT. Apache can be obtained from the Internet and all the core and module source code can be get and modified to suit developers' needs. Hosting multiple IP addresses on an Apache Server is done with little configurations.

2.11.3 Netscape Enterprise Server (NES)

It is produced by Netscape Web Server. It supports Oracle, Informix, Java, LDAP and use to convert pdf files to HTML files. Netscape Enterprise Server can run either in Windows NT Server or UNIX. It has a direct link to a DBMS automatic directory tree.

2.11.4 Microsoft Internet Information Server (IIS)

Microsoft IIS is built into the Microsoft Windows 2000 Server operating system. It was designed to deliver a wide range of intranets and the Internet server capabilities for corporate. IIS can be used alone as a Web server, or in conjunction with compatible

technologies to set up Internet commerce, to access and manipulate data from a variety of data sources, and to build Web applications that take advantages of server script and component code to deliver client-server functionality.

2.11.4.1 Advantages of IIS

1. ***Enables to generate dynamic Web pages with HTML templates***

   IIS 4.0 enables creating HTML page templates with ASP to build pages dynamically and make sites easier to keep up to date.

2. ***Provides an easy information searching on the Web***

   IIS 4.0 introduces updated indexing and searching functionality to the build in content indexing and search engine for IIS. This allows users to search for content on the server using a standard Web browser.

3. ***Provides the facilities to capture specific user information***

   IIS 4.0 provides three new logging features that enable information retrieve from the IIS 4.o Web sites to be customized easily. The features are as follow:

   i. Specify log file information – with IIS 4.0, detailed information about the site visitors such as date, time, client IP address and browser type can be gathered, while minimizing log file size by omitting unneeded fields.

   ii. Track specific files – IIS 4.0 also enables certain files to be chosen to collect data, thus improving performance, reducing the log file size, and making it easier to interpret the log files.

   iii. Customize logs on our site – IIS 4.0 allows developers to create custom components for processing specific user activities and writing to a SQL server data source or a custom log file format.

4. ***Enables to analyze server log files and site contents***

   Microsoft Site Server Express-Analysis makes it easier to analyze the server log files and site contents. The Site Server Express feature set provides basic

functionality to that found in the Microsoft Site Server. In order to analysis log files, Site Server Express-Analysis provides an easy way to identify basic trend and usage information from an IIS 4.0 log file in order to make more informed business decisions. The usage Import and Report Writer modules of Site Server Express-Analysis enable to:

i. Import an IIS log file and translate hits into useful information
ii. Easily organize that information to over 20 predefined reports using HTML or popular business software formats
iii. Automate the import and analysis of log files through an easy to use scheduler.

For the content quality maintenance, Microsoft Site Server Express-Analysis also monitors site quality and provides feedback to the site administrators and content developers that will help them to improve the usability of their sites. The Content Analyzer module of Site Server Express-Analysis lets a site administrator to easily visualize how a site is lay out and report on the content types composing the site. With IIS 4.0 and Site Server Express, problems can be easily identified.

5. ***Improving the user experience***

IIS 4.0 makes it easy to customize how content is presented on the site.

6. ***Enables to custom error messages***

IIS 4.0 allows developers to customize the content that is return to a browser or run an application to handle the error. Instead of using the default error messages, administrators can now choose to send custom error messages to the clients. These customizable error messages can be in the form of a file or an URL. Developers can define custom ASP pages to handle specific errors.

7. ***Tightly integrated with Windows 2000 Server***

IIS provides high performance on Windows 2000 and delivers superior price performance to compatible UNIX based systems. IIS also brings the security of Windows 2000 Server to the Web site, without additional configuration, to protect information with the ease of a single user directory and the ability to logon to a network. Windows 2000 Server with IIS also provides an integrated set of tools for running and managing all the networks, Webs, and application services, thus reducing training time for administrators.

2.12 Methodology

2.12.1 Software Process Model

Many process models are described in the software engineering literature. Some are prescriptions for the way software development should progress, other are descriptions of the way software development is done in actuality. In theory, the two kinds of models should be similar or the same, but in practice, they are not. Building a process model and discussing the sub processes help the team understand this gap between what should be and what is.

There are several other reasons for modeling a process:

- When a group writes down a description of its development process, it forms a common understanding of the activities, resources and constraints involved in software development.
- Creating a process model helps the development team find inconsistencies, redundancies, and omissions in the process and in its constituent parts. As these problems are noted and corrected, the process becomes more effective and focused on building the final product.
- The model should reflect the goals of development, such as building high quality software, finding faults earlier in development, and meeting required budget and schedule constraints. As the model is built, the development team evaluates candidates' activities for their appropriateness in addressing these goals. For example, the team may include requirements review, so that problems with the requirements can be found and fixed before design begins.
- Every process should be tailored for the special situation in which it will be used. Building a process model helps the development team understand where that tailoring is to occur.

Every software development process model includes system requirements as input and a delivered product as output. Many much models have been proposed over the years. Several of the most popular models are shown as below:

2.12.2 Waterfall Model (with prototyping)

The stages in the Waterfall model are depicted as cascading from one to another [Royce, 1970]. One development stage should be completed before the next begins. Thus, when all the requirements are elicited from the customer, analyzed for completeness and consistency, and documented in a requirements document, then the development team can go on to system design activities. The Waterfall model presents a very high-level view of what goes on during development, and it suggests to developers the sequence of events they should expect to encounter.

The Waterfall model has been used to prescribe software development activities in a variety of contexts. For example, it was the basis for software development deliverables in U.S. Department of Defense contracts for many years, defined in Department of Defense standard 2167-A. Associated with each process activities were milestones and deliverables, so that project managers could use the model to gauge how close the project was to completion at a given point in time. For instance, "unit and integration testing" in the waterfall ends with the milestone "code modules written, tested, and integrated;" the intermediate deliverable is a copy of the tested code. Next, the code can be turned over to the system testers so it can be merged with other system components (hardware and software) and tested as a larger whole.

The waterfall model can be very useful in helping developers lay out what they need to do. Its simplicity makes it easy to explain to customers who are not familiar with software development; it makes explicit which intermediate products are necessary in order to begin the next stage of development. Many other, more complex models are really just embellishments of the waterfall, incorporating feedback loops and extra activities.

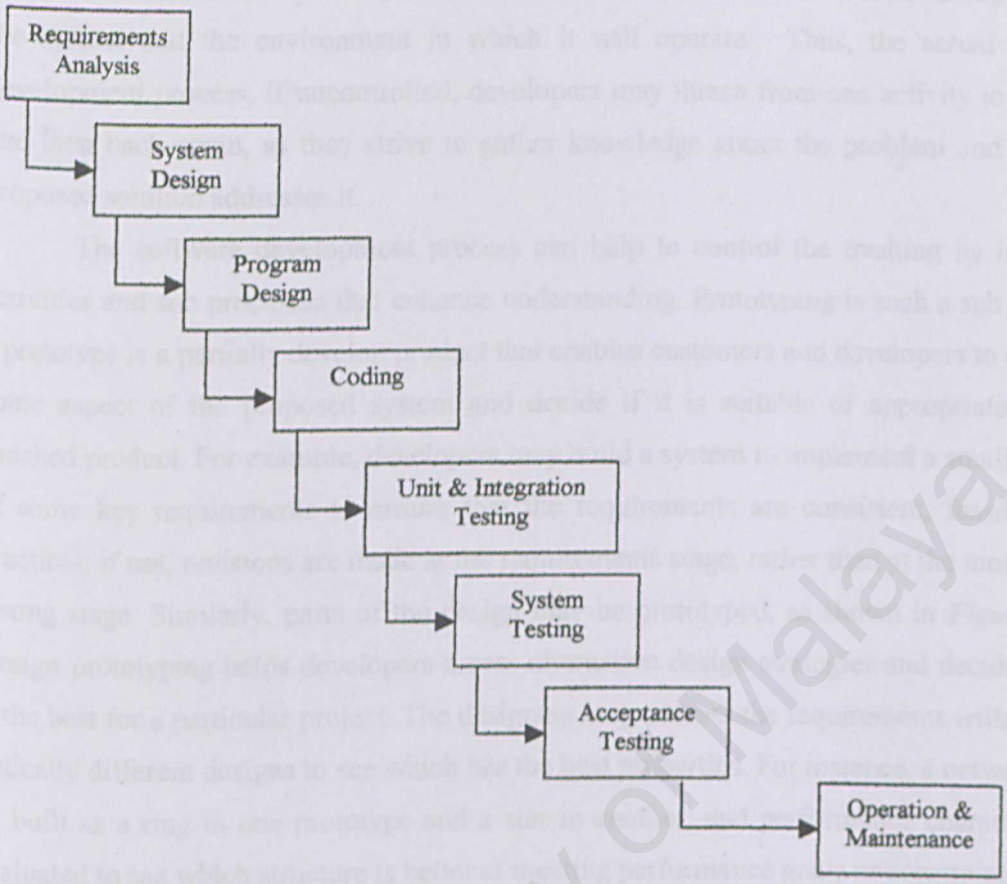


Figure 2.27: The Waterfall Model

The biggest problem with the waterfall model is that it does not reflect the way code is really developed. Except for very well understood problems, software is usually developed with a great deal of iteration. Often, software is used in a solution to a problem that has never before been solved or whose solution must be upgraded to reflect some change in business climate or operating environment. For example, an airplane manufacturer may require software for a new airframe that will be bigger or faster than existing models, so there are new challengers to address, even though the software developers have a great deal of experience in building aeronautical software. Neither the users nor the developers know all the key factors that affect the desired outcome, and much of the time spent during requirements analysis, may be devoted to understanding the items

and processes affected by the system and its software, as well as the relationship between the system and the environment in which it will operate. Thus, the actual software development process, if uncontrolled, developers may thrash from one activity to the next and then back again, as they strive to gather knowledge about the problem and how the proposed solution addresses it.

The software development process can help to control the trashing by including activities and sub processes that enhance understanding. Prototyping is such a sub process; a prototype is a partially develop product that enables customers and developers to examine some aspect of the proposed system and decide if it is suitable or appropriate for the finished product. For example, developers may build a system to implement a small portion of some key requirements to ensure that the requirements are consistent, feasible, and practical; if not, revisions are made at the requirements stage, rather than at the more costly testing stage. Similarly, parts of the design may be prototyped, as shown in ***Figure 2.28***. Design prototyping helps developers assess alternative design strategies and decide which is the best for a particular project. The designers may address the requirements with several radically different designs to see which has the best properties. For instance, a network may be built as a ring in one prototype and a star in another, and performance characteristics evaluated to see which structure is better at meeting performance goals or constraints.

Often, the user interface is built and tested as a prototype, so the users understand what the new system will be like, and the designers get a better sense of how the users like to interact with the system. Thus, major kinks in the requirements are addressed and fixed well before the requirements are officially validated during system testing; validation ensures that the system has implemented all of the requirements, so that each system function can be traced back to a particular requirement in the specification. System testing also verifies the requirements; verification ensures that each function works correctly. That is, validation makes sure that the developer is building the right product (according to the specification), and verification checks the quality of the implementation. Prototyping is useful for verification and validation, but these activities can occur during other parts of the development process.

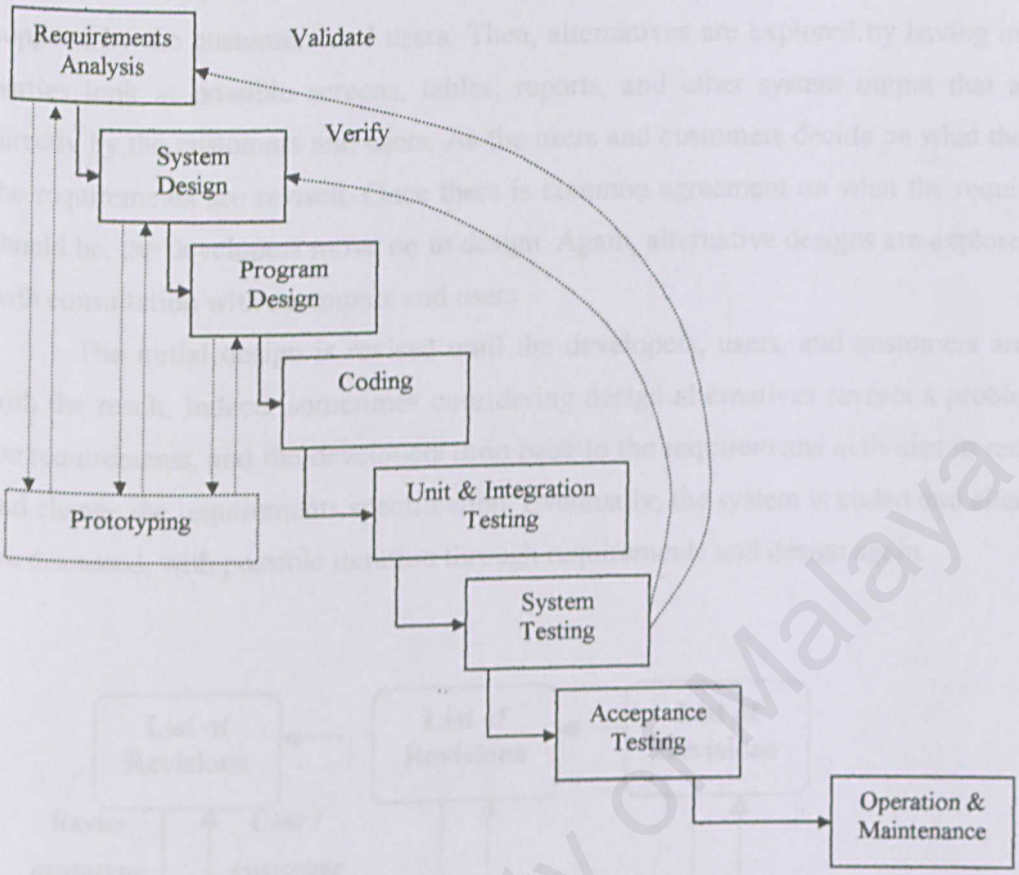


Figure 2.28: The Waterfall Model with Prototyping

2.12.3 Prototyping Model

Prototyping need not be solely an adjunct of a waterfall; it can itself be the basis for an effective process model. Since the prototyping model allows all or part of a system to be constructed quickly to understand or clarify issues, it has the same objective as an engineering prototype, where requirements or design require repeated investigation to ensure that the developer, user, and customer have a common understanding both of what is needed and what is proposed. One or more of the loops for prototyping requirements, design, or the system may be eliminated, depending on the goals of the prototyping. However, the overall goal remains the same, which is reducing risk and uncertainty in development.

For example, system development may begin with a nominal set of requirements supplied by the customers and users. Then, alternatives are explored by having interested parties look at possible screens, tables, reports, and other system output that are used directly by the customers and users. As the users and customers decide on what they want, the requirements are revised. Once there is common agreement on what the requirements should be, the developers move on to design. Again, alternative designs are explored, often with consultation with customers and users.

The initial design is revised until the developers, users, and customers are happy with the result. Indeed, sometimes considering design alternatives reveals a problem with the requirements, and the developers drop back to the requirements activities to reconsider and change the requirements specification. Eventually, the system is coded and alternatives are discussed, with possible iteration through requirements and design again.

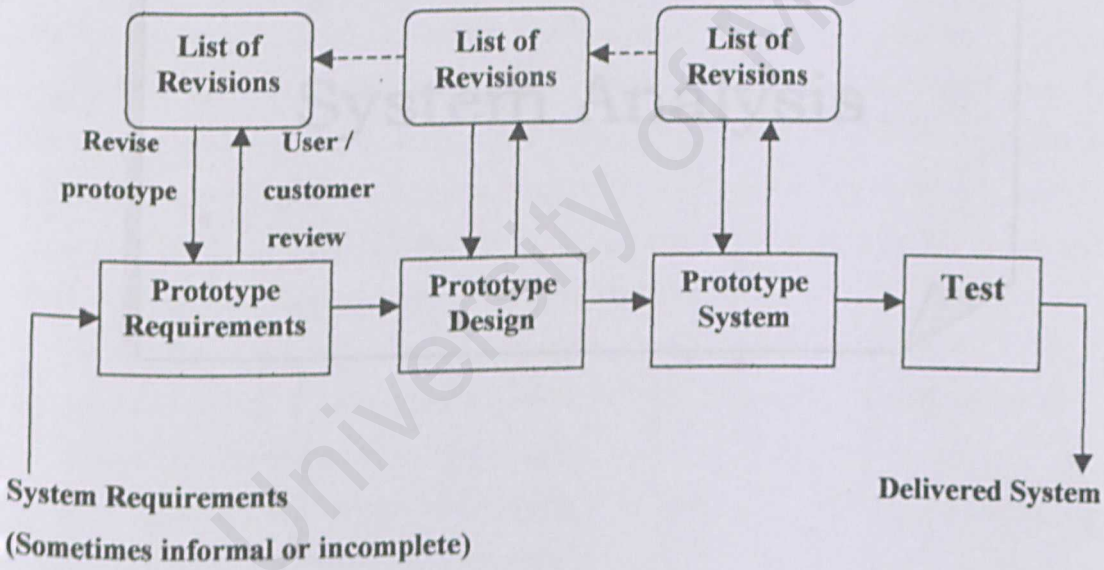


Figure 2.29: The Prototyping Model

Chapter 3:

System Analysis

Chapter 3: System Analysis

3.1 Introduction

System Analysis is a problem-solving technique that decomposes a system into its component pieces for the purpose of studying how well those component parts work and interact to accomplish their purpose. Simply said, it is a phase to find out what a system does and to analyze the system needs. System analysis involves three basic phases, which are:

i) Feasibility assessment (Project Survey Phase)
ii) Organization problem statement (Project Study Phase)
iii) Organization requirements statement (Definition Phase)

Feasibility assessment involves four categories, which are interviews, project scope, problem statement and classification, and proposed project plan. On the other hand, organization problem statement includes project roles, learning current system (use repository), modeling the current system, analysis of problems and opportunities, new system's objectives, and new project scope and plan. Finally, organization requirements statement is formed by identifying requirements, modeling system requirements, discovering prototype, prioritization, and reviewing requirements.

The aim of system analysis is to analyze, specify and define the system, which is to be built. The models developed in this phase will describe what the new system will do. The benefits of system analysis are:

i) Provides an overview of the proposed system
ii) Defines the scope of the system
iii) Defines what the system will do
iv) Provides a solid understanding of the system

System analysis starts with data collection. Some techniques have been carried out in the information collecting process such as listed below:

3.1.1 Information collecting process

- ***Feasibility study***

Times have been given and efforts has been put to gather the information by reading references books, journals, Web sites and user guides that are relevant to the project as listed in Bibliography. The purpose is to find out the suitable Web languages and database that are going to be used for the system development.

- ***Interview***

This is a fact-finding technique whereby information is collected from individuals through face-to-face interaction. Interviewing can be used to achieve any or all of the following goals: find facts, verify facts, clarify facts, generate enthusiasm, get the end user involved, identify requirements and solicit ideas and opinions. There are two roles in conducting an interview where the system analyst is the interviewer (responsible for organizing and conducting the interview), and the system user or system owner is the interviewee (who is asked to respond to a series of questions).

- ***Internet surfing***

Internet surfing is considered as a good method and choice of fast finding for desired information. A lot of information can be obtained from Internet to fulfill the need in this project. Besides, online tutorials regarding programming languages can also be obtained through surfing the Internet. Analyzing the existing online system has made a big help in giving ideas on the features, functionalities as well as the design of the Web based title management system.

3.2 Methodology

3.2.1 Waterfall model (with prototyping)

Waterfall model with prototyping will be use for the development of **OTMS**. This is because such model provides us with the advantages of both the waterfall and prototyping models as listed below:

i. ***Easily explained***

The waterfall model can be easily explained to the customers even if they do not have any basics in software development.

ii. ***High level of view***

The waterfall model gives the system developer a high level of view about what is going on in every stage in the development process. It makes explicit which intermediate products are necessary in order to begin the next stage of development. It also suggests to developers the sequence of events they should expect to encounter.

iii. ***The most basic model***

The waterfall model is the most basic model of software process model. This makes it very easy to learn and use. Besides, more complex models are really just embellishments of the waterfall model, incorporating feedback loops and extra activities.

iv. ***Easy layout***

Software developers can easily layout the whole development process. They can plan exactly what to do in every stage and what is needed (milestone) before they can proceed to the following stages. The plan can even be layout before the actual project started.

v. ***Iteration is enable***

Using the waterfall model with prototyping, iteration is enable in the sense that developers can move back and forth between stages. This can be applied for requirements analysis, system design and program design stage. In the traditional waterfall model, after the developers move to a specific stage, they cannot move back to the previous stage and this is not very practical in the actual development process.

vi. ***Verification and validation are enable***

Combining prototyping into waterfall model enables verification and validation to be made throughout the whole system development process. Customers and developers can also examine some aspect of the proposed system and decide if it is suitable or appropriate for the finished product. Verification ensures that the system has implemented all of the requirements and validation ensures that each function works correctly.

For the prototyping section, users will be selected from FSKTM to represent the target users of the system. For **OTMS**, Puan Miss Laiha Mat Kiah will be the representative for lecturers and one of my course mates, Mr. Lim Tong Leong will be the representative for students/visitor.

3.3 Requirement Analysis

Requirement analysis is a structured document setting out detailed description of the system services. It should only specify the external behavior of the system and should not be concerned with system design characteristics. It is often written in natural language so that it is understandable by customers without knowledge of specialized notations. There are two types of requirement analysis, which are functional requirements and non-functional requirements.

3.3.1 Functional requirements

Functional requirements are a description of activities and services a system must provide, how the system should react to particular inputs, and how the system should behave in particular situations. It is frequently identified in terms of inputs, outputs, processes and stored data that are needed to satisfy the system improvement objectives. In some cases, the functional requirements may also explicitly state what the system should not do.

For **OTMS**, the functional requirements can be categorized into 3 sections. These sections are lecturer module, administrator module and student/visitor module.

i) Administrator module

- Authentication and authorization function:
  This function is used to protect the system database from any unauthorized users. Authorized users will be provided with specific user identification and a password to access the system and the data stored in the database. This function is the default startup function before users can access other functions in Administrator Module.
- Title maintenance function:
  This function will provides the capabilities of adding a brand new record; deleting, updating or searching for an existing record in the database.

Administrators are allowed to maintain all the records stored in the system. Thus, this function gives administrators with more rights to access all the records easily without any limitations.

- User registration function:

  Administrators will have one extra task compared to the lecturers. They will be in charged of the users registration process. Each user must provide an user ID to the administrator for registration purposes. After that, the administrator will give a default password and any registered users can then effortlessly change their password through the change password function. Beside that, administrators can also delete any certain users from the system.

- Change password function:

  This function enables registered administrators to change their password. They just need to key in the user ID and the original password followed by the desired new password to update the system. It's main objective is to maintain a secure environment for **OTMS** all the time.

ii) Lecturer module

- Authentication and authorization function:

  This function is the same as in the administrator module. Please refer to the administrator module above for more information.

- Title maintenance function:

  This function will provides the capabilities of adding a brand new record; deleting, updating or searching for an existing record in the database. Each registered lecturer will have access to this function so he/she can maintain the titles in the system easily. This function differs from the administrator module in the sense that lecturers are only allowed to modify their own titles and thus prevent other lecturers from accidentally make some unnecessary changes.

- Compare function:
  This is the only different function that makes lecturer module differs from administrator module. It is very useful for a lecturer before he/she add a new title to the system. With this function, lecturers can easily compare their title with the existing titles to avoid redundancy.
- Change password function:
  This function is the same as in the administrator module. Please refer to the administrator module above for more information.

iii) Student/visitor module

- Browsing and searching function:
  This is the only function available for students/visitors. With this function, they can easily search and view the titles available for any semester of an academic year. Information about the titles, such as its status, descriptions, etc. is available through the use of this function. The browsing and searching function will be using input from students/visitors such as title, lecturer name, department, session and semester.

3.3.2 Non-functional requirements

Non-functional requirements are a description of other features, characteristics, and constraints that define a satisfactory system. They include timing constraints, constraints on the development process, standards and so on. Although the non-functional requirements are quite subjective, they are as important as the functional requirements.

❖ **Reliability**

It is reliability if the application system, software and hardware do not cause unnecessary failure and downtime when they are used in a reasonable manner.

- **Usability**

  The application system shall be easy to use. There should be no complex and unnecessary step to perform. They shall enhance and support rather than limit or restrict business processes.

- **User-friendly**

  The application system is required to have a user-friendly interface for the ease of usage. Graphical User Interface approach should be applied for better visual effect to the user. Effective error handling and validation will also help the user to navigate through the system with more confidence. Error message should be display to the user to indicate what is going wrong rather than for the user to guest what is happening.

- **Response time**

  The response time should be within a reasonable interval time in retrieving any data or information. Good application system should have a shorter response time.

- **Maintainability and expandability**

  The architecture and database design should be able to maintain and can be extended if necessary amendment is required in the future.

- **Security**

  The application system shall be able to prevent unauthorized users from accessing the system.

- **Manageability**

  The application system, hardware and software shall be easy to manage and operate. The system shall be managed by not more than 2 persons to avoid unnecessary human resource wastage.

- **Robustness**

  The application system shall be able to handle or at least avoid disaster in facing any unexpected situation and data.

3.4 Technology Consideration

3.4.1 Programming languages

3.4.1.1 Active Server Pages (ASP)

ASP is a new technology from Microsoft that provides the capability for the Web server to process application logic and then delivers standard HTML to the client browser.

ASP and Internet Information Server (IIS) act as a medium for porting existing applications to build new application for the Web. ASP is faster in execution and it will save customers a lot of time because it is more convenient to the customers that they can get some information quickly. When come to the time frame consideration, ASP properly is the appropriate technology used to develop **OTMS** since ASP is fast in execution, time saving, easy to learn and use Web language.

ASP is selected also because of the features provided and high stability of ASP to the system. It includes performance, security and relatively low cost. Besides, it also well supported by many applications from Microsoft especially such as the integration with Microsoft Internet Information Server (IIS), the ODBC drivers, VBScript, SQL 7, ActiveX and of course Windows 2000.

3.4.1.2 VBScript

VBScript is selected as the scripting language because it is a fast, portable, lightweight interpreter for the use in World Wide Web. Beside that, it is extremely easy to learn and use, and can be easily integrated with ASP.

3.4.2 Database

3.4.2.1 Microsoft SQL Server 7.0

There are two types of database, which are relational database and object-oriented database. Both of the databases have their own strengths and lacks. However, **OTMS** will be using the relational database – Microsoft SQL Server 7.0, because of the following reasons:

- The lacks of reporting features in object-oriented database compared to the relational database.
- Records of **OTMS** are not in type of graphic and the system is not a scientific application or CAD or engineering system.
- The stability and features in relational database.
- The loading time is extremely faster for relational database in terms of retrieving many text-based records.

Database is a self-describing collection of integrated records. After the research and review from Chapter 2, Microsoft SQL (MSSQL) is chosen as the database management system for **OTMS**. This is because:

- MSSQL is very suitable for Web based application
- MSSQL has the speed to support more than a few hundred users over the network at one time
- MSSQL is not limited in the size of each record and each table

3.4.3 Programming tool

3.4.3.1 Microsoft Visual InterDev 6.0

I have decided to use Visual InterDev because it is a tool that enabled developers to build dynamic and interactive applications for the Web. A problem occurs under the Web-based model in that the Internet doesn't inherently support interactive sessions with the user like the client-server model. Visual InterDev supports the use of server-side components to accomplish a persistent dialogue with the user and, thus, an interactive session.

Visual InterDev provides the tool to take advantage of the new Web-based model of computing. Some development tools focus on supporting a single Internet technology. Others support several technologies for Web-based development but don't provide visual tools to accomplish these tasks. Visual InterDev exceeds existing Web development tools by providing a way to integrate multiple technologies and supplies visual tools to greatly enhance a developer's productivity. Visual InterDev also surpasses and extends the reach of client-server tools to the Internet and the Web.

3.4.4 System Platform

3.4.4.1 Windows 2000

Windows 2000 operating system is selected because it provides us with lots of benefits, such as:

1. ***Compatible with Microsoft and Netscape***
   - ❖ Microsoft Windows 95/98 is the worldwide most preferred desktop operating system and the leader in developing software. Meanwhile, Netscape specializes in innovate cutting edge of today Internet applications.
   - ❖ Most of the programming tools in today's trend are Microsoft based developed languages. For instance, Visual Basic, Visual C++, and Visual J++. Moreover, some DBMS tools such as Microsoft Visual InterDev and Microsoft SQL are gaining popularity in most corporations.
   - ❖ Most of today applications are Windows and DOS compatible. Integrity of module into a system will not need any patch from different software developers. Thus, it lessens the time and cost in developing a system. This is beneficial for the previous developed software in terms of investment.
   - ❖ LINUX or UNIX operating systems do not offer much software development tools especially Web based publishing software. Lack of development tools has contributed to less efficient and slow progressing

operating systems. Most of the applications in UNIX are developed via a command line compiler and simple text editor.

2. *Cost savings*
   - ❖ Windows 2000 is a very cost effective operating system. With the budget around RM 2000, a fully functional Internet server is running in a matter of days.
   - ❖ Microsoft provides support and maintenance for the purchased software. It is advisable to utilize other Microsoft products since incentive will be given.
   - ❖ Windows 2000 scalability where it can moves to an even faster, multiprocessor PC server or a high performance Reduced Instruction Set Computer (RISC) based server, thus offer flexibility of the operating system.
   - ❖ In order to make certain modification in UNIX, the whole operating system need to be recompiled which is differ from Windows 2000 in major configuration only by rebooting the server. Hence, it cuts administration time and cost.

3. *User friendly environment*
   - ❖ Interface of Windows 2000 environment origin from the Windows 3.X operating system where the first and foremost windows like interface that offers multitasking with the user friendly feature.
   - ❖ Differ from UNIX or LINUX operating systems where most of the tasks are performed by command script, which is definitely very troublesome. Furthermore, the user interface is too cryptic and not user friendly at all. Learning some command task may waste a great deal of time.
   - ❖ Besides cryptic user interface, UNIX applications demand step learning curve where it is very difficult for a new user. Thus, it is time consuming and waste lots of learning time and cost.

- Complicated installation procedure – Each UNIX machine has their different documented installation procedure. Prior to installation, the source code (Kernel) needs to be compiled.

4. *Security*
   - The distribution of LINUX and UNIX source code is widely available in the Internet. Besides the administrator benefited from the source code, the potential hacker has the possible vulnerability point of attack.
   - However, this differs from Windows 2000 where most of the applications are not freely available in the Internet. Besides, Windows 2000 operating system utilizes binary codes, thus result complicated vulnerability points of security. As a result, this can seriously compromise the security of many servers at a time where they use the same codes.

3.4.5 Web Server Software

3.4.5.1 Internet Information Server (IIS) 4.0

OTMS will be using Windows 2000 as the platform and since IIS 4.0 is free together with Windows 2000, it is the best Web server to choose. Not only that, Windows 2000 and IIS 4.0 is integrated together to give a high performance, security and feasibility **OTMS** environment. Other advantages of IIS 4.0 includes:

- A robust and capable Web server program
- Suitable for small sites right up to enterprise-class sites
- Provides a central server management from any server on the network
- Inclusions of ASP
- Includes its own Internet Services API (ISAPI)
- Provides database support including ODBC and Microsoft SQL

3.5 System Requirements

System requirements consist of two components, which are software and hardware requirements.

3.5.1 System development environment

3.5.1.1 Hardware requirements

- PC with at least Pentium II processor
- 128 MB of RAM
- 4 GB of Hard Disk

3.5.1.2 Software requirements

- Windows 2000 Professional
- Microsoft Visual InterDev 6.0
- Microsoft SQL Server 7.0
- Microsoft Internet Information Server (IIS) 4.0

3.5.2 System deployment environment

3.5.2.1 Server hardware requirements

- PC with Pentium II processor or higher
- 128 MB of RAM or higher
- 8 GB of Hard Disk or higher

3.5.2.2 Server software requirements

- Windows 2000 Server or Advanced Server
- Microsoft SQL Server 7.0
- Microsoft Internet Information Server (IIS) 4.0

3.5.2.3 Client's hardware and software requirements

- Any computer with IE 5.0 or above and connection to the Internet

Chapter 4:

System Design

Chapter 4: System Design

4.1 Introduction

System Design is a stage of system development where the requirements for the system are translated into the system characteristics to meet the user requirement and satisfaction. System Design is the creative process of transforming the problems into solutions. It is also called Physical Design. System Design is a very important phase in the system development, which determines the successfulness and accomplishness of a system. System design is composed of three phases:

- Select a design target from a candidate solutions (Selection Phase)
- Acquire necessary hardware and software (Acquisition Phase)
- Design and integrate the new system (Design and Integration Phase)

System design is driven by the technical concerns of system designers. Therefore, with respect to the information systems building blocks, system design addresses the data process and interface building blocks from the system designer's perspective. The design of **OTMS** can be viewed from the following aspects:

- System functionality design
- Database design

4.2 General System Design

OTMS has 3 main modules:

1. Administrator module
2. Lecturer module
3. Student/visitor module

4.2.1 Administrator module

This module is especially for the administrators to do controlling and maintenance of the system. Before the administrators can access this particular module, they need to logon as authorized and authenticated users.

Administrators will use this module to help them in title maintenance, which are the data entry (add) and editing process (delete, update or search) of all the thesis titles whenever it is necessary. The data involved in this feature are lecturer name, title, language, tools, descriptions, department, student name, matric number, session, semester and last updated. Then, administrators will also use this module to maintain the user registrations for the system. They will be able to add, delete or update a user's information such as user name, user ID, password and status. Beside that, administrators can also change their own password whenever they like.

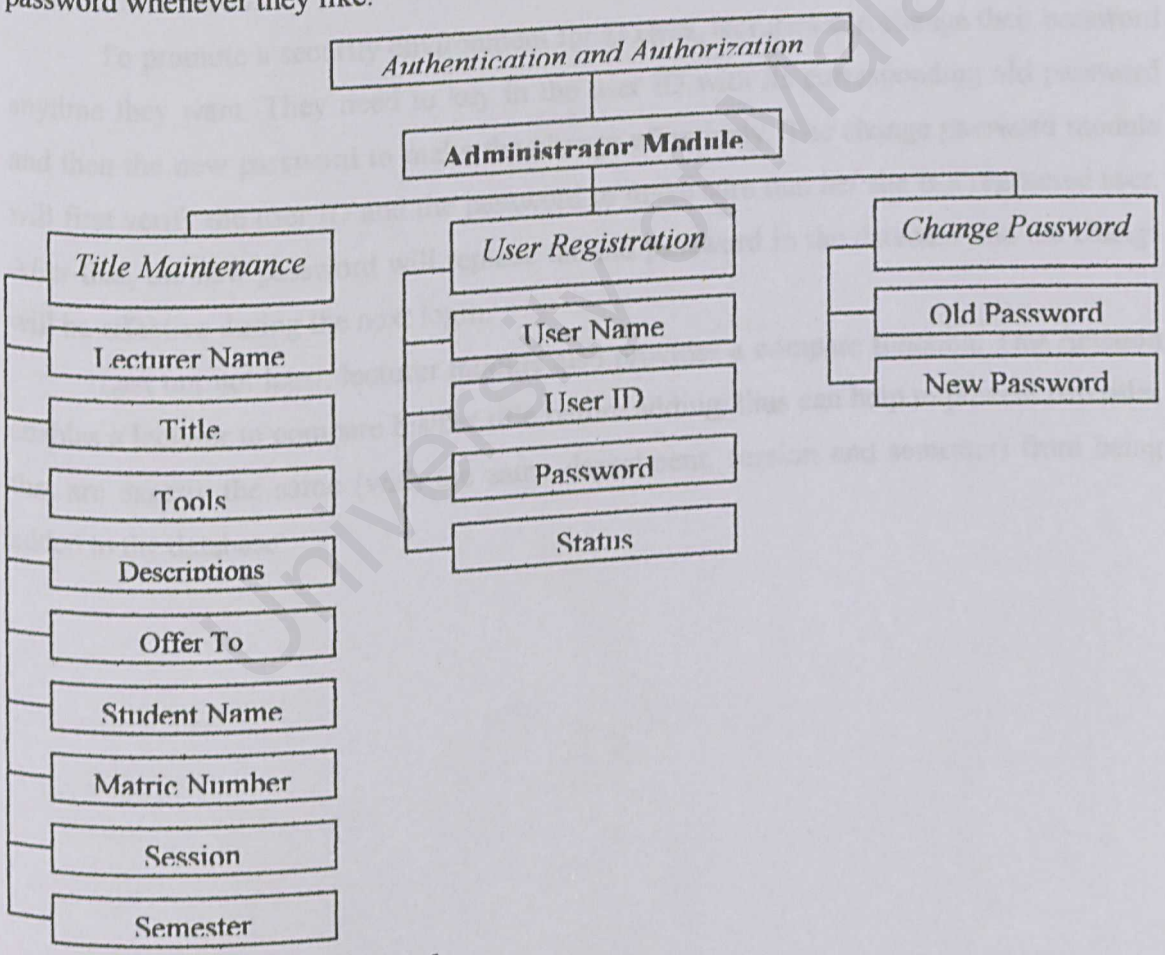


Figure 4.1: Administrator Module

4.2.2 Lecturer module

This module is designed specially for the main group of users, which are the lecturers, to perform their task. All lecturers must register with the administrator to become authorized users of **OTMS**. Before lecturers can access the functionalities provided in this module, they must logon through the logon page first.

Lecturers can use this module to maintain their thesis titles for every session in an academic year. However, please notice that this function is not the same as in the administrator module. In lecturer module, users can only maintain their own titles. Every lecturer can add a brand new title, edit or update their existing titles and search for all titles in the database. In other words, lecturers will not have the rights to interfere with other lecturers' thesis titles. The data involved in the title maintenance function are the same as in administrator module.

To promote a security environment for **OTMS**, lecturers can change their password anytime they want. They need to key in the user ID with its corresponding old password and then the new password to make the change effectively. The change password module will first verify the user ID and the password to make sure that he/ she is a registered user. After that, the new password will replace the old password in the database and the change will be effective during the next login.

Last but not least, lecturer module also provides a compare function. This function enables a lecturer to compare his/her title before adding, thus can help to prevent two titles that are exactly the same (with the same department, session and semester) from being added to the database.

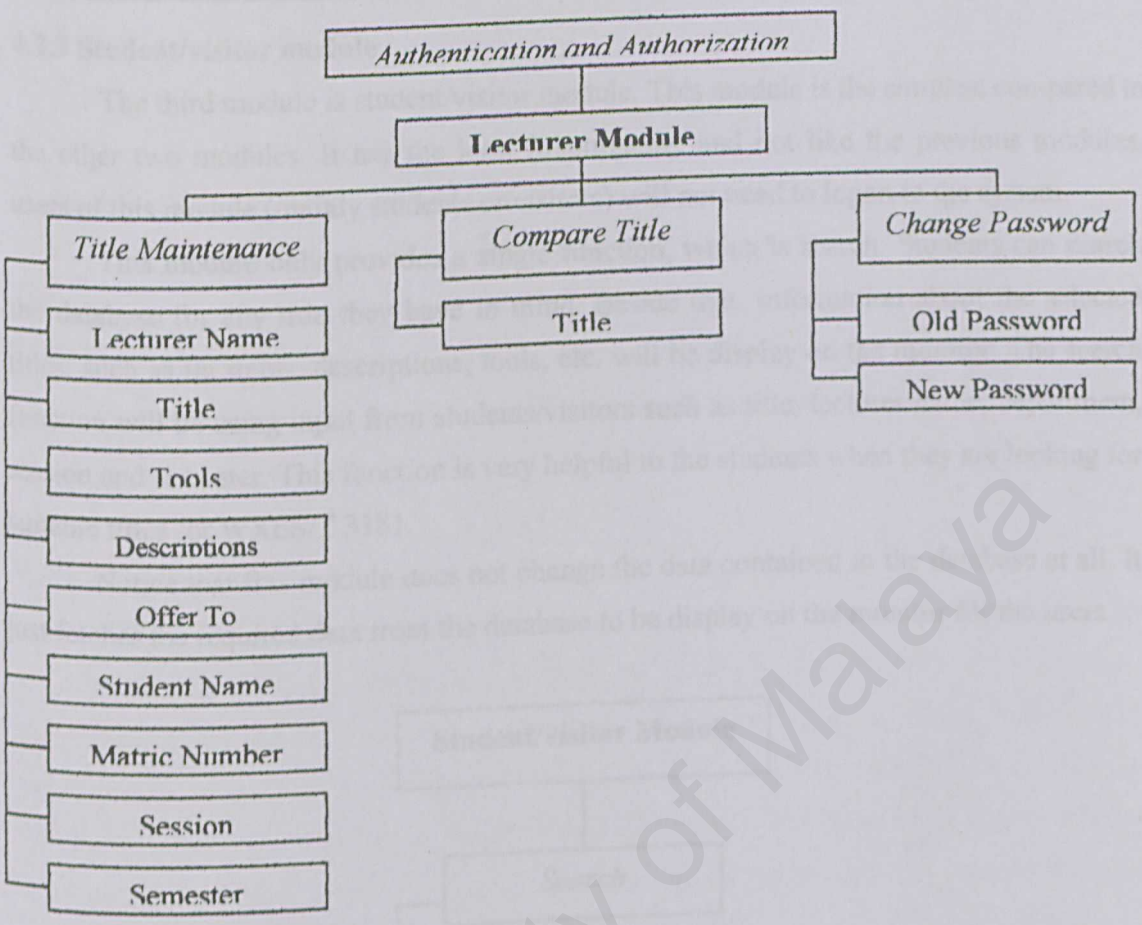


Figure 4.2: Lecturer Module

4.2.3 Student/visitor module

The third module is student/visitor module. This module is the simplest compared to the other two modules. It has the least functionality and not like the previous modules, users of this module (mainly students or visitors) will not need to logon to the system.

This module only provides a single function, which is search. Students can search the database for any title they have in mind. Beside that, information about the selected titles, such as its status, descriptions, tools, etc. will be display on the monitor. The search function will be using input from students/visitors such as title, lecturer name, department, session and semester. This function is very helpful to the students when they are looking for suitable titles for WXES/T 3181.

Notice that this module does not change the data contained in the database at all. It just fetches the required data from the database to be display on the monitor for the users.

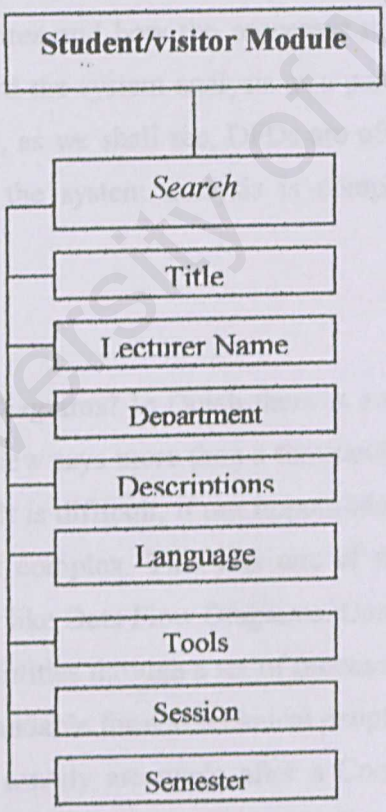


Figure 4.3: Student/Visitor Module

4.3 System Functionality Design

4.3.1 Data Flow Diagram (DFD)

4.3.1.1 Preface

Data Flow Diagrams were developed in the early sixties. In that particular period, there has been a lot of writing in professional magazines and books. Because there has been so much writing in the literature about it, nowadays it is a trivial topic. As a consequence of it, it is hard to find any article about Data Flow Diagrams. Especially about the history and the development throughout the time, there is hardly any information.

4.3.1.2 General Information

Data Flow Diagrams (DFDs) are diagrams that show the flow of data from one place to another. DFDs describe the processes of a system, showing how these processes link together through data stores and how the processes relate to the users – the outside world. They are used to record the system analysis as a part of the design documentation. At their lowest level of detail, as we shall see, DFDs are often included in a programmers working specification when the system analysis is complete and the system is being programmed.

4.3.1.3 Importance of DFD

Why use Data Flow Diagrams? In Dutch there is an expression: 'n blik zegt meer dan duizend woorden' (one view says more than a thousand words). This is also true for a graphical description model. It is difficult, if not impossible, to describe a model in words and still being clear and not complex. This was one of the main reasons to develop a graphical modeling technique like Data Flow Diagrams. Data Flow Diagram describes how data flows from the external entities through a set of processes or activities in the system. It makes a system more understandable for non-technical people.

Data Flow Diagrams usually are made after a Context Diagram has been made, because the Context Diagram functions as the basis of a Data Flow Diagram. It is important not to forget the Data Flow Diagrams are not a model of control or sequence of processing

in a system. Data Flow Diagrams must be as a model that shows the flow of data through a system.

The system design is based on data flow oriented or structured design. In the Data Flow Diagram (DFD), functional transformations changes a process to another process, it is transformed as it moves. The symbols used in the DFD are shown as below:

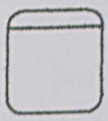
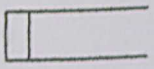
Name	Symbol	Description
Data Flow		Data transfer in the direction indicated by the arrow. Each arrow should be labeled to indicate what data is being transferred.
Process		Manual or computer process that changes data.
External Entity		Source or destination of data that is external to the system.
Database		Manual or computer storage of data

Table 4.1: DFD Symbols

4.3.1.4 DFD of functionalities in OTMS

As mentioned in the General System Design section above, **OTMS** will have three modules in total and from them, we can get 6 different functionalities (please refer to section 3.4.1 Functional requirements of Chapter 3: System Analysis). The DFD for the system overview and its functionalities together with its explanation is provided below.

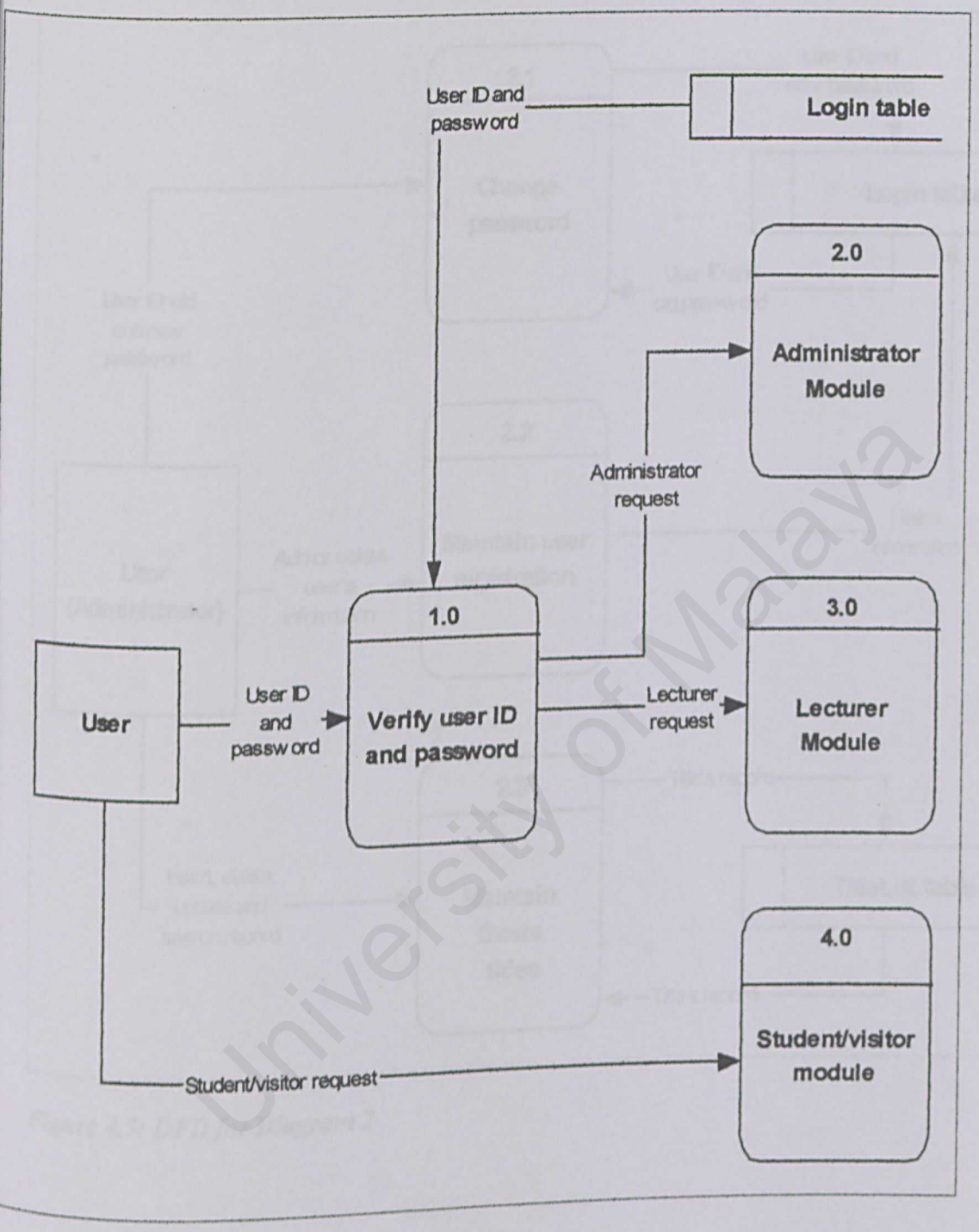


Figure 4.4: DFD for Diagram 0

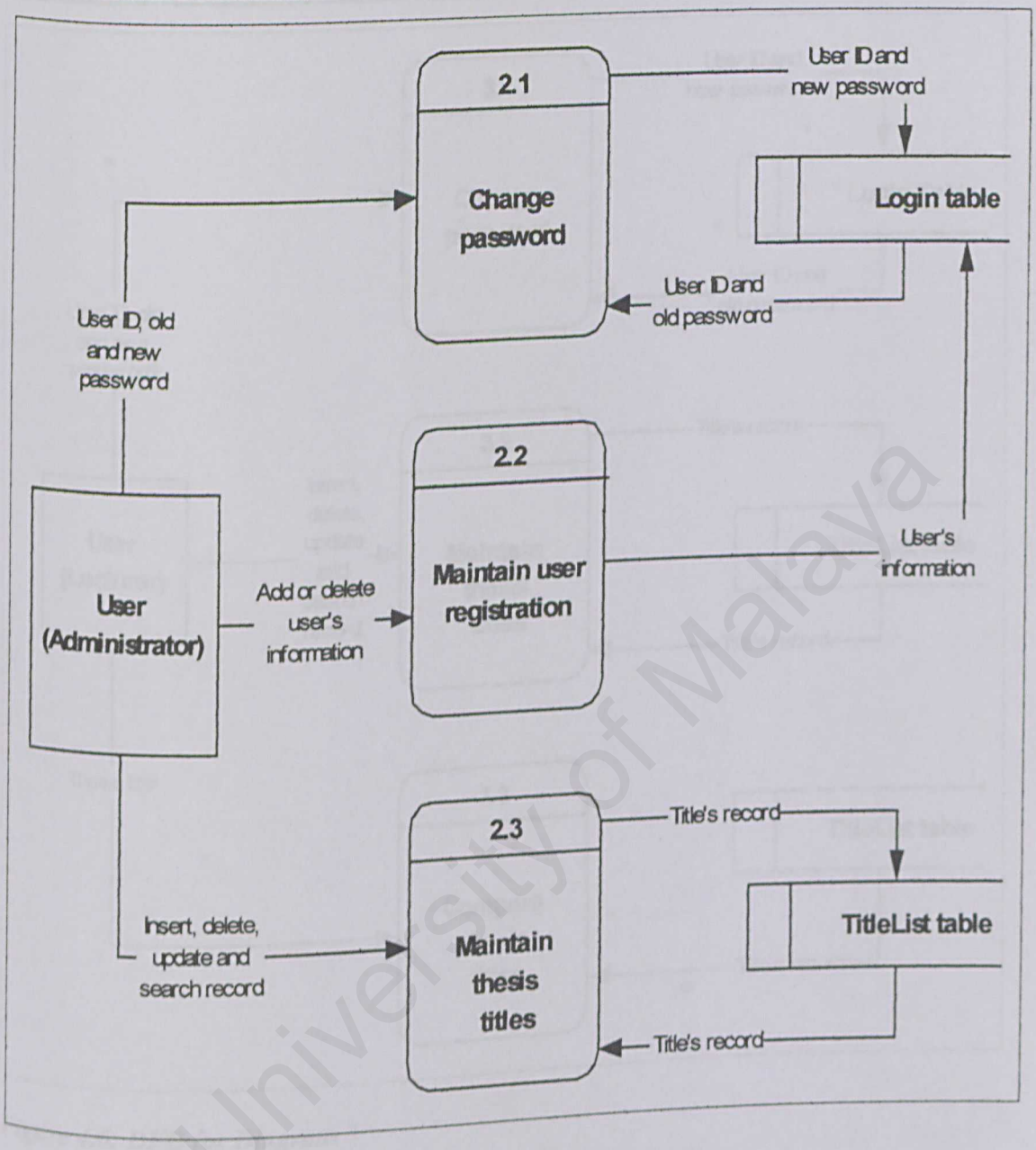


Figure 4.5: DFD for Diagram 2

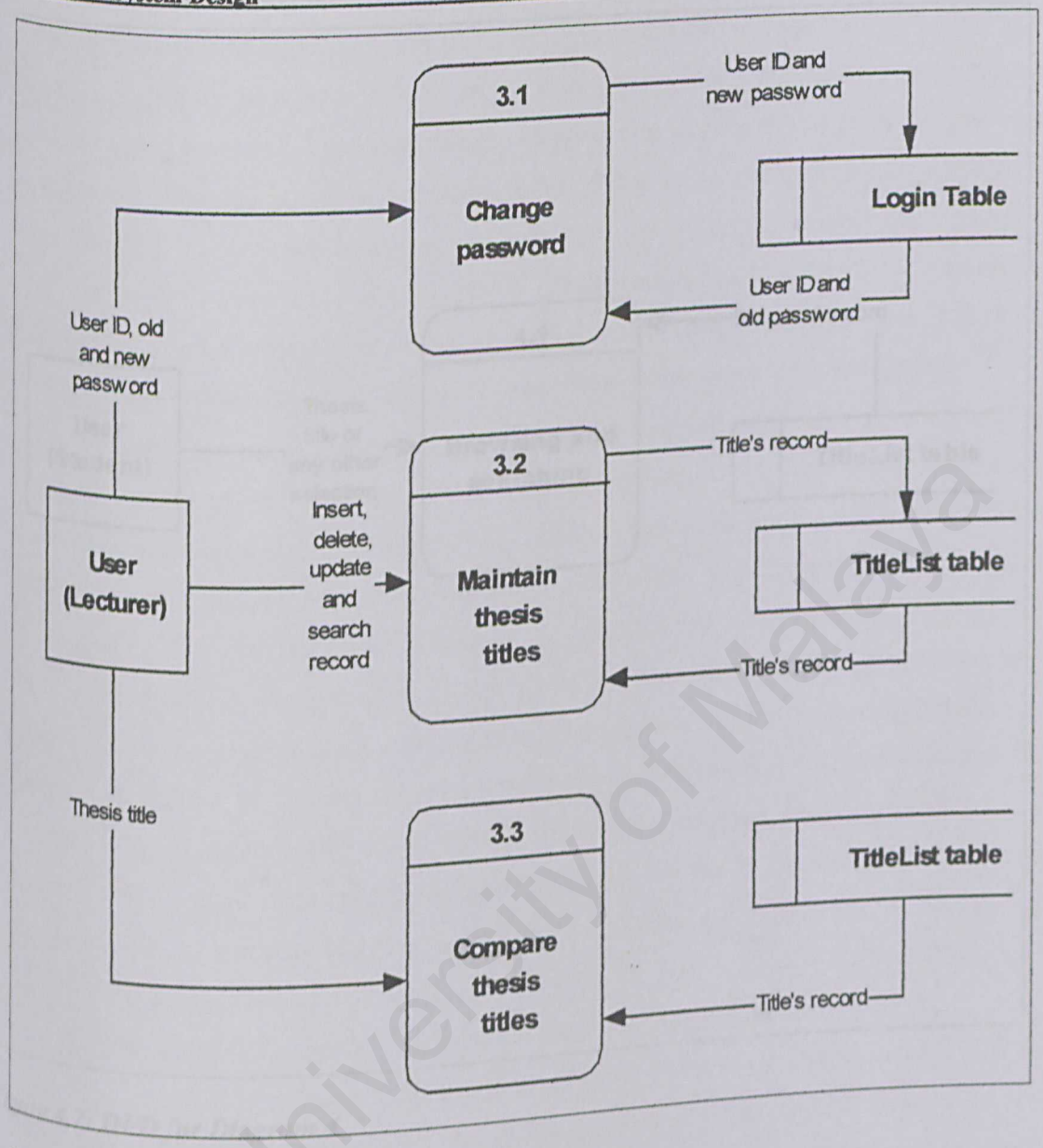


Figure 4.6: DFD for Diagram 3

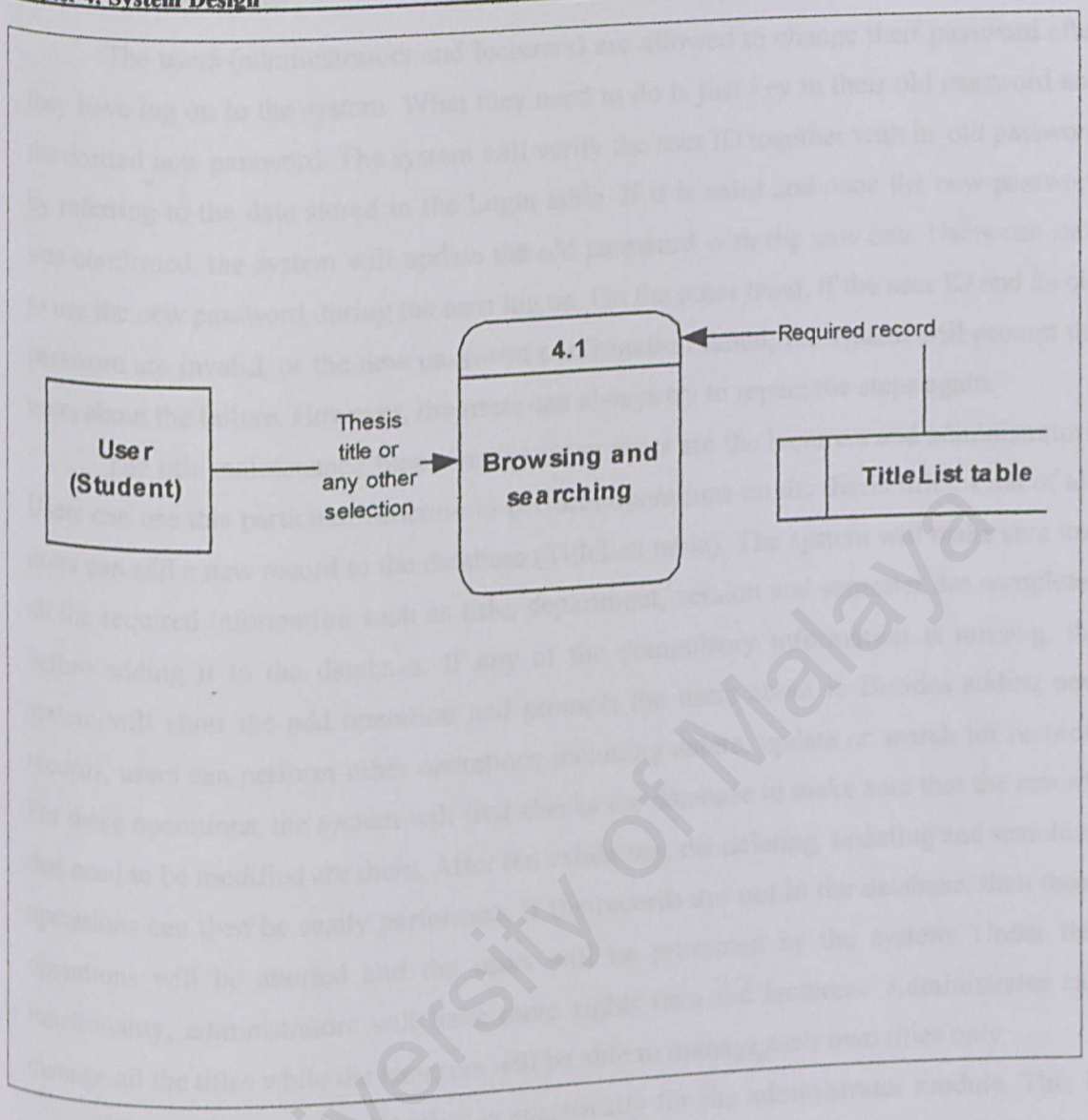


Figure 4.7: DFD for Diagram 4

Before users (administrators or lecturers) can proceed to the other functionalities to help them in performing their desired task, they must logon to the system first. Users are required to key in their user ID and the corresponding password. Verification will be done on the user ID and password by retrieving the records from the Login table. Successful login will get the users to another page to let them start their work. On the other hand, unsuccessful login will brings the user back to the login page again and an "Unsuccessful login" message will be displayed.

The users (administrators and lecturers) are allowed to change their password after they have log on to the system. What they need to do is just key in their old password and the desired new password. The system will verify the user ID together with its old password by referring to the data stored in the Login table. If it is valid and once the new password was confirmed, the system will update the old password with the new one. Users can start to use the new password during the next log on. On the other hand, if the user ID and its old password are invalid, or the new password confirmation failed, the system will prompt the users about the failure. However, the users can always try to repeat the steps again.

The title maintenance functionality target users are the lecturers and administrators. Users can use this particular function to perform operations on the thesis titles. First of all, users can add a new record to the database (TitleList table). The system will make sure that all the required information such as title, department, session and semester are completed before adding it to the database. If any of the compulsory information is missing, the system will abort the add operation and prompts the users about it. Besides adding new records, users can perform other operations including delete, update or search for records. For these operations, the system will first checks the database to make sure that the records that need to be modified are there. After the validation, the deleting, updating and searching operations can then be easily performed. If the records are not in the database, then those operations will be aborted and the users will be prompted by the system. Under this functionality, administrators will have more rights than the lecturers. Administrator can manage all the titles while the lecturers will be able to manage their own titles only.

User registration functionality is specifically for the administrator module. This is because only the administrators have the rights to maintain the registered users for the whole system. To add a new user to the system, administrators need to key in the new user name, user ID, password and status. After that, the system will check whether all the required information (name and user ID) is complete. If the answer is no, then a failure message will be prompted to the administrator, or else if the answer is yes, the system will proceed to check the database (Login table) to see whether the desired user ID already existed or not. If the desired user ID already existed, a failure message will be prompted to

notify the administrator, otherwise, the new user ID will be added into the database to complete the registration process.

User registration functionality also provides the delete function for the administrators to unregister any specific users. Administrator needs to key in the user's name and ID for this purpose. The system will check the name and user ID to make sure whether he/she is a valid user. Only valid users can be deleted from the database while invalid users will be reported to the administrator through an operation failure message.

Compare functionality is specially designed for the lecturer module. The only users for this functionality are the registered lecturers. Lecturers can key in the thesis title that they wanted to compare. After that, **OTMS** will checks the database (TitleList table) to compare the lecturer's input with existing records. The comparison results will be display on the monitor to help the lecturers in making their decision on whether to put up the particular title or not.

Last but not least, the users (students/visitors) of browsing and searching functionality do not need to log on to the system. They can access it by clicking the link to this functionality, which is the Student Access link. Users must key in the data or information of what they intended to search, such as thesis title, lecturer name, department, session semester, etc. Drop down lists will also be provided in the user interface to simplify their selections. After that, the system will search the database (TitleList table) for the records. If the records exist, all the matching records will be display on the monitor. Else, the system will notify the users that no matching records were found.

4.4 Database Design

Database is collections of a large storage of computerize data. With no doubt, databases always are the nucleus of most information systems. Database design is the process of translating logical data models into physical database schemas. Almost every request in the **OTMS** needs the access to database either for simple query or update records in the database.

4.4.1 Database Tables

The database of **OTMS** is created using Microsoft SQL Server 7.0 and it contains 2 main tables. The database diagram for **OTMS** is as follow.

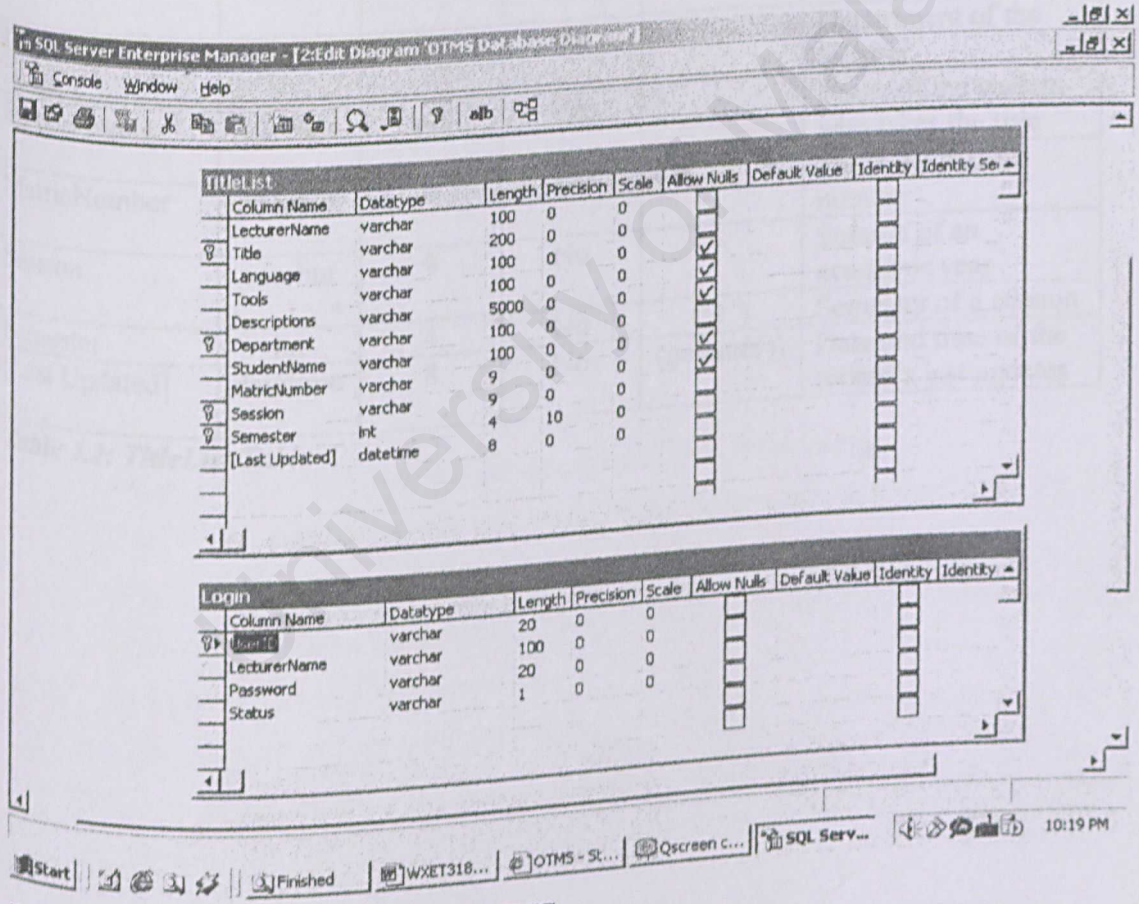


Diagram 4.8: Database Diagram for OTMS

4.4.1.1 TitleList table

This table is use to record all the information about the thesis titles. The design of the TitleList table is shown in the table below.

Column Name	Data Type	Length	Allow Nulls	Default Value	Description
LecturerName	varchar	100	No	-	Name of the lecturer who put up the title (*Foreign Key*)
Title	varchar	200	No	-	Thesis title
Language	varchar	100	Yes	-	Suggested language for the title
Tools	varchar	100	Yes	-	Suggested tools for the title
Descriptions	varchar	5000	Yes	-	Descriptions of the title
Department	varchar	100	No	-	Department of the lecturer
StudentName	varchar	100	Yes	-	Name of the student who takes the title
MatricNumber	varchar	9	Yes	-	Student's matric number
Session	varchar	9	No	-	Session of an academic year
Semester	int	4	No	('1')	Semester of a session
[Last Updated]	datetime	8	No	(getdate())	Date and time of the record's last updates

Table 4.2: TitleList Table

4.4.1.2 Login table

This table is use to record all the information about the registered users of **OTMS**. The design of the Login table is shown in the table below.

Column Name	Data Type	Length	Allow Nulls	Default Value	Description
LecturerName	varchar	100	No	-	Lecturer's or administrator's name (*Primary Key*)
UserID	varchar	20	No	-	Lecturer's or administrator's user ID
Password	varchar	20	No	('password')	Lecturer's or administrator's password
Status	Char	1	No	('L')	User's status (L=Lecturer, A=Administrator)

Table 4.3: Login Table

4.4.1.3 Explanation for some of the database terms

- Column Name property is the name of a particular column in the SQL database table.
- Datatype property specifies the data type of that particular column.
- Varchar datatype is variable-length non-Unicode character data with length of *n* characters. *n* must be a value from 1 through 8000.
- Int datatype is integer (whole number) data from -2^31 (-2,147,483,648) through 2^31 - 1 (2,147,483,647).
- Datatime datatype is date and time data from January 1, 1753, to December 31, 9999, with an accuracy of three-hundredths of a second, or 3.33 milliseconds.
- Char datatype is fixed-length non-Unicode character data with a maximum length of 8000 characters.
- Length property is the length of the datatype property for that particular column.

- For both of the table above, 'Allow nulls = Yes' means that particular column may accept the value null. In other words, users are not required to fill in the value for those columns, even though they are encourage doing so. For columns that do not allow nulls, users have no choice but to fill in the required information in the input form.

4.5 User Interfaces

The user interface design of a system is often the yardstick by which that system is judged. An interface, which is difficult to use will at best, result in a high level of user errors. At worst, it will cause the software system to be discarded, irrespective of its functionality.

In **OTMS**, the goal of user interface design is to provide an easier and faster way for the users to interact with the computer, or what is commonly known as human-computer interaction (HCI).

4.5.1 Screen Design of OTMS

As known, **OTMS** is a Web based application. So, the screen design of **OTMS** is presented in the form of Web pages. To generate a better and user friendly interface, OTMS's screens design are formatted in a standard layout so that various types of information, instructions and messages always appear in the same general display area. The standard layout will guide the users to use the system effectively. Beside that, the hyperlinks in this system provide quick and easy reference for the users.

Examples of user interfaces for **OTMS** can be found in the User Manual Section.

Chapter 5:

System Implementation

Chapter 5: System Implementation

5.1 Introduction

System implementation is a process that takes place after system design phase. It is a process to convert the system requirements into program codes. This phase describes how the initial and revised design was put into the real work.

Under this stage, we transformed the design model of the Online Title Management System into workable software. The system implementation of **OTMS** will be divided into two components, which are platform implementation and modules implementation.

5.2 Platform Implementation

The platform implementation includes setting up the operating system and database server, which are Microsoft Windows 2000 Professional and Microsoft SQL Server 7.0. It is very important to have suitable hardware and software in speeding up the system development and make a success to this project.

5.2.1 Setting up Microsoft Windows 2000 Professional

Before the system can be actually developed, it needs to run under Microsoft Windows 2000 Professional. During the installation of Windows 2000 Professional, the hard disk was formatted using NT File System (NTFS) format to ensure a more stable and secured NT transaction environment across the platform. The main reason to set up this operating system is because of the multi-users feature. It allows concurrent processes by different users.

5.2.2 Setting up Microsoft SQL Server 7.0

Microsoft SQL Server is installed in another personal computer. It is separated from the Internet Information Server (IIS). After the Microsoft SQL Server has been installed successfully, a database with the name "**OTMS**" is created. Then, I create the tables according to the database design (Section 4.4). This database will become the database storage for the whole system. The tables are created in order to keep the data used in all modules of the Online Title Management System.

I allocated the hard disk space for the database to maximize the performance of the SQL Server and to ensure that there is always enough space to store all the records. The file growth of the database is set to 10% of the original database size.

5.2.3 Other Software Tools Needed

Many software tools are also used to develop OTMS as shown below.

Software	Usage	Description
Microsoft Windows 2000 Professional	System requirement	Operating System
Internet Information Server 4.0	System requirement	Web Server
SQL Server 7.0	System requirement	Database Server
Internet Explorer 5.0	System requirement	Web Browser
Adobe Photoshop 6.0	System development	Graphic editing
Microsoft Visual InterDev 6.0	System development	Coding the Web pages
Microsoft Word 2000	System development	Documentation
Microsoft FrontPage 2000	System development	Interface graphical design and HTML editor.
Xara 3D 4.0	System development	Graphical design and animation.

Table 5.1: Other Software Tools Needed

5.3 Modules Implementation

Online Title Management System has three main modules:

- Administrator module
- Lecturer module
- Student/Visitor module

Each module is implemented using Active Server Pages (ASP) with the help of Microsoft Visual InterDev 6.0.

5.3.1 Administrator module

Administrator module is designed specifically for the system administrators of **OTMS**. With this module, administrators can maintain all data in the database besides keeping track of the registered users for the system.

Administrators can:

1. Browse/Edit registered users records
2. Browse/Edit thesis titles records

5.3.2 Lecturer module

Lecturer module is designed specially for lecturers who are involved with the WXES/T 3181: Latihan Ilmiah. With this module, lecturers can maintain their thesis titles easily. They can also compare the thesis titles with other lecturers without any complexity.

For the maintain thesis titles function, lecturers can maintain the following fields:

- Title
- Language
- Tools
- Descriptions
- Student name

- Matric number
- Session
- Semester

For the compare functionality, lecturers are required to input only the title or any keyword they wish to compare.

5.3.3 Student/visitor module

This module will allow the students or any other visitors to browse and search for thesis titles. For the searching functionality, users will be ask to provide some details such as:

- Title
- Lecturer name
- Language
- Tools
- Descriptions
- Department
- Session
- Semester

5.4 Coding of OTMS

5.4.1 HTML

HTML is the most common standard use for developing interface of Web application since it is essentially for a scripting language that marks up a Web page with formatting command and these commands are supported and interpreted by most Web browser. By using HTML, Web sites may include colorful and animated graphics, sound and video that make the Web page more attractive. In addition, HTML also used to generate form, which enable system to collect data from users. Buttons, text boxes, list boxes, check boxes, dropdown lists and radio button are all supported by HTML form. All these features are employed in **OTMS**.

5.4.2 Server Side Scripting

ASP is a server side scripting that is embedded in the HTML scripts. ASP codes are located within the delimiter <%-------%> in the HTML scripts. It is invisible to the client and only can be executed in the server side, hence it is called server side scripting. It is suitable to be employed in **OTMS**, which provides the capability for the Web server to process application logic and then deliver standard HTML to the client browser.

5.4.3 Client Side Scripting

Besides server side scripting, client side scripting such as VBScript and JavaScript are also embedded in the HTML code for **OTMS** development. Client side scripting is a type of scripting that does not need to be sent to server side for processing but only interpreted by user browser. As a result, it helps to reduce the network traffic problem since it reduces user requests that need to be sent to server and get response from the server. Another advantage of client side scripting is that it provides a better and quicker response to the user.

For **OTMS**, the client side scripting is use to do simple interactive tasks such as validate user's input and form completeness checking. An alert message will be prompted if the user's input is invalid or the form is not complete.

5.4.4 Database Connection

The database connection is an important step that must be done before we start the coding of Web pages that need to retrieve or insert data into the database. All important information such as database server, database name, user name as well as the password required to access the database must be specified correctly.

After configuration, the codes can directly connect to the server and communicate with the database. Figure 5.1 illustrates a simple example that shows the database connectivity.

```
<%
    dim objConn
    set objConn=Server.CreateObject("ADODB.Connection")
    objConn.Open "Provider=SQLOLEDB; Data Source=10.100.1.195;" & _
                 "Database=OTMS; User ID=wet990039; Password=fsktm"
%>
```

Figure 5.1: Sample Code of Database Connectivity

Chapter 6:

System Testing

Chapter 6: System Testing

6.1 Introduction

System testing is the analysis and validation of the product as a whole. System testing encompasses all product components, including both hardware and software. Its purpose is to verify that all essential functions and features are present and working properly. System testing often includes testing the system's performance, stability and response time, besides validating that all system components working as intended. System testing can also include a wide range of industry standard tests. These may be performed separately on each system component, jointly on multiple system components, or both.

Standard tests commonly use to validate a system may include, but are not limited to:

- Functionality testing to determine how well your product functions based on its stated design specifications.
- Performance testing to identify whether your product functions within the expected performance parameters, and to allow you to compare your product's performance against its competitors.
- Stress/load testing to verify your system's ability to successfully handle excessive amounts or challenging types of input data.
- Compatibility testing to validate your system's ability to function smoothly with each of the various hardware devices and software programs for which you designed and built it.

The goal of testing is to find out the errors and faults in the coding. Therefore, a systematically test procedure is needed to make sure that the system is tested thoroughly and completely. There are several stages of testing conducted in this project, it is shown in Figure 6.1 and each stage will be discussed in the sub chapters as following.

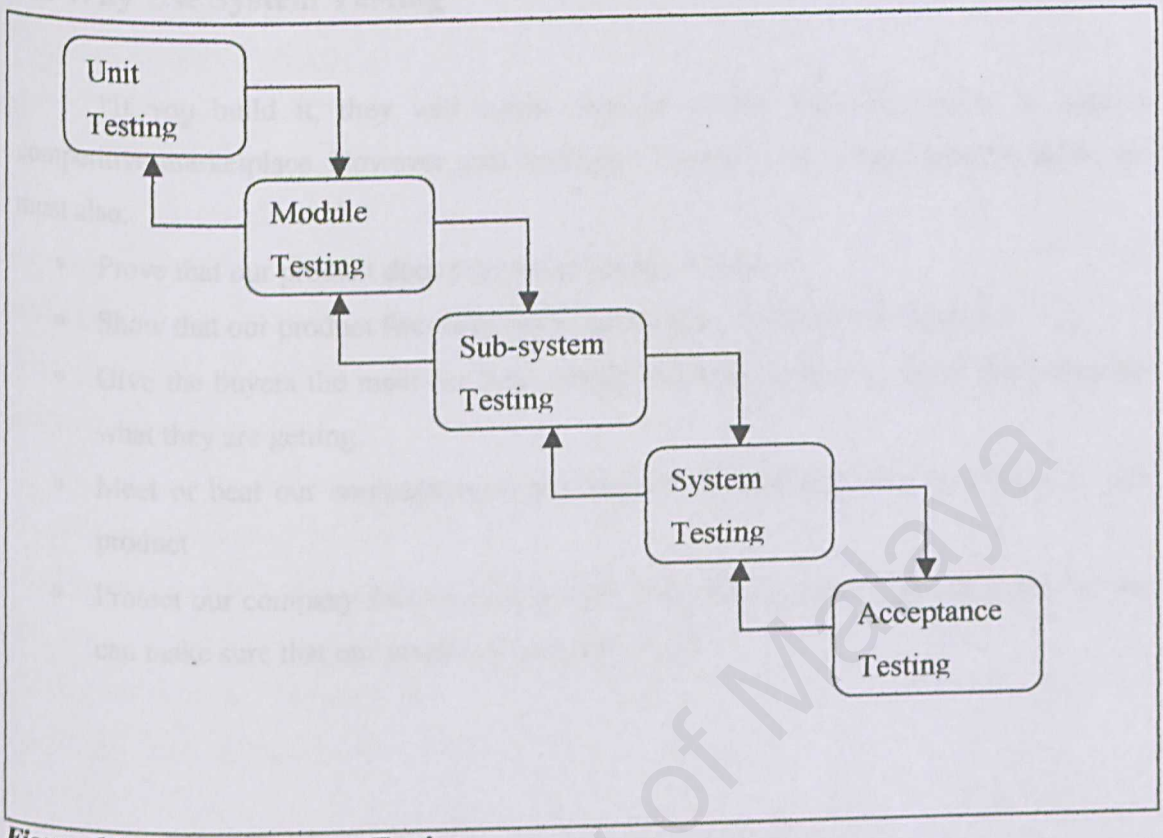


Figure 6.1: Stages in System Testing

6.2 Why Use System Testing

"If you build it, they will come," use to be the marketing norm in today's competitive marketplace. However, just building a product is no longer enough. Now, we must also:

- Prove that our product does everything we say it does
- Show that our product fits seamlessly into existing and planned networks
- Give the buyers the most for their money and make sure they know that's exactly what they are getting
- Meet or beat our competition in any number of tests that may be thrown at our product
- Protect our company from potential liability by taking every reasonable step so we can make sure that our product is safe and sound

6.3 Definition of Unit, Component and Integration Testing

Unit is the smallest compilable component. A unit typically is the work of one programmer (at least in principle). As defined, it does not include any sub-components (for procedural languages) or communicating components in general.

In unit testing, called components or communicating components are replaced with stubs, simulators or trusted components. Calling components are replaced with drivers or trusted super-components. The unit is tested in isolation.

Component is the integration of one or more units. The reason for "one or more" as contrasted to "two or more" is to allow for components that call themselves recursively.

Component testing is the same as unit testing except that all stubs and simulators are replaced with the real thing. Two components (actually one or more) are integrated when:

a) They have been compiled, linked and loaded together.
b) They have successfully passed the integration tests at the interface between them.

Thus, component A and B are integrated to create a new, larger component (A, B). Note that this does not conflict with the idea of incremental integration – it just means that A is a big component and B, the component added, is a small one.

Integration testing means carrying out integration tests. Integration test for procedural languages is easily generalized for Object Oriented languages by using the equivalent constructs for message passing. In the following, the word "call" is to be understood in the most general sense of a data flow and is not restricted to just formal subroutine calls and returns – for example, passage of data through global data structures and/or the use of pointers.

As to the difference between integration and system testing, system testing specifically goes after behaviors and bugs that are properties of the entire system as distinct from properties attributable to components (unless, of course, the component in question is the entire system). Examples of system testing issues include resource loss bugs, throughput bugs, performance, security, recovery, transaction synchronization bugs (often misnamed "timing bugs") and many others.

6.4 Unit Test

Unit test is the test of the software elements at the lowest level of development. Units may be aggregates of software elements. Planning for unit test should occur concurrently with the software design activity. Reused software will probably not undergo unit test, unless changes were made to these units. Then, appropriate testing is performed as in regression testing.

In general, the steps for unit test are:

1) Test planning – Establish the objectives of the unit test, the strategies to be employed, the coverage requirements, reporting and analysis, and closeout of anomalies.
2) Generate, monitor and update the unit test plan to accomplish objectives.
3) Trace test design, cases, procedures and execution results to the unit designs.

4) Confirm that anomalies during test are software anomalies and not problems detected for other reasons.
5) Generate test cases and procedures – Develop test cases and procedures for unit test and continue tracing as required by software test plans.
6) Perform unit test – Check individual software units for typographical, syntactic and logic errors to ensure that each correctly implements the software design and satisfies the software requirements; execute the test cases; analyze results to verify anomalies; recommend changes to software design or code; and conduct retesting as necessary.
7) Document test activities and results.

6.5 Software Integration Test (Module and Sub-system)

The software integration test activity is performed to examine how units interface and interact with each other with the assumption that the units and the objects (e.g., data) they manipulate have all passed unit tests [BEIZER, 1990]. Software integration tests check the interaction with other software (e.g., libraries) and hardware. The software integration test schedule depends upon the development and integration schedules for software units, hardware and other components. For large system, software integration test planning may require close coordination among all system personnel to ensure that the overall test objectives are achieved by the selected test strategy. For each major integration that has successfully undergone interface and interaction testing, functional tests may be developed and executed [BEIZER, 1990]. When all system components have been integrated and have successfully undergone software integration test, then the system moves into software system test. During software integration test, reduce software units are integrated into the system. It is critical to test that the interfaces are correct, and that the resulting software meets operating requirements.

General

- Test planning – Establish the objectives of the software integration test, the strategies to be employed, the coverage requirements, reporting and analysis, and close out of anomalies. Ensure that interface testing of reused software to other system software is planned.
- Generate, monitor, and update a software integration test plan to accomplish identified objectives.
- Trace test design, cases, procedures, and execution result to software requirements.
- Generate test cases and procedures – Develop test cases and procedures for unit test and continue tracing as required by software test plans.
- Perform software integration test.
  - Check the inter-unit communication links and test aggregate functions formed by groups of units.
  - Confirm that anomalies during test are software anomalies, and not problems detected for other reasons.
  - Ensure any changes to software requirements, software design, or codes are made. Conduct retesting as necessary.
  - Conduct functional, structural, performance, statistical, and coverage testing of successfully integrated units after each iteration of software integration and successful testing of interfaces and interactions.
- Document test activities and results.

6.6 System Test

System test, in the context of software V&V, involves the conduct of tests to execute the complete integrated system. Software system test is the validation that the software meets its requirements. Validation of the complete system may involve many tests involving all system components. The software system tests exercise those system functions that invoke software to determine whether the software behaves as intended relative to complete system performance. These tests must be conducted in such a manner as to stress the system based on software responses to system inputs (e.g., from sensors, operators, databases). Tests and data collected from the tests are designed to provide an operational profile of the system which support a statistical analysis of the system reliability [MUSA87, MUSA89, BUTLER]. This section of the report addresses only the tests that validate that the software implements the system requirements; other tests for other components and perspectives are necessary for complete system validation.

While software system tests are conducted after the system has been built, it is imperative that planning for these tests is conducted currently with the software requirements activity because:

- Analyzing the software requirements for test requirements may result in finding software requirements errors and/or discovery of non-testable requirements.
- Establishing test facilities (e.g., model of operational environment) and Computer-Aided Software Engineering (CASE) tools (e.g., test case generators, test database) may require as much time as development of the system.

For reused software, software system test is performed to assure that the software is correct, consistent with prior documentation, complete for use and/or modification, and accurate. At the system level, reused software should be considered part of the system. Tests are in accordance with test procedures. Results are documented and traced as required by the software system test plan.

General

- Test planning – Establish the objectives of the software system test, the strategies to be employed, the coverage requirements, reporting and analysis, and close out of anomalies.
- Generate, monitor, and update a software system test plan to accomplish objectives.
- Trace system and software requirements to test software design, cases, procedures, and execution results.
- General test cases and procedures – Develop test cases and procedures for unit test and continue tracing as required by software system test plan.
- Test the operation of the software as an entity (sometimes a simulated environment may be used); confirm that anomalies during test are software anomalies, not problems detected for other reasons; ensure any changes to software (software requirements, software design, code, or test cases) have been made; and conduct retesting as necessary.
- Document test activities and results.

6.7 Acceptance Test

This is the final stage in the testing process before the system is accepted for operational use. This usability testing is done and performed by the users. The choice of tests to be used was made by the users.

A user manual should be the source of the test. All of the errors found in the testing were reported to the developer of **OTMS**. Then, modifications can be carried out until the final release of the system.

6.8 Software Installation Test

The software installation test activity is the final step before launching full customer acceptance testing. The purpose of installation test is to demonstrate that the correct software has been delivered and that the software interfaces are correct relative to any interfaces at the installation site. Acceptance testing, which involves the user/customer, is outside the scope of this document.

General

- Conduct an installation configuration audit.
  - Determine that all software outputs needed to operate the system are present.
  - Check that the software installed in the system is the software that underwent software V&V.
- Develop and execute tests that will examine and stress site unique parameters (e.g., printer interface, operating system interface, monitor interfaces).
- Generate applicable documentation.
- Generate an SVVR (or generate it at the end of the software V&V process).

***Please refer to Appendix A for more information about system testing.**

Chapter 7:

System Evaluation

Chapter 7: System Evaluation

7.0 Introduction

After system testing for the Online Title Management System, the end product was brought up for system evaluation. After the system had been completed for the first time, it had been sent to several end-users to test it. There are many evaluation techniques that use to evaluate the final system. The purpose of the system evaluation is to highlight the strengths, limitations, constraints and possible future enhancement of the developed and completed system. The results gathered from the user testing stage are of a very good means of system evaluation. The following section will explain in detail about the system strengths and its limitations.

7.1 System Strengths

1. **User-friendliness**

The graphic interface design of the system was designed to let the users feel comfortable and easy-to-use. The GUI ensured user friendliness. Thus, the users should find it easy to use. The system will return error message if it detect inconsistencies.

2. **Paper-less**

Reducing paper use has been the attempt ever since the developments in computerize and much has been mentioned in earlier chapter of this documentation. By reducing paper, it saves long-term cost.

3. **Search Capability**

Searching plays an important role in database management. It helps user to store and retrieve data sufficiently and effectively. User could search the database for thesis titles.

4. Easy accessibility

This system could be accessed easily using a web browser where it could be downloaded free from any authorized web site such as Internet Explorer from http://www.microsoft.com. Moreover, access to the system is made available at all the time, 24 hours a day and 7 days a week.

5. Relatively fast response time

This system will be designed in such a manner that its web pages will be loaded in a reasonable amount of time. This is to ensure that users do not need to wait too long to view the pages. All usage of unnecessary components such as graphic files, multimedia files and others will be kept at the most minimum level to ensure the relatively fast response time.

6. Enforce different access levels

This system is expected to enable administrators to have a different access level from the other end users. This is to provide a better and secure environment among administrators. Beside that, lecturers will also reside in another access level. The purpose of this is to enable different users perform different tasks in the system. However, administrators will have the highest access level, followed by the lecturers and finally the students.

7. System validation

This system is expected to provide a validation system for its users. It will validate the user to be either authorized or unauthorized. Certain modules in the system can only be access by the authorized users. This validation system also makes sure that only the creator of the thesis titles will have the rights to modify the records (except for under certain circumstances, administrators will need to take over the task).

7.2 System Limitations

1. Limited Target Users

The target users for the system are lecturers and students from FSKTM as the main users. Administrators and any other visitors will be considered as minority users. Due to the limitations of time and human resources, this system will be used in FSKTM only with no integration or link with other faculties. This means any other faculties in University of Malaya that provide similar thesis paper for their students will not be able to use the system.

2. Limited to Subject WXES/T 3181

OTMS will only be use for the subject WXES/T 3181: Projek Ilmiah I. The system cannot be used for other subjects included in the Master or PhD levels offered by FSKTM. However, expansion can be made in the future if necessary.

3. Advance booking not allowed

Advance booking or reservation for thesis titles cannot be made through the system. This feature will not be considered to be implement into the system to ensure that all students will have the same opportunity to get their titles respectively and thus the unfairness competition among students can be avoided.

7.3 Future Enhancement

Some functionality of the system can be enhanced in order to improve the quality of the system. The following are the functionality that can be enhanced o this system.

1. Maintenance of User Interface

The GUI should be updated after a certain period to give the user a fresher and greater view.

2. No Online Help Facilities

OTMS does not provide any online help facilities. Users who require more information may not satisfy with the brief instruction and description. It should be considered to enhance in the future.

7.4 Problems Encountered And Solutions

Throughout the development of **OTMS**, a number of problems were encountered. Below are some of the problems found during the development of **OTMS** and the solutions of it.

1. Problem In Development Tools And Languages Selection

After the system analysis was done, I had the difficulties to select the most appropriate tools and languages in system development. This is because I myself have the limited knowledge of the Web based programming.

Solution: To overcome this problem, I have tried to gain more information of web-based programming and identified the most appropriate approach to develop **OTMS**. Depth studies and research on the Web-based programming language was carried out in the earlier

stage of the development. Besides, discussions with course mates and opinion from seniors was collected and all of these help to finalize the development tools and languages.

2. Lack Of Knowledge In Language

The learning and developing process was carried out in parallel due to the time constraint. Without a strong base of the languages such as ASP scripting, JavaScript and VBScript, I have to spend a lot of time looking for solutions of the problem occurred during the system development. This was usually happen to cases related to concepts of language that were new for me.

Solution: The difficulty is solved through reference books, tutorial downloaded from Internet and also the assistance of my course mates.

3. Difficulty In Designing User Interface

It is a difficult task to develop suitable, standard and systematic user interface because I have no experience in such a development. The lack of experience in designing standard layout caused the arrangement of the control and graphics on the Web pages has eaten up a lot of time.

Solution: To solve this problem, real time Web sites have been taken as reference to gain more information on the layout of GUI (Graphical User Interface). Adobe Photoshop and Xara 3D 4.0 were used to generate attractive images and hence improve the user interface of **OTMS**.

Final Conclusions

The purpose of this report is to document essential information gathered and implemented throughout the development of the project. It covers the project literature review and analysis and also partial of the design of the software. This report is divided into four chapters, which are described below:

Chapter 1: Introduction

This section gives a general overview of the project, its' objectives, scopes, strengths, limitations, significance, etc. This section also consists of the project planning and scheduling to smoothen the project and get the expected outcome.

Chapter 2: Literature Review

This section explains the studies done, from analysis to synthesis, for the **OTMS**. It includes details of the fields that the studies has involved. It covers the survey on current conventional system and several online systems. It too describes several methods and considerations on programming tools used, programming languages, database implementation, operating systems preferences, Web server selection, and other technologies that are related to **OTMS**. Several strategies or methodologies to develop a successful system are also included.

Chapter 3: System Analysis

This section describes the analysis that has been done since the project started. It also includes functional and non-functional requirements of the system based from the requirement analysis in Waterfall methodology (with prototyping). This chapter also concludes the programming languages, software and hardware requirements as well as the methodology used for this project.

Chapter 4: System Design

This section describes the considerations for the input design, output design (user interfaces) and database implementation into the system development. It also includes a general system design and the more detailed Data Flow Diagram of **OTMS**.

Chapter 5: System Implementation

This section describes the processes to convert the system requirements into program codes and also how the initial and revised design was put into the real work. Under this section, the design model of **OTMS** is transformed into workable software.

Chapter 6: System Testing

This section describes about the processes involve in system testing. Each type and level of system testing is explained in detailed.

Chapter 7: System Evaluation

This final section covers the system evaluation in general. The final system was tested to determine its strengths, limitations, future enhancement, problems encountered and their solutions and so on.

As a conclusion, **OTMS** will be design as a Web based system with three different modules, each for a different group of target users. The three modules are Administrator Module, Lecturer Module and Student/visitor Module. The main objective of **OTMS** is to help the lecturers in comparing the thesis titles among each other to avoid duplication. Other objectives include:

i. *Manage the thesis titles*
ii. *View the thesis titles*
iii. *Avoid students from registering with more than a lecturer*
iv. *Provide better information to meet the students' need*
v. *Create a paperless environment*
vi. *Save students' time*

OTMS will be develop using ASP, HTML and VBScript. The development tool for this project is Microsoft Visual InterDev 6.0. It will also include a Web based database implemented using Microsoft SQL Server 7.0. Other software requirements are Microsoft Windows 2000, Internet Information Server 4.0, and Internet Explorer 5.0. The methodology chosen for the development of **OTMS** is waterfall model with prototyping.

Upon completion of this project, lecturers will have a more convenient system for managing their thesis titles. In the meantime, students will also benefit from the system in sense that they can get more information about thesis titles easily.

References

[1] Amundsen, M. (1999). Practical Visual InterDev 6. Indiana: QUE.

[2] Bannan, J. (1997). Intranet document management: A guide for webmasters and content providers. Addison-Wesley Developers Press.

[3] Kendall, K.E., & Kendall, J.E. (1999). System Analysis and Design (4th ed.). United States of America: Prentice Hall.

[4] Perry, G. (1998). Sams teach yourself Visual Basic 6.0 in 21 days. Sams Publishing.

[5] Pfleeger, S. L. (2001). Software Engineering: Theory and Practice (2nd Ed.) New Jersey: Prentice Hall.

[6] Rob, P., & Coronel, C. (1995). Database system design: Implementation and management. (2nd ed.). Boyd and Fraser Publishing.

[7] Royce, W. W. (1970, August). Managing the development of large software system: Concept and techniques. In Proceedings of WESCON.

[8] Schneider, G.P., & Perry, J.T. (2000). Electronic commerce. Cambridge: Course Technology – ITP.

[9] Sellappan, P. (1999) Access 2000 (1st Ed.). Selangor: Federal Publication.

[10] Sommerville,I. (2001) Software Engineering (6th Ed.) Addison Wesley.

[11] Whitten, J.L., Bentley, L.D., & Dittman, K.C. (2001). System analysis and design methods (5th ed.). New York: McGraw-Hill.

[12] Wong, S. L. (1998). The Malaysia Internet. Selangor: LONGMAN.

[13] Wyatt, A. L., & Pease, C. (1998). Internet Information Server 4.0: Administrator's guide. Prima Publishing.

[14] Buku Panduan Fakulti Sains Komputer dan Teknologi Maklumat, Universiti Malaya, Sesi 1999/2000.

[15] SQL Server 7.0 Books Online

[16] http://dtp.epsb.net/index.htm

[17] http://intranetjournal.com/reference/library/

[18] http://windowsmagazine.com/help/2000/hot_or_hype/03.htm

[19] http://www.computing.net/windows95/wwwboard/forum/5019.html

[20] http://www.fsktm.um.edu.my/~ilmiah/main.html

[21] http://www.hhkaravan.com/microsoft_access.html

[22] http://www.microsoft.com

[23] http://www.mkdata.dk/click/w98/01.htm

[24] http://www.perl.com

[25] http://www.saaconsultants.com/Extranet_Advice_Centre/Benefits_of_EC.html

[26] http://www.utoronto.ca/writing/litrev.html

[27] http://www.w3schools.com/asp/asp_intro.asp

[28] http://www.winplanet.com/winplanet/opinions/574/3/

[29] http://thesis.mit.edu

[30] http://www.lib.nsysu.edu.tw/eThesys/default_e.htm

[31] http://www.theses.org/vt.htm

[32] http://elib.uni-stuttgart.de/opus/doku/english/index.html

[33] http://www.dissertation.com/dps07a.htm

[34] Implementing A Database On Microsoft SQL Server 7.0 CD

[35] Mastering Microsoft Visual Basic 6.0 Development CD

[36] Mastering Microsoft Visual Basic 6.0 Fundamental CD

[37] Mastering Web Application Development Using Microsoft Visual InterDev 6.0 CD

[38] System Administration for Microsoft SQL 7.0 Server CD

[39] Teach Yourself Microsoft Visual InterDev in 21 Days CD

Appendix A

Appendix A: Testing

A1. Unit Test

Test Case 1:

Before a user can access certain sites of **OTMS,** such as the administrator section and lecturer section, he/she must have a valid User ID and password. Only then, he/she can login to the system. The following table shows the testing procedures to check the User ID and it's corresponding password.

No.	Test Procedure	Output	Analysis of Test Result and Solution
1	Insert a valid User ID and password into the login page.	Able to login to the system successfully.	Successfully login to the system. Objective is to be able to login to the system.
2	Insert a valid User ID and invalid password.	Login is denied.	Login is denied. User is redirected back to the login page to try again. Objective is to prevent any unregistered user from login to the system.
3	Insert an invalid User ID.	Login is denied.	Login is denied. User is redirected back to the login page to try again. Objective is to prevent any unregistered user from login to the system.
4	Case 2 or 3 is repeated more than once.	Login is denied.	Login is denied until the correct User ID and password is provided. Objective is to prevent any unregistered user from login to the system.

Table A1: Test Case for OTMS Login Function

Test Case 2:

One of the functions in lecturer module is to let the lecturers to add a thesis title into the database. In other words, lecturers can enter whatever they want by themselves.

No.	Test Procedure	Output	Analysis of Test Result and Solution
1	Click on the "Add New Title" link on the left hand side of Lecturer Section's Main Page.	Form displayed.	The form is displayed correctly. Objective is to test whether the hyperlink is working properly.
2	Fill in all the required fields and click the "Add Title" button.	The record is inserted into the database.	Record is inserted into the database successfully. Objective is to insert a new record into the appropriate table correctly when the "Add Title" button is clicked.
3	Does not fill in all the required fields and click the "Add Title" button.	Message box prompted user to enter all the required fields.	A message box is prompted to remind user to fill in all the required fields. The form is not processed and no record is inserted into the database. Objective is to ensure all the required fields are not blank.
4	Fill in all the required fields with existed record from the database.	Could not insert the record into the database.	Checking is done to ensure there will be no duplicated record in the database. User is notified of the error and they can re-insert a new record. Objective is to prevent users from entering two identical records.

Table A2: Test Case for Adding New Thesis Title

Test Case 3:

One of the functions in lecturer module is to let the lecturers to compare a thesis title with all the records in the database.

No.	Test Procedure	Output	Analysis of Test Result and Solution
1	Click on the "Compare Title" link on the left hand side of Lecturer Section's Main Page.	Form displayed.	The form is displayed correctly. Objective is to test whether the hyperlink is working properly.
2	Fill in the text box with a single word and click the "Compare" button.	Comparison result.	All records that contain the keyword are displayed. Objective is to check whether the comparison result is correct.
3	Fill in the text box with multiple words and click the "Compare" button.	Comparison result.	All records that contain the multiple words and any single word (from the multiple words) are displayed. Objective is to check whether the comparison result is correct.
4	Does not fill in the text box and click the "Compare" button.	Comparison result.	All records are displayed. Objective is to check whether the comparison result is correct.

Table A3: Test Case for Compare Thesis Title

A2. Integration Test

Test Case 1:

The first integration is done on to the Lecturer Module. All units for Lecturer Module are integrated and tested.

*Assumption: The lecturer is a registered user.

No.	Test Procedure	Output	Analysis of Test Result and Solution
1	Login as a valid user.	Users are redirected to the Lecturer Section main page.	Users are redirected to the Lecturer Section main page.
2	Browse all thesis titles by clicking the "Browse Title" link.	All titles are displayed in a well-organized format.	All titles are displayed (if the database contains any record) or else a message will notified there is no title in the database.
3	Add a new thesis title.	Refer to Test Case 2 of unit test.	
4	Click the "Browse Title" link.	All titles are displayed in a well-organized format.	All titles are displayed, including the previously added title. It means that both the "Browse Title" and "Add New Title" functions are correct.
5	Click the "Find Titles" link.	Form displayed.	The form is displayed correctly. Objective is to test whether the hyperlink is working properly.
6	Type in the previously added title and click the "Find Record" button.	The title is displayed.	The title is displayed. It means that both the "Find Title" and "Add New Title" functions are correct.
7	Click on one of the links to edit that particular title.	Form displayed.	A form that contains the information of that particular title is displayed correctly.
8	Edit the information of that particular title and click the "Update" button.	Users are redirected back to the "Find Title" or "Browse Title" page.	All titles are displayed, including the previously updated title. It means that the "Upadate Title" function is correct.

9	Click on one of the links to delete the particular title.	Form displayed.	A form that contains the information of that particular title is displayed correctly.
10	Click the "Delete" button.	Users are redirected back to the "Find Title" or "Browse Title" page.	All titles are displayed, excluding the previously deleted title. It means that the "Delete Title" function is correct.
11	Click the logout link.	Users are redirected to the Home Page of **OTMS**.	Users are redirected to the Home Page of **OTMS**. Session variables are destroyed.
12	Attemp to access any pages of the Lecturer Section by typing the corresponding URL.	Users are redirected to the login page of **OTMS**.	Users are redirected to the login page of **OTMS**. Users cannot access any of pages in the Lecturer Section without login. This is done by Session variables.

Table A4: Test Case for Title Maintenance Function

A3. System Testing

Three end users were selected to test the system. They were given a chance to access all the modules in **OTMS**. After they test the system, they were asked a few questions to gather some feedbacks. Because of time limitation, the system testing was carried out for a short time involving only three users. Complete testing involving lecturers and administrators cannot be carried out.

From the system testing, all three users agreed that the system is user friendly. This is because of the all the links and buttons are very clear and easy to locate. Moreover, users are required to fill in only certain fields (marked with *) when using forms, while the rest are optional. This can avoid users from spending a long time to fill in all the available fields that are not compulsory.

All the users also agreed that they understand the error messages prompted to them by the system. Most of them received the pop up error message while trying to submit an

incomplete form. Other than that, when a particular function is successfully performed, the notification is clearly stated.

Two users said that the user interface does not really attract them. The interface is still lacking behind when compared to most of the commercial Web sites. However, I considered that **OTMS** is a formal Web site that lets the lecturers to maintain their thesis titles and students to browse for available titles. Therefore, not many pictures, graphics, animations and fancy design are included in the interface as it will take a long time to download and slow down the performance of the whole system.

All the users also agreed that **OTMS** is quite secured in the sense that it only lets registered users to access certain section of the system. However, they think that the system is still lack of the "Forgot Password" function. I explained that users who forgot their password could always approach the administrator of **OTMS** to assign a new password for them.

Beside that, one of the users complained that there was no online help provided. However, all of them agreed that the response time of **OTMS** is relatively fast and they can get all the information they need without any hassle.

Overall, users are quite satisfied with the system's performance. Some modification will be made according to their suggestions. However, I felt that the system testing is incomplete because some technical aspect of the system cannot be tested without involvement from lecturers and administrators.

User Manual

User Manual: Table of Contents

User Manual: List of Figures

User Manual: List of Tables

Chapter 1: Introduction

Online Title Management System (**OTMS**) is an easy to use database driven Web application that supports lecturers in storing their thesis titles for WXES/T 3181 via online. Basically, it consists of three different sections, which will be explained in detail in the following chapters.

OTMS gives the opportunity to the administrators to maintain the user registration and all the thesis titles. On the other hand, lecturers can compare and maintain their own titles while students/visitors can browse and search for titles.

For security reason and management use, **OTMS** will only allows administrators and lecturers who have registered with the system to access either the Administrator Section or Lecturer Section. However, students and visitors can still browse and search the website freely through the Student/Visitor Section which does not requires users to login.

OTMS is developed to be user friendly, which means it is easy to learn and use. All the functions in **OTMS** can be easily executed with just a simple click on the appropriate button or hyperlink.

The main purpose of this User Manual is to serve the users of **OTMS** with a tour guide through all the functions available in the system. This User Manual consists of four different sections to provide a complete guide for all categories of users:

- Getting Started
- Administrator Section
- Lecturer Section
- Student/Visitor Section

Chapter 2: Getting Started

2.1 Home Page of OTMS

OTMS is a Web application that provides services to the users. Therefore, users can simply access the Internet to reach the home page of the system. All the tasks or functions can then be perform through the Web browser. Current URL for the home page of **OTMS** is located at http://10.100.1.195/wet990039/OTMS/mainMenu.asp. Figure 2.1 below illustrates the home page of **OTMS**.

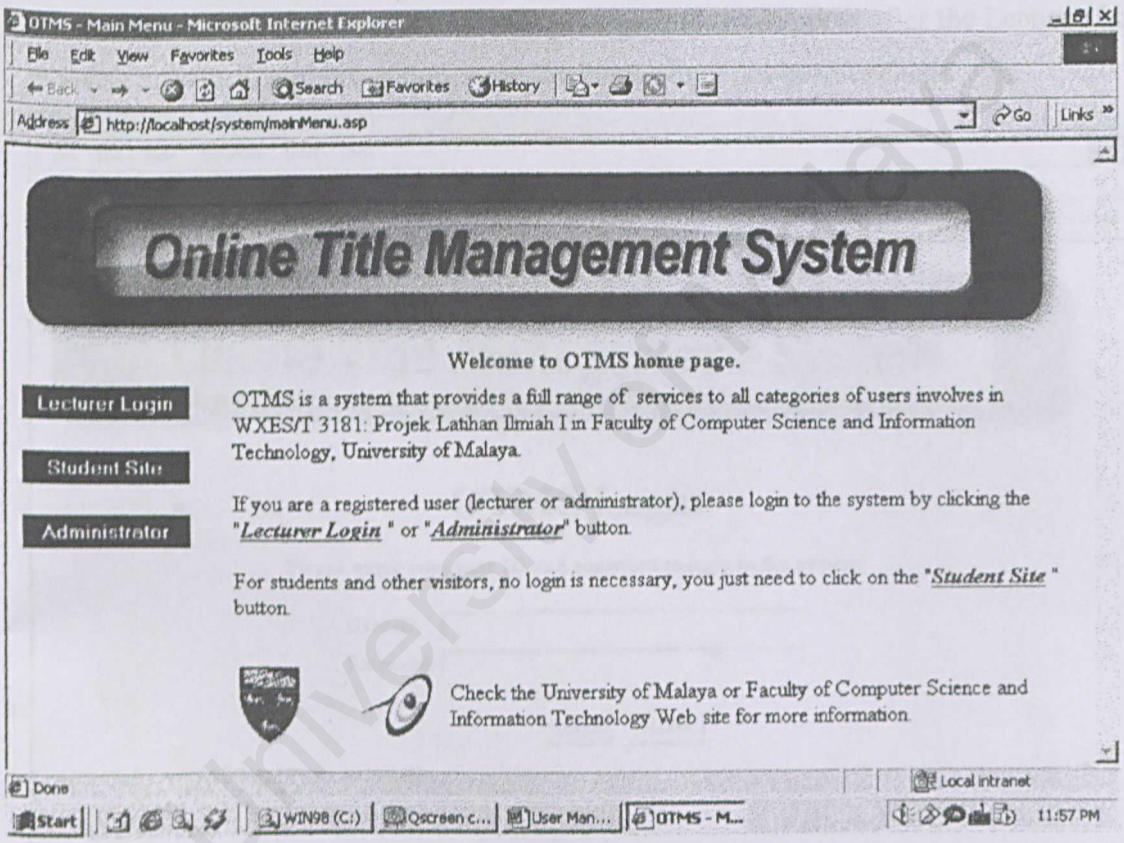


Figure 2.1: Home Page of OTMS

The home page displays a short description of the **OTMS**. Three hover buttons are located at the left hand side of the home page. Each hover button will redirect the users to different section. Table 2.1 describes the function of each hover button in detail.

Hover Button	Function
Lecturer Login	Redirect users to the login page of **OTMS**
Student Site	Redirect users to the Student/Visitor Section
Administrator Login	Redirect users to the login page of **OTMS**

Table 2.1: Function of each hover button in home page

2.2 Login Page of OTMS

Figure 2.2 below illustrates the login page that will appear after the Lecturer Login or Administrator Login hover button (in the home page) has been clicked.

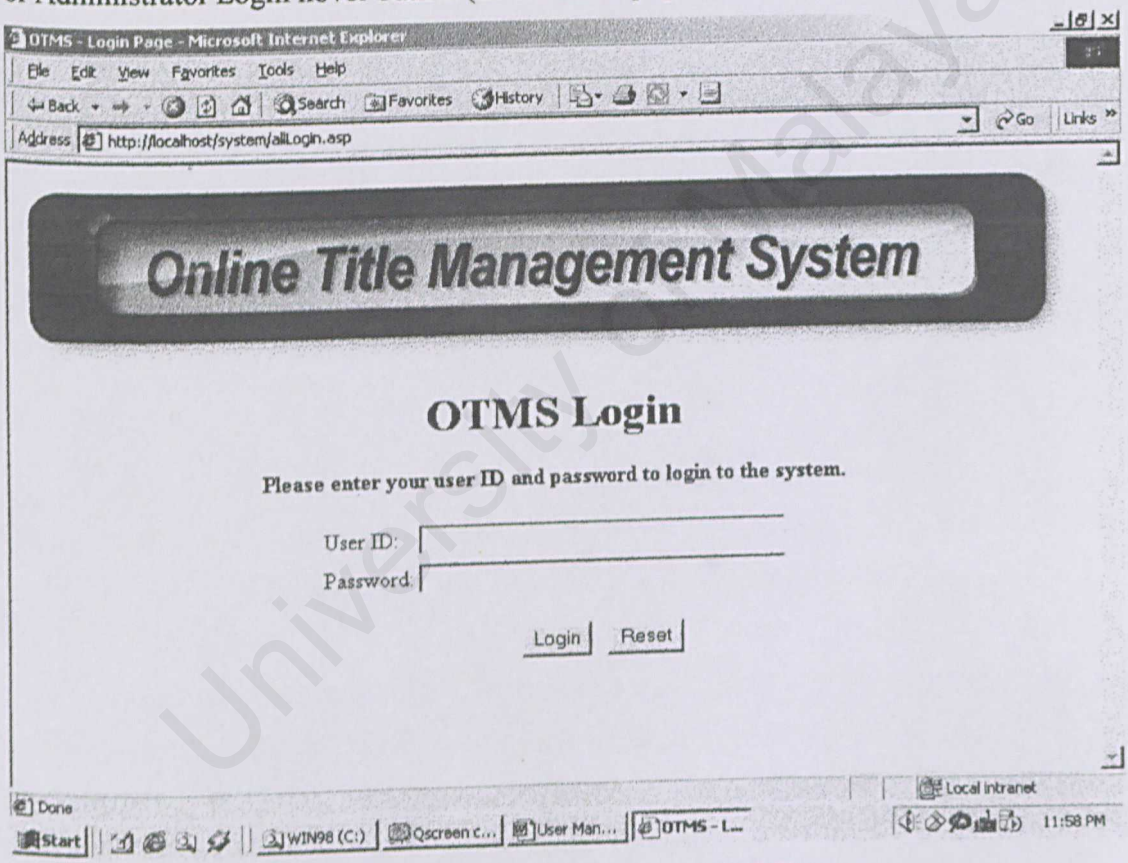


Figure 2.2: Login Page of OTMS

To login to the system, users will need to type in their User ID and corresponding password. After that, just click on the Login button to let the system automatically verifies

the User ID and password. Upon successful login, users will be redirect by the system to their corresponding section, which is either the Lecturer Section or Administrator Section. The function of the Reset button is to clear all the information that has been typed into the text boxes.

Chapter 3: Administrator Section

3.1 Main Page of Administrator Section

Figure 3.1 below illustrates the main page of the Administrator Section.

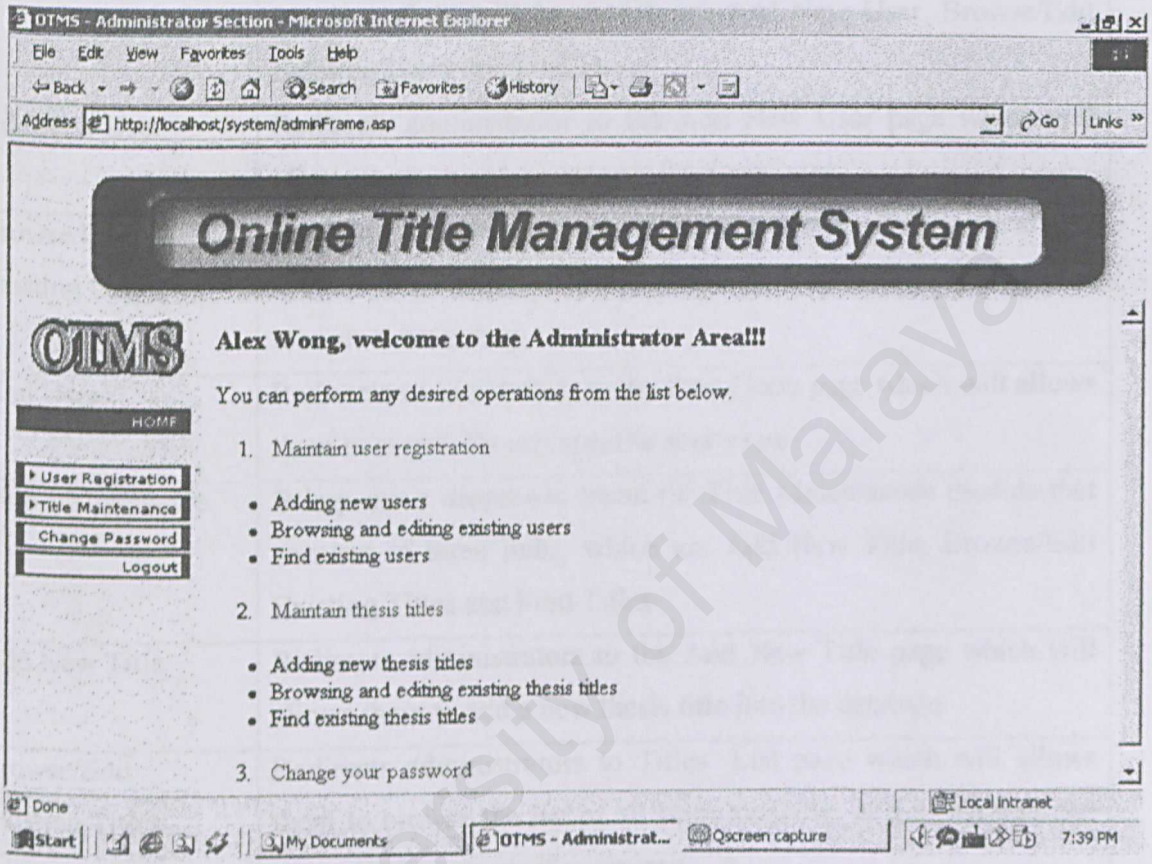


Figure 3.1: Main Page of Administrator Section

A short description about all the functions and modules in that particular section is also available. Almost identical with the home page of **OTMS**, all the links to various modules in the Administrator Section are also available on the left hand side of the page. Table 3.1 describes the function of all the links in detail.

Link	Function
Home	Brings out a new window which consists of the home page of **OTMS**
User Registration	Brings out a dropdown menu for User Registration module that consists of three links, which are Add New User, Browse/Edit Existing Users and Find Users
Add New User	Redirects administrator to the Add New User page which will allows them to add a new user for the system
Browse/Edit Existing Users	Redirects administrators to Users' List page which will allows them to browse the list of all registered users and then edit the records if necessary
Find Users	Redirects administrators to the Find Users page which will allows them to search for any specific user/users
Title Maintenance	Brings out a dropdown menu for Title Maintenance module that consists of three links, which are Add New Title, Browse/Edit Existing Titles and Find Titles
Add New Title	Redirects administrators to the Add New Title page which will allows them to add a new thesis title into the database
Browse/Edit Existing Titles	Redirects administrators to Titles' List page which will allows them to browse the list of all titles available in the database and then edit the records if necessary
Find Titles	Redirects administrators to the Find Titles page which will allows them to search for any specific title/titles
Change Password	Redirects administrators to the Change Password page which will allows them to change their password respectively
Logout	Logout from the system and then redirects administrators back to the home page of **OTMS**

Table 3.1: Function of each link in Administrator Section main page

3.2 Home

- Click on the Home link will bring out a new window which consists of the home page for **OTMS**.

3.3 User Registration Module

There are a total of three pages under this module, which are Add New User page, Browse/Edit Existing Users page and Find Users page. Figure 3.2 below illustrates the Add New User page.

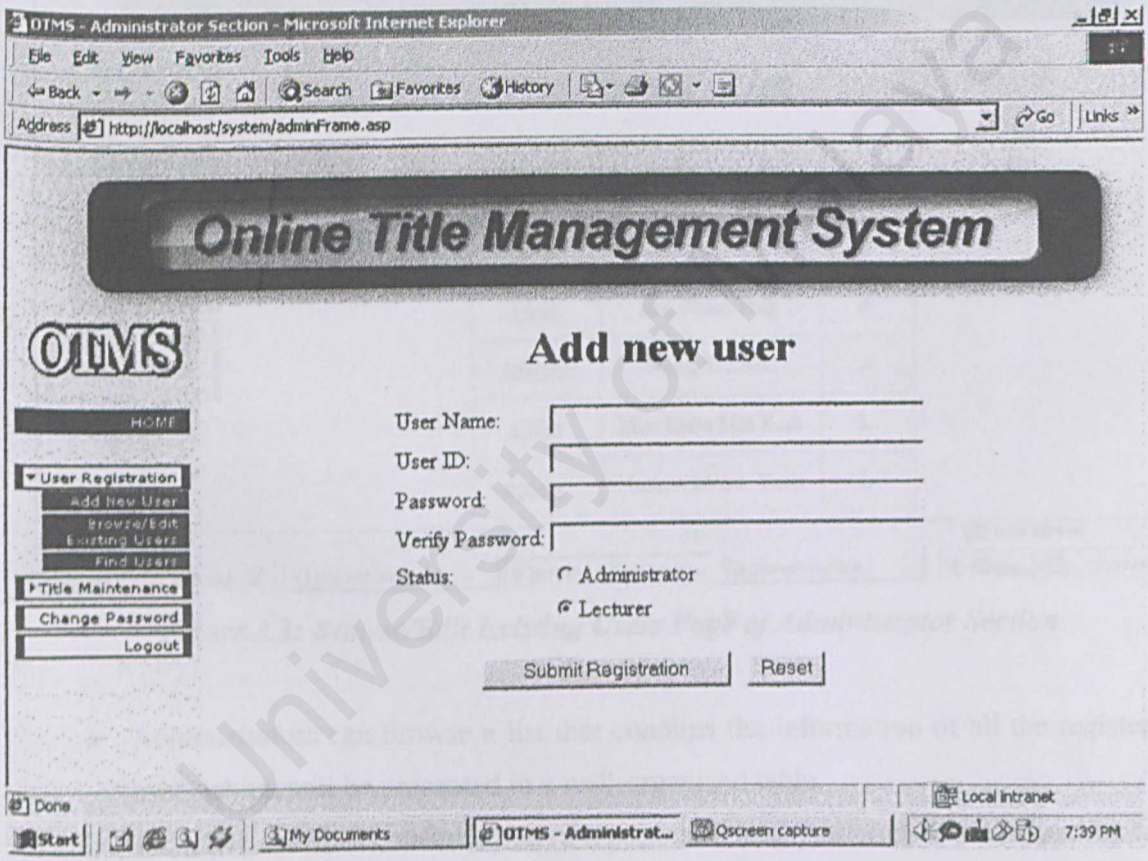


Figure 3.2: Add New User Page of Administrator Section

- Administrators will have to enter all the relevant information (User Name, User ID, Password, Verify Password and Status) before they can click the Submit Registration button to complete the new registration process.

- Other wise, administrators can click the Reset button to clear all the information that they have keyed in.

Figure 3.3 below illustrates the Browse/Edit Existing Users page.

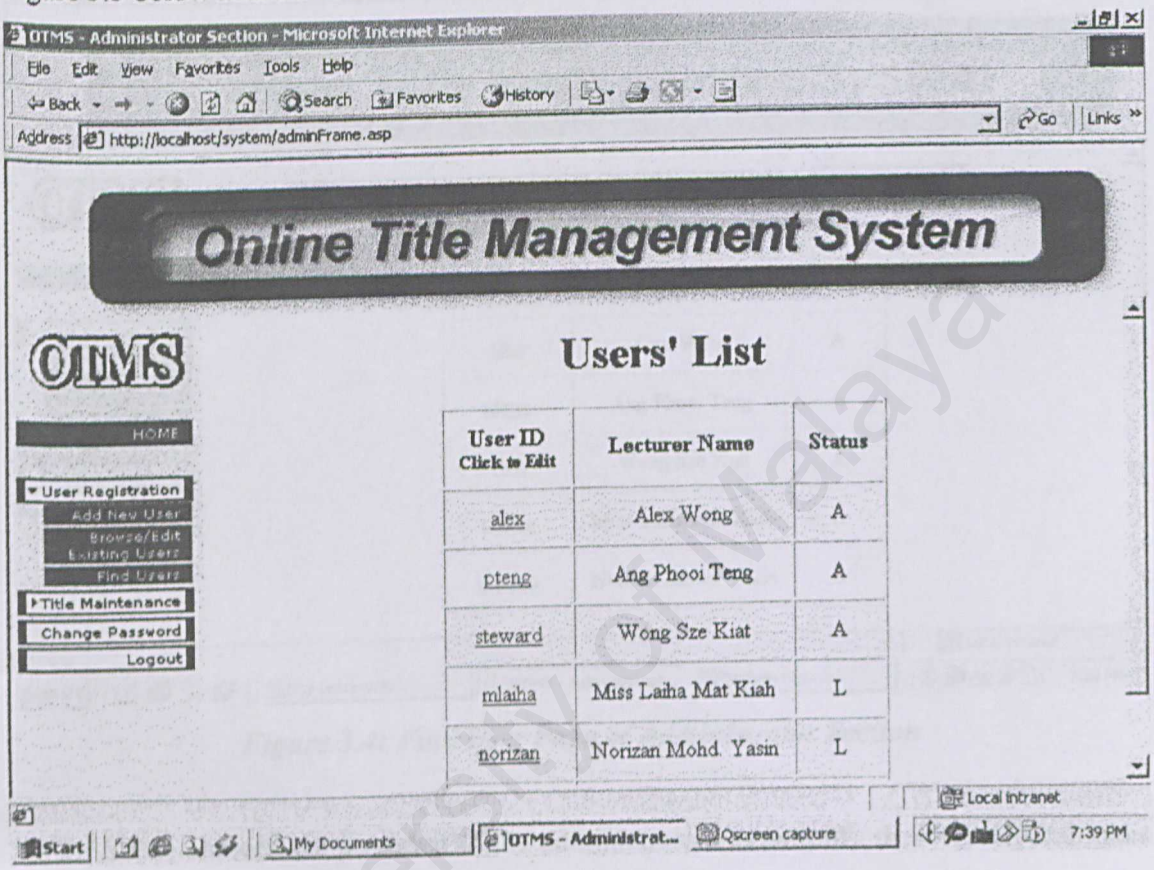


Figure 3.3: Browse/Edit Existing Users Page of Administrator Section

- Administrators can browse a list that contains the information of all the registered users which will be presented in a well-organized table.
- To delete or edit the information of a particular user, just click on the hyperlink text available in the table to link to the Edit User page.

Figure 3.4 below illustrates the Find Users page.

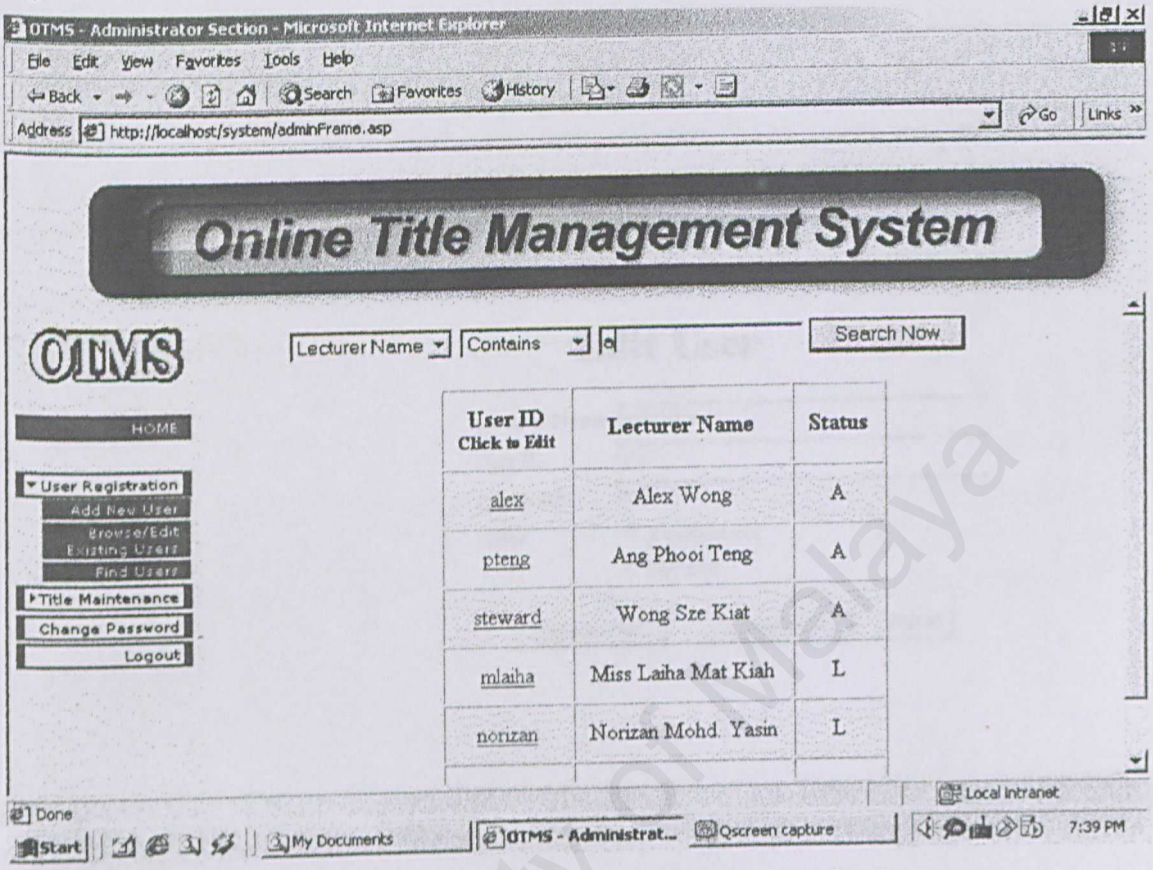


Figure 3.4: Find User Page of Administrator Section

- Some choices are provided to the administrators to simplify their search and make the results more precise. The choices can be change by selecting the appropriate dropdown lists.
- By clicking the Search Now button, the search result will be displayed in a tabular format.
- Administrators can also click on the hyperlink text in the table to direct them to the Edit User page.

Figure 3.5 below illustrates the Edit User page.

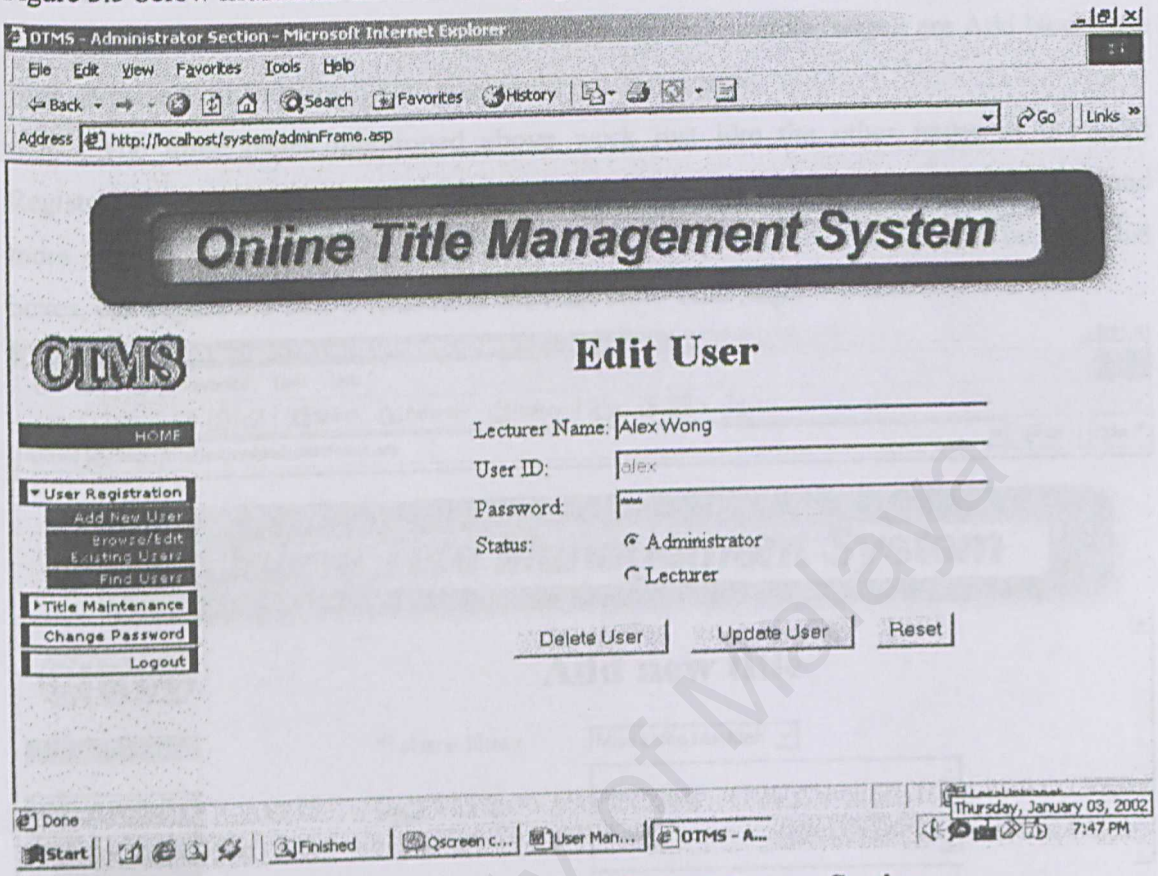


Figure 3.5: Edit User Page of Administrator Section

- Administrators can delete or update specific registered user's information through this page. Only Lecturer Name, Password and Status can be updated.
- Click on Delete User button will deletes the current record and then redirects administrators back to the Browse/Edit Existing Users page or Find User page.
- Click on Update User button will updates the current record and then redirects administrators back to the Browse/Edit Existing Users page or Find User page.
- Click on Reset button will clears all the information typed in by the administrators and restores it back to the original information.

3.4 Title Maintenance Module

There are a total of three pages in this particular module, which are Add New Title page, Browse/Edit Existing Titles page and Find Titles page.

The three pages mentioned above work just like the other pages in the User Registration Module. The main differences are more information needs to be fill in and more choices are available through the utilization of dropdown lists, radio buttons, list boxes, etc. Figure 3.6 below illustrates the Add New Title page.

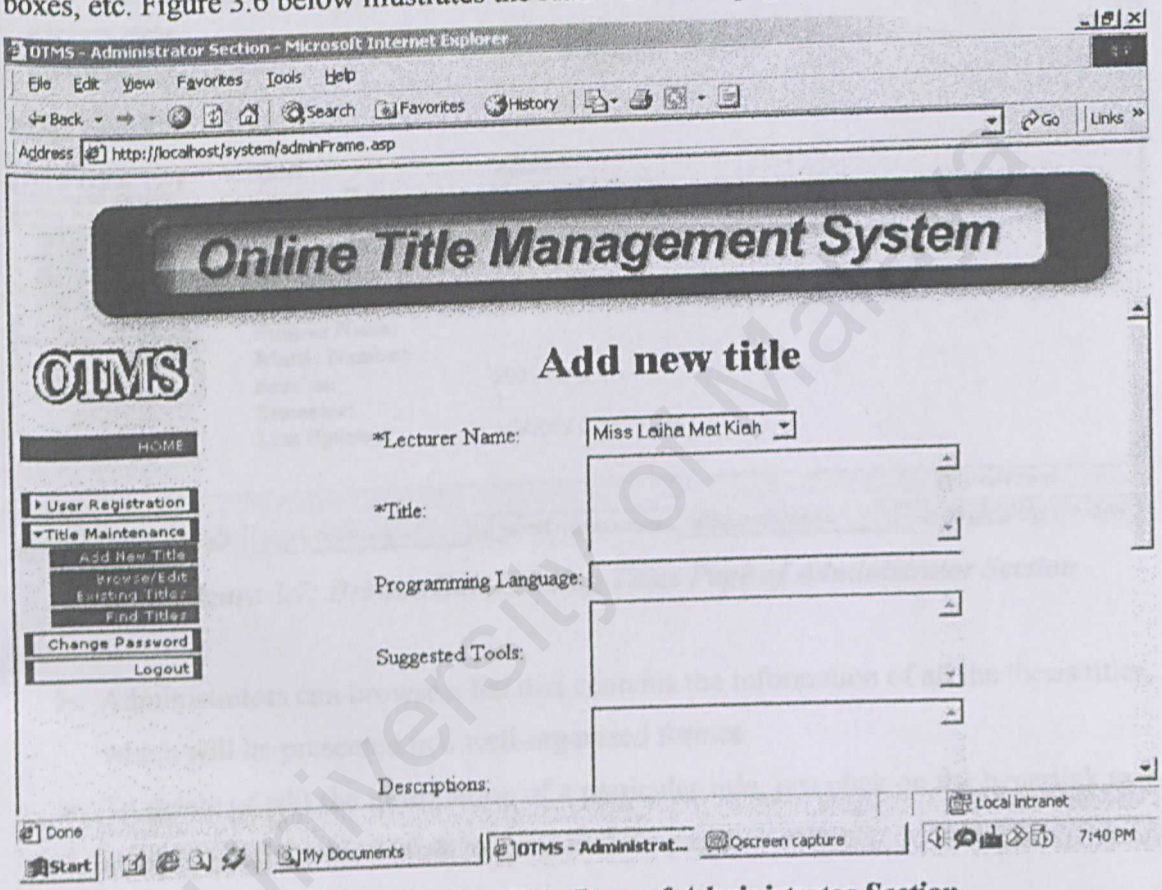


Figure 3.6: Add New Title Page of Administrator Section

- Administrators will have to enter all the relevant information (Lecturer Name, Title, Department, Session and Semester) before they can click the Add Title button to complete the process.
- Other wise, administrators can click the Reset button to clear all the information that they have keyed in.

Figure 3.7 below illustrates the Browse/Edit Existing Titles page.

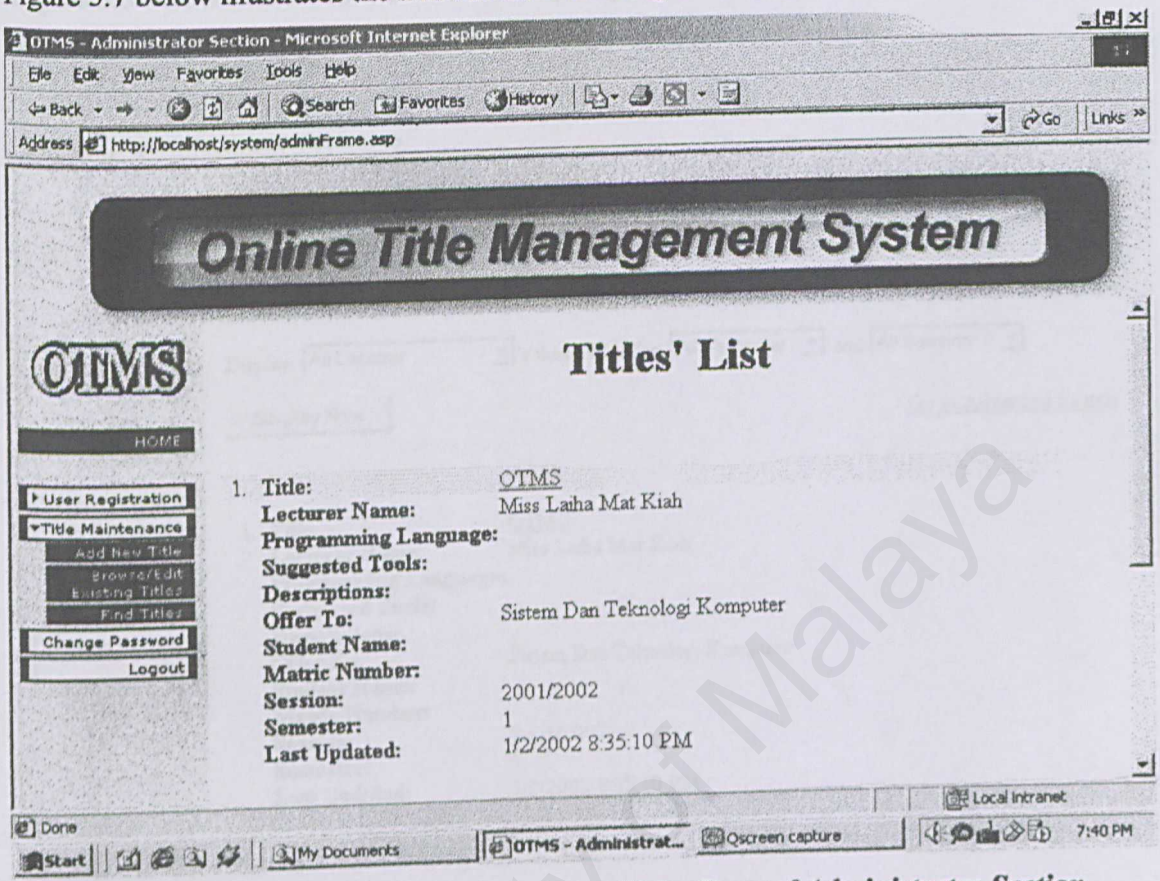


Figure 3.7: Browse/Edit Existing Titles Page of Administrator Section

- Administrators can browse a list that contains the information of all the thesis titles, which will be presented in a well-organized format.
- To delete or edit the information of a particular title, just click on the hyperlink text available in the table to link to the Edit Title page.

Figure 3.8 below illustrates the Find Titles (Simple Search) page.

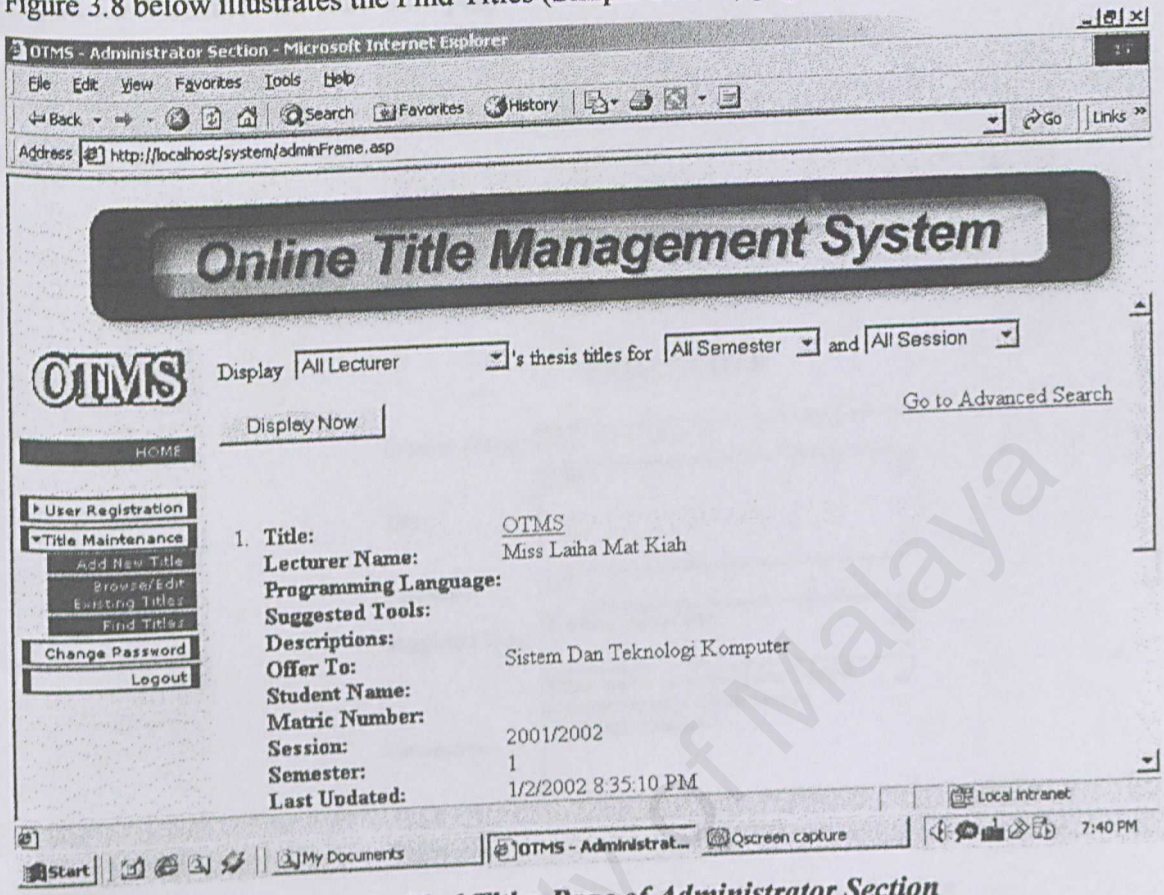


Figure 3.8: Find Titles Page of Administrator Section

- Some choices are provided to the administrators to simplify their search and make the results more precise. The choices can be change by selecting the appropriate dropdown lists.
- By clicking the Display Now button, all titles of a particular lecturer's will be displayed in a well-organized format.
- By clicking the "Go to Advanced Search" link on the right hand side, the advanced search page will be displayed.
- Administrators can also click on the hyperlink text in the results in both simple and advanced search to direct them to the Edit User page.

Figure 3.9 illustrates the Edit Title page.

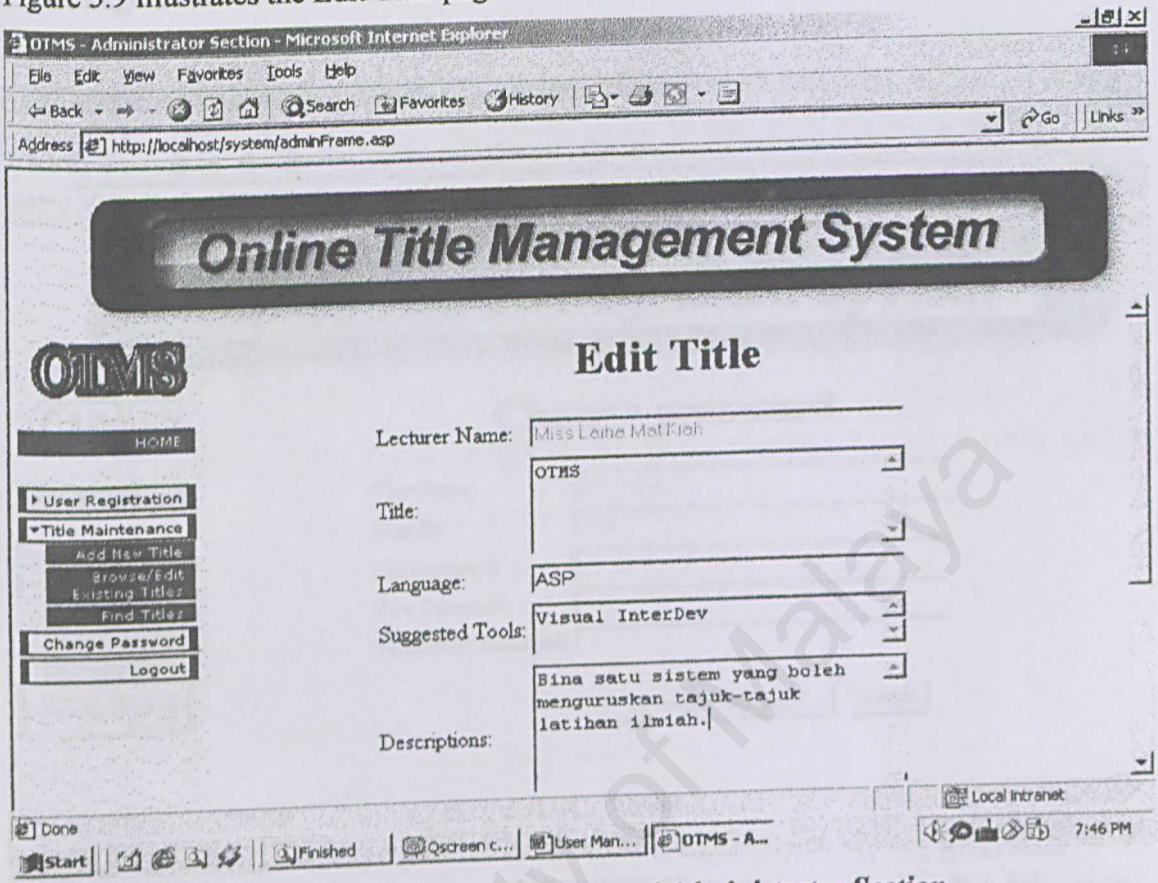


Figure 3.9: Edit Title Page of Administrator Section

- Administrators can delete or update specific title through this page. All information can be updated except for Lecturer Name and Title.
- Click on Delete Title button will deletes the current record and then redirects back to the Browse/Edit Existing Titles page or Find Title page.
- Click on Update Title button will updates the current record and then redirects back to the Browse/Edit Existing Users page or Find Title page.
- Click on Reset button will clears all the information typed in by the administrators and restores it back to the original information.

3.5 Change Password

Figure 3.10 below illustrates the Change Password page.

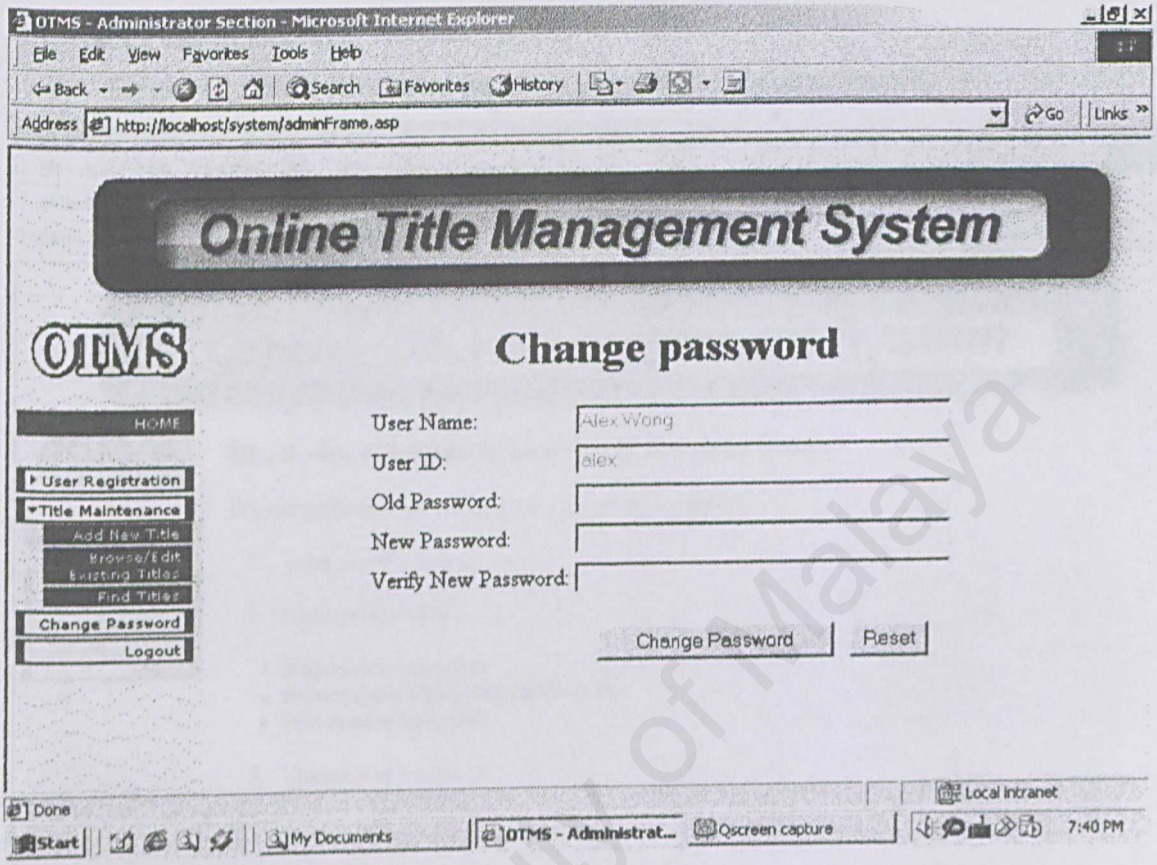


Figure 3.10: Change Password Page of Administrator Section

- After entering the relevant data (Old Password, New Password and Verify New Password), just click the Change Password button to change the existing password to the new password.
- Click Reset button to clear any information typed in by the users.

3.6 Logout

- Click on the Logout link to logout from the system. Users will be redirected back to the home page of **OTMS**.

Chapter 4: Lecturer Section

4.1 Main Page of Lecturer Section

Figure 4.1 below illustrates the main page of the Lecturer Section.

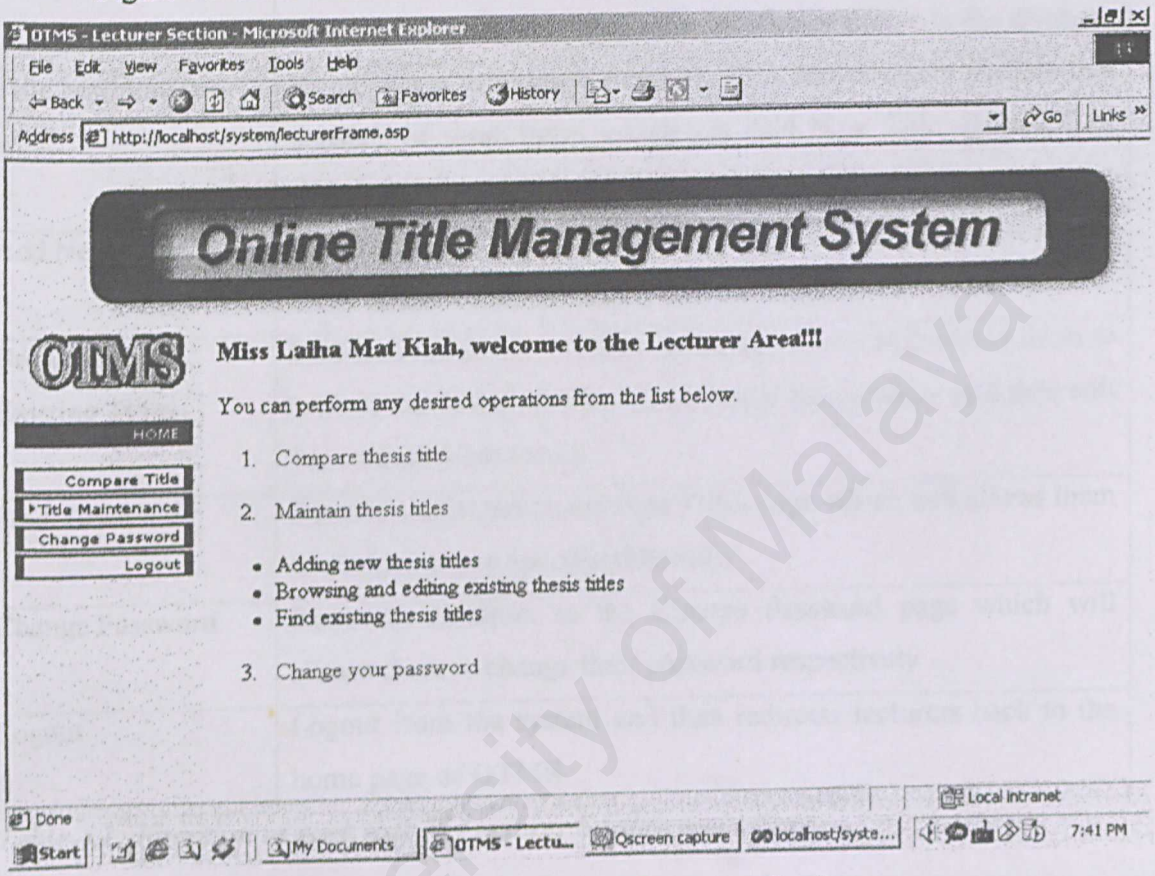


Figure 4.1: Main Page of Lecturer Section

A short description about all the functions and modules in that particular section is also available. All the links to various modules in the Lecturer Section can be found on the left hand side of the page. Table 4.1 describes the function of all the links in detail.

Link	Function
Home	Brings out a new window which consists of the home page of **OTMS**
Compare Title	Redirects lecturers to the Compare Title page which will allows them to compare their title with the titles available in the database
Title Maintenance	Brings out a dropdown menu for Title Maintenance module that consists of three links, which are Add New Title, Browse/Edit Existing Titles and Find Titles
Add New Title	Redirects lecturers to the Add New Title page which will allows them to add a new thesis title into the database
Browse/Edit Existing Titles	Redirects lecturers to Titles' List page which will allows them to browse the list of all titles available in the database and then edit the records if necessary
Find Titles	Redirects lecturers to the Find Titles page which will allows them to search for any specific title/titles
Change Password	Redirects lecturers to the Change Password page which will allows them to change their password respectively
Logout	Logout from the system and then redirects lecturers back to the home page of **OTMS**

Table 4.1: Function of each link in Lecturer Section main page

4.2 Home

- Refer to section 3.1 of Chapter 3: Administrator Section

4.3 Compare Title

Figure 4.2 illustrates the Compare Title page.

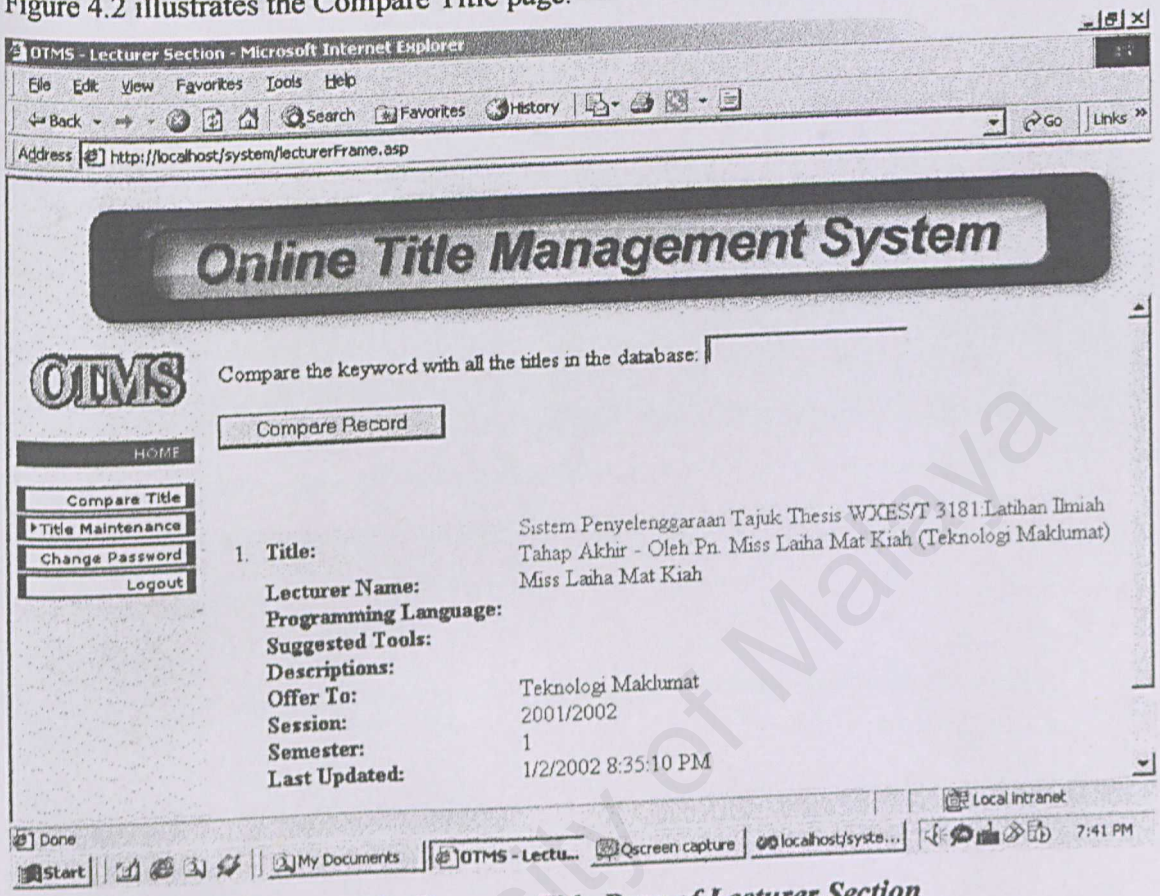


Figure 4.2: Compare Title Page of Lecturer Section

- Only one text box is available for the lecturers to key in their title or any keyword that they wish to compare.
- By clicking the Compare button, the comparison results will be displayed in a well-organized format.

4.4 Title Maintenance Module

- Refer to section 3.3 of Chapter 3: Administrator Section
- A Compare button has been added to the side of the Title text box for lecturers to easily compare their title.

4.5 Change Password

- Refer to section 3.4 of Chapter 3: Administrator Section

4.6 Logout

- Refer to section 3.5 of Chapter 3: Administrator Section

Chapter 5: Student/Visitor Section

5.1 Main Page of Student/Visitor Section

Figure 5.1 below illustrates the main page of the Student/Visitor Section.

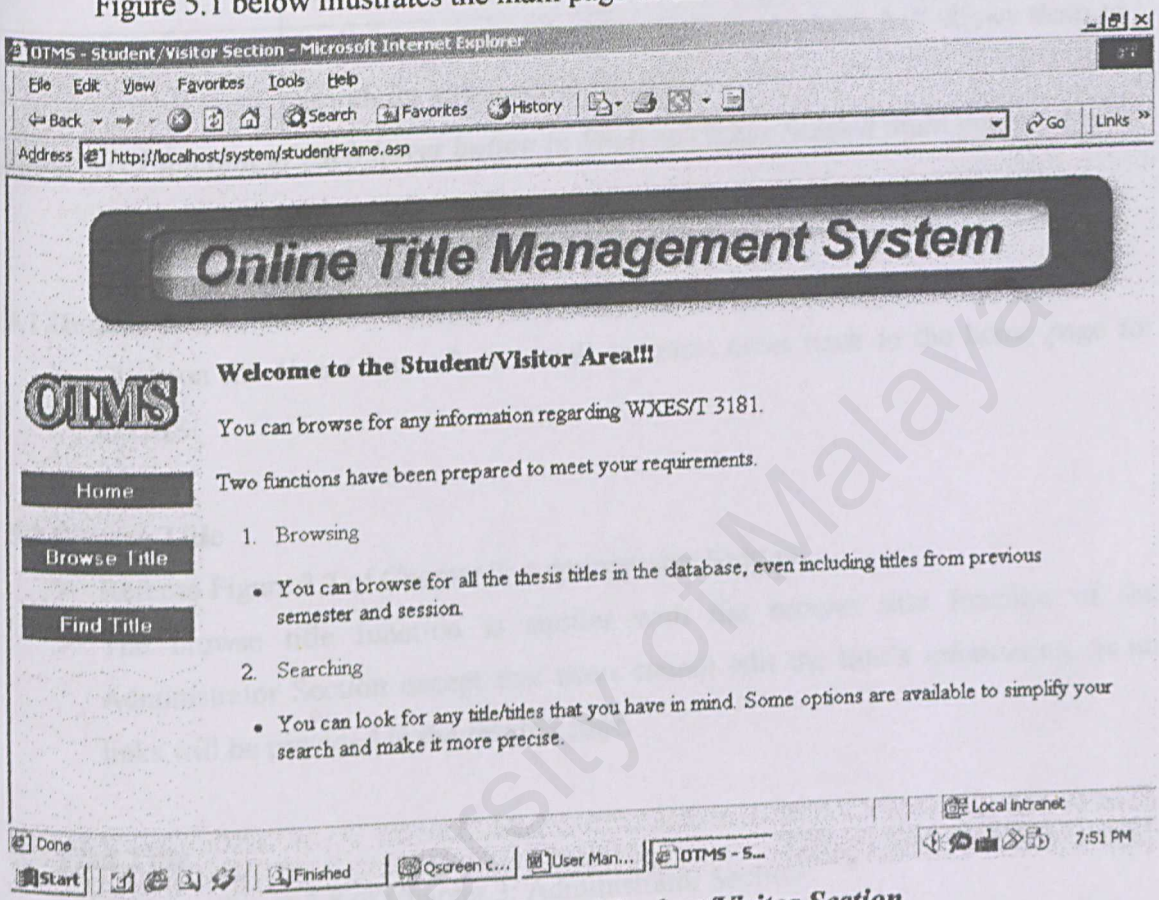


Figure 5.1: Main Page of Student/Visitor Section

A short description about all the functions and modules in that particular section is also available. All the hover buttons to various modules in the Student/Visitor Section can be found on the left hand side of the page. Table 5.1 describes the function of the hover buttons in detail.

Hover Button	Function
Home	Redirects users back to the home page of **OTMS**
Browse Title	Redirects users to the Titles' List page which will allows them to browse the list of all titles available in the database
Find Title	Redirects users to the Find Title page which will allows them to search for any specific title/titles

Table 5.1: Function of each hover button in Student/Visitor Section main page

5.1 Home

- Click on the Home hover button will redirects users back to the home page for **OTMS**.

5.2 Browse Title

- Refer to Figure 3.7 of Chapter 3: Administrator Section
- The browse title function is similar with the browse title function of the Administrator Section except that users cannot edit the title's information, as no links will be provided in the result's page.

5.3 Find Title

- Refer to Figure 3.8 of Chapter 3: Administrator Section
- The find title function is similar with the find title function of the Administrator Section except that users cannot edit the title's information, as no links will be provided in the result's page.