UNIVERSITY OF MALAYA

FACULTY OF COMPUTER SCIENCE & INFORMATION TECHNOLOGY

SESSION 2004/2005

ONLINE DISSERTATION REGISTRATION SYSTEM (ODRS)

TEE SIEW WEI (WEK020264)

SUPERVISOR: ASSOC. PROF. DR. LEE SAI PECK

MODERATOR: MRS. AZWINA MOHD YUSOF

Submitted to the

Faculty of Computer Science & Information Technology
University of Malaya

In Fulfillment Of The Requirement For
The Degree Of
Bachelor Of Computer Science
Session 2004 / 2005

ABSTRACT

Online Dissertation Registration System (ODRS) is purposely developed to the Postgraduate Coordinator, lecturers, administrators and students to handle the registration process and related events efficiently and systematically. This information system converts the traditional manual registration process into a computerized web-based system in managing all the related events. This helps to create a paperless environment by integrating the tasks of registration process carried out manually in the previous time.

Basically, this project will involve four parties, which are the Postgraduate Coordinator, lecturers, administrator and students. Those parties will be involved in the eight modules of this project. The eight modules are Authentication Module, Profile Module, Documents Downloading Module, Bulletin Board Module, Discussion Record Module, Event Calendar Module, Project Title Proposal Module and Project Title Assigning Module. These modules are expected to perform the fundamental functions of the registration system as well as the additional functions like bulletin publishing, documents downloading etc to the users by implementing the dynamic concept. In addition, this ODRS is expected to provide a user-friendly, high usable and easy-to-navigate interface to the users too.

An object-oriented approach is used to develop and document the entire project. The Unified Process is chosen as the software process for the development of this project. The ODRS is built in the Windows platform and ASP.NET (C#), which is written with Microsoft Visual Studio .NET, is used as the programming language for the development of the system. Web-based information system becomes significant and useful in today's trend. It is believed that the Online Dissertation Registration System (ODRS) will be an essential system for the educational institutes to manage the registration process and project titles easily, effectively and efficiently.

AKNOWLEGEMENT

First and foremost, I would like to express my highest gratitude and gratefulness to my project supervisor, Assoc. Prof. Dr. Lee Sai Peck for her patience, guidance, advices, comments and coordination towards my completion in the final year project of the Bachelor Computer Science (Software Engineering).

Secondly, I would like to grab this opportunity to thank my moderator, Puan Azwina. Thanks for her precious time to attend Viva presentations and constructive advice, suggestion and ideas to further enhance value of this project and system.

Thirdly, special thanks to the course mates for their suggestions, encouragement and their helping hands whenever I had doubts. In addition, I would like to thanks staffs of FSKTM. They had provided me some brief information pertaining to the manual registration process of master students' projects. They had tried to provide me the answers of my enquiry through an informal interview.

Last but not least, I would like to express my heartiest appreciation to my beloved parents and siblings for all their love, patience, moral supports and encouragement along this period. I sincerely appreciate the things that they had done for me. Their kindness will always get a special place in the bottom of my heart

Table of Content

Page

Abstract.. ii
Acknowledgement.. iii
Table of Content.. iv
List of Figures.. x
List of Tables... xii

Chapter 1 Introduction

1.1 Project Overview.. 1
1.2 Project Motivation.. 2
1.2.1 Manual Dissertation Registration Process.......................... 2
1.2.2 Statements of problems.. 2
1.3 Project Objectives.. 3
1.4 Project Scope... 3
1.4.1 Authentication Module... 4
1.4.2 Profile Module.. 4
1.4.3 Documents Downloading Module...................................... 4
1.4.4 Bulletin Board Module... 4
1.4.5 Discussion Record Module.. 4
1.4.6 Event Calendar Module... 5
1.4.7 Project Title Proposal Module..................................... 5
1.4.8 Project Title Assigning Module.................................... 5
1.5 Expected Outcome.. 5
1.6 Project Schedule.. 6
1.7 Organization of Report.. 7
1.8 Chapter Summary... 9

Chapter 2 Literature Review

2.1 Domain studies.. 10
2.2 Definition.. 10
2.2.1 Case Study Analysis... 11
2.2.1.1 Traditional Registration Process in FSTKM, UM.................. 11
2.2.1.2 Current FSKTM Undergraduate Thesis Registration System (e-Ilmiah)... 13
2.2.1.3 USC Web Registration... 16
2.2.1.4 University Demo Registration System from regresources.com.. 17

2.2.1.5 The Vérité Registration System (VRS) 18

2.2.2 Proposed System 19

2.3 Technologies Review 20

2.3.1 System Architecture 20

2.3.1.1 Mainframe architecture 20

2.3.1.2 Basic client-server Architecture 21

2.3.1.3 Two tier client-server Architecture 21

2.3.1.4 Three-tier client-server Architecture 22

2.3.2 Application platforms 22

2.3.2.1 UNIX 22

2.3.2.2 LINUX 23

2.3.2.3 Windows XP Professional 24

2.3.3 Web Servers 26

2.3.3.1 Internet Information Server (IIS) 26

2.3.3.2 Apache Server 26

2.3.3.3 Netscape Enterprise Server 27

2.3.3.4 Internet Information Server (IIS) versus Apache Server.. 28

2.3.4 Web Browsers 29

2.3.4.1 Microsoft Internet Explorer 29

2.3.4.2 Microsoft Internet Explorer versus Netscape Navigator.. 30

2.3.5 Programming Languages 31

2.3.5.1 ASP 31

2.3.5.2 JSP 32

2.3.5.3 ASP versus JSP 33

2.3.5.4 ASP.NET 33

2.3.5.5 JavaScript 34

2.3.5.6 PHP 35

2.3.6 Web Application Development Tools 36

2.3.6.1 Microsoft FrontPage XP 36

2.3.6.2 Macromedia Dreamweaver MX 36

2.3.6.3 Microsoft Visual Studio .NET 37

2.3.6.4 Adobe Photoshop 38

2.3.7 Database Management System 38

2.3.7.1 Microsoft SQL Server 2000 38

2.3.7.2 Oracle 9i 39

2.3.7.3 MySQL 40

2.3.7.4 Comparison of SQL Server 2000 with MySQL v4.1...... 41

2.3.8 Data Access Technology 42

2.3.8.1 Microsoft Open Database Connectivity (ODBC) 42

2.3.8.2 ActiveX Data Objects (ADO) 43

2.3.8.3 ADO.NET 43

2.3.8.4 ADO.NET versus ADO 45

2.4 Chapter Summary 46

Chapter 3 Methodology

3.1 Methodology 47
3.1.1 Development Model 47
3.1.1.1 Waterfall Model 47
3.1.1.2 Prototyping Model 47
3.1.1.3 The Unified Process 48
3.1.1.4 Rapid Application Development (RAD) 48
3.1.1.5 V-shaped Model 48
3.1.1.6 Spiral Model 49
3.1.2 Conclusion on chosen methodology 49
3.1.3 Justification of methodology 53
3.2 Information gathering methods 55
3.2.1 Internet research 55
3.2.2 References 55
3.2.3 Observation and informal interviews 55
3.2.4 Review sample thesis documents 56
3.2.5 Discussion with supervisor 56
3.3 Conclusion on tools and technology 56
3.3.1 Chosen System Architecture 56
3.3.2 Chosen Application Platform 57
3.3.3 Chosen Web Servers 57
3.3.4 Chosen Web Browsers 58
3.3.5 Chosen Programming Languages 58
3.3.6 Chosen Authoring Tools 58
3.3.7 Chosen Database Management System 59
3.3.8 Chosen Data Access Technology 59
3.4 Chapter Summary 59

Chapter 4 System Analysis

4.1 System Requirement Analysis 61
4.1.1 Functional Requirements 61
4.1.1.1 Authentication Module 62
4.1.1.2 Profile Module 63
4.1.1.3 Documents Downloading Module 64
4.1.1.4 Bulletin Board Module 65
4.1.1.5 Discussion Record Module 66
4.1.1.6 Event Calendar Module 67
4.1.1.7 Project Title Proposal Module 68
4.1.1.8 Project Title Assigning Module 69
4.1.2 Non-functional Requirements 70
4.1.2.1 GUI 70
4.1.2.2 Flexibility 70

4.1.2.3 Functionality........ 70
4.1.2.4 Portability........ 70
4.1.2.5 Usability........ 70
4.1.2.6 Reliability........ 71
4.1.2.7 Modularity........ 71
4.1.2.8 Manageability........ 71
4.1.2.9 Expandability........ 71
4.1.2.10 Security........ 71
4.2 Run Time Requirements........ 72
4.2.1 Hardware Requirements........ 72
4.2.2 Software Requirements........ 73
4.3 Chapter Summary........ 74

Chapter 5 System Design

5.1 System Architecture........ 75
5.2 Data Design........ 77
5.2.1 Class diagram........ 78
5.2.2 Data dictionary........ 80
5.3 Process Design........ 84
5.3.1 Authentication Module........ 85
5.3.2 Profile Module........ 87
5.3.3 Documents Downloading Module........ 89
5.3.4 Bulletin Board Module........ 91
5.3.5 Discussion Record Module........ 93
5.3.6 Event Calendar Module........ 97
5.3.7 Project Title Proposal Module........ 99
5.3.8 Project Title Assigning Module........ 101
5.4 User Interface design........ 103
5.5 Chapter Summary........ 106

Chapter 6 System Implementation

6.1 Development Environment........ 105
6.1.1 Hardware........ 105
6.1.2 Software........ 106
6.2 Platform Development........ 106
6.2.1 Database Management System........ 106
6.2.2 Web Server........ 107
6.3 Coding and Project Development........ 107
6.3.1 Review Program Documentation........ 108
6.3.2 Prototype Development........ 108
6.3.3 System Coding........ 109

6.3.4 Coding Approach........ 109
6.4 Coding Styles........ 110
6.4.1 Include File........ 111
6.4.2 Standard Naming Convention........ 111
6.4.3 Spacing, Functional Segments and Comments........ 112
6.4.4 User Control........ 113
6.4.5 Configuration Setting........ 113
6.4.6 Cascading Style Sheet (CSS)........ 114
6.4.7 JavaScript........ 115
6.4.8 C# Functional Codes........ 116
6.4.9 Stored Procedures........ 117
6.5 The Modules Implementation........ 117
6.5.1 Authentication Module........ 118
6.5.2 Profile Module........ 118
6.5.3 Documents Downloading Module........ 118
6.5.4 Bulletin Board Module........ 119
6.5.5 Discussion Record Module........ 119
6.5.6 Event Calendar Module........ 120
6.5.7 Project Title Proposal Module........ 120
6.5.8 Project Title Assigning Module........ 120
6.6 Chapter Summary........ 121

Chapter 7 System Testing

7.1 Introduction........ 122
7.2 Stages of Testing........ 122
7.2.1 Unit Testing........ 122
7.2.1.1 Review the code........ 123
7.2.1.2 Trace and test control objects........ 124
7.2.1.3 Examine and test different data type........ 124
7.2.1.4 Choose and implement test cases........ 124
7.2.2 Module Testing........ 126
7.2.3 Integration Testing........ 127
7.2.4 System Testing........ 129
7.3 Chapter Summary........ 129

Chapter 8 System Evaluation & Conclusion

8.1 Introduction........ 130
8.2 Problems and Solutions........ 130
8.2.1 Limited Information from Interviewee........ 131
8.2.2 Inexperienced In the Chosen Software Tools........ 131
8.2.3 Inexperienced In the Chosen Programming Languages........ 131

8.2.4 Lack of Knowledge and Inexperienced in Setting up IIS and Database Server 132
8.2.5 Lack of Resources 132
8.3 System Strengths 133
8.3.1 Consistent and Easy-to-use Interface with Good Layout of the Website 133
8.3.2 Organized and Systematic With Clear Division of the Sections 133
8.3.3 Wide Accessibility 134
8.3.4 Support High Volume of Data and Database Backup 134
8.3.5 Confidentially and Integrity of Information with Security Protection 134
8.3.6 Data Validation and Error Checking 134
8.3.7 Information Listing 135
8.4 System Limitations 135
8.4.1 No Encryption and Decryption 135
8.4.2 Browser Constraints 136
8.4.3 Lack of Utilities 136
8.5 Future Enhancements 136
8.5.1 Encryption and Decryption 136
8.5.2 Browser Consideration 137
8.5.3 Forum or Chat Room Utilities 137
8.5.4 E-mail Server and Appointment Making 137
8.5.5 Students' Registration and Report Submission 138
8.5.6 Report Generating 138
8.6 Knowledge and Experience Gained 138
8.6.1 Exposure to the Real System Design and Development Environment 138
8.6.2 The Importance of the Requirements Elicitation and System Analysis 139
8.6.3 The Learning of the Development Tools and Programming Language 139
8.7 Conclusion 139

References 141
Bibliography 143
Appendix 146

LIST OF TABLES

Page

Chapter 2 Literature review

Table 2.1 Operating System Platform Statistic.............................. 25
Table 2.2 Difference between IIS and Apache.............................. 28
Table 2.3 Microsoft Internet Explorer versus Netscape.................... 30
Table 2.4 Browser Statistic.. 30
Table 2.5 Comparison of SQL Server 2000 with MySQL v4.1............ 41
Table 2.6 ADO.NET versus ADO... 45

Chapter 3 Methodology

Table 3.1 The chosen tools and technologies................................ 60

Chapter 4 System Analysis

Table 4.1 Hardware Requirements for Developers............................ 72
Table 4.2 Hardware Requirements for Users 73
Table 4.3 Software Requirements for Developers 73
Table 4.4 Software Requirements for Users 74

Chapter 5 System Design

Table 5.1 The description of Three-tier Client Server Architecture 76

Chapter 6 System Implementation

Table 6.1 Software tools utilized in the development of ODRS........... 106
Table 6.2 Example codes for the Inherits attribute of the @ Page directive in .aspx 110
Table 6.3 Examples of the standard naming conventions used in ODRS... 111
Table 6.4 Examples server-side scripting (ASP.NET) used in ODRS....... 112
Table 6.5 User control codes 113

Chapter 7 System Testing

Table 7.1 Unit Test case on Adding New Title.............................. 125
Table 7.2 Unit Test case on Approving Proposed Title..................... 125
Table 7.3 Module Test case on Authentication.............................. 126
Table 7.4 Module Test case on User Management........................... 127

LIST OF FIGURES

Page

Chapter 1 Introduction

Figure 1.1 Gantt chart for Project schedule ODRS.............................. 7

Chapter 2 Literature Review

Figure 2.1 A simple page for students to get the thesis titles.................. 11
Figure 2.2 An example static page for students in e-Ilmiah system........... 13
Figure 2.3 The flows of the registration process in FSKTM.................. 15
Figure 2.4 An example print screen of the USC On-Line Registration Form.. 16
Figure 2.5 An example print screen of the University Demo Registration System.. 17

Chapter 3 Methodology

Figure 3.1 The workflows and the phases of the Unified Process............. 52

Chapter 4 System Analysis

Figure 4.1 Use Case diagram for Authentication Module........................ 62
Figure 4.2 Use Case diagram for Profile Module.................................. 63
Figure 4.3 Use Case diagram for Documents Downloading Module.......... 64
Figure 4.4 Use Case diagram for Bulletin Board Module........................ 65
Figure 4.5 Use Case diagram for Discussion Record Module.................. 66
Figure 4.6 Use Case diagram for Event Calendar Module....................... 67
Figure 4.7 Use Case diagram for Project Title Proposal Module.............. 68
Figure 4.8 Use Case diagram for Project Title Assigning Module............. 69

Chapter 5 System Design

Figure 5.1 The component layers found in distributed applications and services built with .NET.. 76
Figure 5.2 Class Diagram for Online Dissertation Registration System 79
Figure 5.3 Sequence Diagram for Login 85
Figure 5.4 Sequence Diagram for Edit account 85
Figure 5.5 Sequence Diagram for Delete account 86
Figure 5.6 Sequence Diagram for Add profile 86
Figure 5.7 Sequence Diagram for Edit profile 87
Figure 5.8 Sequence diagram for Delete profile................................. 87
Figure 5.9 Sequence diagram for View profile.................................. 88
Figure 5.10 Sequence Diagram for Upload document.......................... 88
Figure 5.11 Sequence Diagram for Download document....................... 89

Figure 5.12 Sequence Diagram for Edit upload 89
Figure 5.13 Sequence Diagram for Delete upload 90
Figure 5.14 Sequence Diagram for Add bulletin 90
Figure 5.15 Sequence Diagram for Edit bulletin 91
Figure 5.16 Sequence Diagram for Delete bulletin 92
Figure 5.17 Sequence Diagram for View bulletin 92
Figure 5.18 Sequence Diagram for Register student details 93
Figure 5.19 Sequence Diagram for Add record 93
Figure 5.20 Sequence Diagram for Edit record 94
Figure 5.21 Sequence Diagram for Delete record 94
Figure 5.22 Sequence Diagram for Print record 95
Figure 5.23 Sequence Diagram for view report 95
Figure 5.24 Sequence Diagram for print report 96
Figure 5.25 Sequence Diagram for Add Event 96
Figure 5.26 Sequence Diagram for Edit Event.................................. 97
Figure 5.27 Sequence Diagram for Delete Event 97
Figure 5.28 Sequence Diagram for Add title and Approve title 98
Figure 5.29 Sequence Diagram for View status 98
Figure 5.30 Sequence Diagram for Edit title 99
Figure 5.31 Sequence Diagram for Delete title 99
Figure 5.32 Sequence Diagram for Assign title 100
Figure 5.33 Sequence Diagram for View status use case 100
Figure 5.34 Sequence Diagram for Delete title assigning.................................. 101
Figure 5.35 Screen Design of Login Page 102
Figure 5.36 Screen Design of Student Main Home Page 103
Figure 5.37 Screen Design of Administrator Management Area Home Page. 103

Chapter 7 System Testing

Figure 7.1 Bottom-up Approach in Integration Testing.................................. 128

Chapter 1 Introduction

1.1 Project Overview

In today's fast-moving world, every task is expected to be done on-time, efficiently and effectively. Today we live in an information economy. Information itself has value. Thus, there is a lot of information management systems created to assist every one of us in managing our daily tasks. Furthermore, various online information systems are available throughout the World Wide Web (WWW) too and become more important and preferred as they can be accessed anytime and anywhere once the connection to the Internet is established and users are authenticated.

Traditionally, the registration process is done manually. The registration process can only be completed after many steps of procedures and it may need us to travel to different places. It needs a lot of paper work too in completing the whole process of registration. Thus, a feasible and easy-to-navigate online registration system should be built to solve this problem and make the registration process more efficiently and effectively. The online system will avoid the resource-waste phenomenon and simultaneously enhance the process of registration.

Having realized the importance of the online information system in facilitating the process of registration, Online Dissertation Registration System (ODRS) is proposed to bring conveniences to the individuals involved in the registration of dissertation and also to improve the management of the related events. The existence of the Internet has created the ability to reinvent the way that we fundamentally do registration and thus makes us all more interconnected, closer in time and space, with less manual work, with our processes more timely, and our operations more and more streamlined.

Online Dissertation Registration System (ODRS) is an online information system that enables lecturers, students, postgraduate coordinators, and the system administrator to log in using the services and the functions provided in order to complete the registration process and to manage the related events. This Online Dissertation Registration System (ODRS) is a necessary tool for every institute nowadays as the manual registration is no longer the efficient way to manage the registration process. With the online system, the registration dissertation process can be easier, systematic and efficient.

1.2 Project Motivation

Many motivations to develop Online Dissertation Registration System (ODRS) are due to the weaknesses of current traditional manual registration processes that use paper forms to complete the whole process of dissertation registration.

1.2.1 Manual Dissertation Registration Process

Students encountered inconveniencies when they have to register their preferred master project titles under certain supervisors. Lecturers will publish their titles as the notices. Students need to fill in the forms and seek for the approval from the supervisor in order to obtain project titles. Besides, lecturers use the manual paper forms filling method to record all the details and comments about students who are under their supervision. Lecturers do not have a suitable and proper environment to publish the suggested titles and also any announcements throughout the period of supervising their students.

1.2.2 Statements of problems

Below listed are the problems that occur in the manual dissertation registration process and also the management of the master project titles:

- Time consuming and inefficient – Without an online management information system, the postgraduate coordinator, lecturers and students consume much time in the manual registration procedures and the registration process may not be carried out efficiently as the paper-based documents are used.

- Interaction failure – Students do not know the acceptance status of their preferred titles from respective lecturers unless both parties involved are meeting face-to-face.
- Ineffective in information delivery – Students do not know the status of availability of certain project titles. Furthermore, students will not aware of the updated announcements or notices from lecturers or coordinators as students may miss out the published notices.
- Inefficient processing and control failure – Students may register to different titles under different supervisors at the same time. The administrator can not keep track of the duplication of the registration. In addition, lecturers are not aware of the students who register for more than one title.

1.3 Project Objectives

The Online Dissertation Registration System (ODRS) has the listed objectives below:-

- To develop an organized and systematic online system that is effective and efficient in handling the registration process.
- To create a consistent and easy-to-use interface with a good layout of the website.
- To convert the current manual registration process into a computerized web-based information system using the latest technology.
- To create a paperless environment by integrating the tasks of registration process carried out manually.

1.4 Project Scope

Online Dissertation Registration System (ODRS) will involve 4 parties; they are the Postgraduate Coordinator, administrator, lecturers and students. They will be involved into the eight modules of this project, namely Authentication Module, Profile Module, Documents Downloading Module, Bulletin Board Module, Discussion Record Module, Event Calendar Module, Project Title Proposal Module and Project Title Assigning Module.

Details of these modules are shown below:

1.4.1 Authentication Module

- This module is needed for all the users of the system, who are the Postgraduate Coordinator, administrator, lecturers and students to log into the system.
- This module also enables the administrator to manage the users of the system.

1.4.2 Profile Module

- This module enables lecturers and students to view their personal profile that are kept by administrator after they have logged into the system.
- This module also enables the administrator to add, edit, delete and view lecturers and students' profile after they have logged into the system.

1.4.3 Documents Downloading Module

- This module enables lecturers to upload the forms, articles, reports or any related documents to let students to view and download the desired documents.

1.4.4 Bulletin Board Module

- This module is needed for the Postgraduate Coordinator and administrator to publish updated news and announcements so as to inform the related individuals like students or lecturers pertaining to the updated information.

1.4.5 Discussion Record Module

- This module is needed for students to register themselves with their supervisors before they start any discussion with supervisors.
- This module enables lecturers to record the details of the face-to-face discussion when students have discussion with their respective supervisors.

- This module also enables the lecturers to view and print the report of the summary of the discussions that had been hold between lecturers and students.

1.4.6 Event Calendar Module

- This module enables Postgraduate Coordinator to publish the events according to the academic calendar of the faculty.
- This module enables the lecturers and students to view the published events.

1.4.7 Project Title Proposal Module

- This module enables lecturers to suggest the master project titles and obtain the approval from the Postgraduate Coordinator.
- This module also enables the Postgraduate Coordinator to view all the proposed project title and thus approve the title from lecturers.

1.4.8 Project Title Assigning Module

- This module enables students to view all the listed approved project titles and thus they can approach their preferred lecturers (supervisors) to obtain their desired titles by having an informal discussion with their supervisors.
- It also enables lecturers assign titles to the students after the consensus has been met between lecturers and students.

1.5 Expected Outcome

The aim of the Online Dissertation Registration System (ODRS) is to streamline the registration process through internet and thus facilitate the individuals involved in completing the registration process and managing the related events. It is expected to achieve the following outcomes:

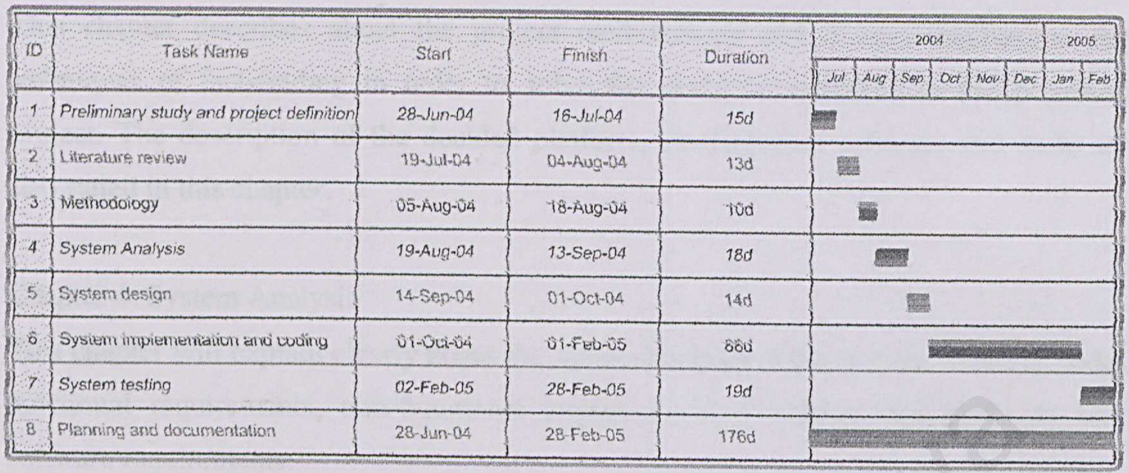
- The system can be accessed anywhere and anytime via the connection of the Internet as it is an online system.
- The system shall provide a user-friendly, high usable and easy-to-navigate interface to the users.
- The system can perform the fundamental functions of the registration system such as adding titles, deleting titles, listing titles and others.
- The system enables the Postgraduate Coordinator, administrator and lecturers to manage the master project titles and some other related events.
- The system can assist the lecturers to record comments and details during the face-to-face discussion with their respective students who are under their supervision.
- The system implements the dynamic concept in managing the additional modules besides registration, such as the bulletin board, downloads and calendar modules.
- The system is able to facilitate the access authentication and it ensures that only authorized users can access the system.
- The final implementation of the system should allow for future enhancement as well as additional modules to extend the system functionality.

1.6 Project Schedule

For any system development project, effective project management is necessary to ensure that the project meets the deadlines, is developed within an acceptable budget, and fulfills customer expectations and specifications [Schach, 2005]. This shows the importance of the project management to plan, organize and control the development of the system within the specified time frame. Thus, a schedule is essential in order to develop the system in a proper manner where the development phase follows a certain time frame allocated. In this project, the forward scheduling is applied where it establishes a project start date and schedules forward from the date.

A Gantt chart shown in Figure 1.1 indicates the project schedule for the Online Dissertation Registration System.

Figure 1.1 Gantt chart for Project schedule ODRS



ID	Task Name	Start	Finish	Duration
1	Preliminary study and project definition	28-Jun-04	16-Jul-04	15d
2	Literature review	19-Jul-04	04-Aug-04	13d
3	Methodology	05-Aug-04	18-Aug-04	10d
4	System Analysis	19-Aug-04	13-Sep-04	18d
5	System design	14-Sep-04	01-Oct-04	14d
6	System implementation and coding	01-Oct-04	01-Feb-05	88d
7	System testing	02-Feb-05	28-Feb-05	19d
8	Planning and documentation	28-Jun-04	28-Feb-05	176d

1.7 Organization of Report

The full detailed report consists of eight chapters. The purpose of the report is to describe all the major phases occurred during the development of the Online Dissertation Registration System (ODRS). Below is shown the brief description about the contents of each chapter in the report.

Chapter 1: Introduction

This chapter gives an overview about the project and the intended information system to be developed. It explains about the objectives of the project, the definition of the problem statements, project scope, project schedule and expected outcomes of the system.

Chapter 2: Literature Review

Basically, this chapter gives explanation about the review and study of the similar existing systems as well as suitable technologies that can be utilized to develop the system. Thus, it is the combination of existing system research and technologies research. The discussion topics of technology review include system architecture, application platform, web servers, web browsers, programming languages, web application development tools, database management system and data access technology.

Chapter 3: Methodology

This chapter describes about the project methodology and it also emphasizes the techniques of fact-finding in order to solve the problems occurred in the proposed project. The description of the decided platform, development software and tools are also stated in this chapter.

Chapter 4: System Analysis

This chapter will explain clearly about the system analysis of the project which includes functional requirements, non-functional requirements as well as the hardware and software requirements.

Chapter 5: System Design

This chapter explains the conceptual and technical design of the system. It covers the system architecture, data design, process design, database design and user interface design. It describes also the combination of all modules into a full functional system.

Chapter 6: System Implementation

This is the most significant sections in the report which illustrates how the suggested programming technology and development tools together with web technology were used in the development of the system. It contains detailed explanation of the implementation phase and the coding process involves transforming the design into a programming language for a complete system.

Chapter 7: System Testing

This chapter describes about the testing phase. This is also an important stage because testing is essential to assure the completion and quality of the system. The objective of testing is to determine system errors or faults for debugging.

Chapter 8: System Evaluation

This chapter describes about the problems encountered and solutions, system strengths, system constraints and limitations, future enhancements and also knowledge as well as experience gained during the development of this project. Finally, an overall conclusion is presented at the end of this chapter.

1.8 Summary

This chapter mainly describes about the introduction of the project. It includes a brief description of project overview and definition of the problems' statements. Apart from that, the relevant topics are also discussed, which are Project Objectives, Project Motivation, Project Scope, Expected Outcome, Project Schedule and Organization of Report.

The next chapter will give the description about the Literature Review of the proposed project. Existing and similar registration systems will be reviewed and also research on the proposed system will be done. Related issues will be discussed. Besides, the tools and technologies will be reviewed based on which a comparison will be done too.

Chapter 2 Literature Review

2.1 Domain studies

This literature review mainly consists of two sections which are analysis case study and technologies review. The analysis case study section contains all the outcome of the research has been done on the existing current system. The features, capabilities and also weaknesses are studied and identified so as to produce a high-useable project. The weaknesses are studied and are tried to overcome in the proposed project. The technologies review section indicates all the suitable tools and programming languages for developing the project. The features, pros and cons are studied so as to identify the most suitable tools for the project.

2.2 Definition

Today's computer technology allows another processing which is called online, real-time or online transaction processing (OLTP) [Stair et al, 2001]. The Online Transaction Processing refers to computerized processing in which each transaction is processed immediately, without the delay of accumulating transaction into a batch.

Online is a computing of equipment or a process directly controlled by or connected to a center processor.

Online system or real-time system is a system where the components (or hosts) are connected; communication between hosts is up all the time. Thus, this enables real-times data transaction and process to occur.

According to Oxford Dictionary, ***Dissertation*** refers to detailed discourse on a subject, especially one submitted in partial fulfillment of the requirements of a master, degree or diploma [Fowler et al, 1998].

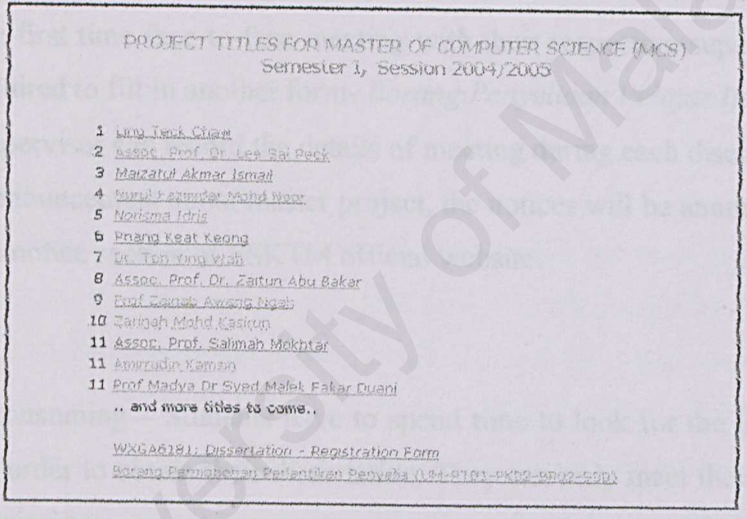
Registration involves the process of record the data automatically (if in electronic devices) into a location of data storing for a specific purpose and with quick access time.

2.2.1 Case Study Analysis

2.2.1.1 Traditional Registration Process in FSTKM, UM

Traditionally, the registration process for the master project (dissertation) in the Faculty of Computer Science and Information Technology, University Malaya is done manually by the students involved.

Figure 2.1 A simple page for students to get the thesis titles.



PROJECT TITLES FOR MASTER OF COMPUTER SCIENCE (MCS)
Semester 1, Session 2004/2005

1 Ling Teck Chaw
2 Assoc. Prof. Dr. Lee Sai Peck
3 Maizatul Akmar Ismail
4 Nurul Fazmidar Mohd Noor
5 Norisma Idris
6 Phang Keat Keong
7 Dr. Teh Ying Wah
8 Assoc. Prof. Dr. Zaitun Abu Bakar
9 Prof Zainab Awang Ngah
10 Zarinah Mohd Kasirun
11 Assoc. Prof. Salimah Mokhtar
11 Amirrudin Kamsin
11 Prof Madya Dr Syed Malek Fakar Duani
.. and more titles to come..

WXGA6181: Dissertation - Registration Form
Borang Permohonan Perlantikan Penyelia (UM-PT01-PK02-BR02-S00)

Procedures (Step-by-step) involved in the manual registration

- Lecturers will suggest the suitable master project titles to the Deputy Dean and thus wait for the approval from the Deputy Dean.
- After the Deputy Dean has approved the titles, the titles will be published on the web pages by the administrator.
- The titles will be published on the static html page in the student notice section of FSKTM official website.
- Students can start the registration by selecting a project title from the list of project titles given.

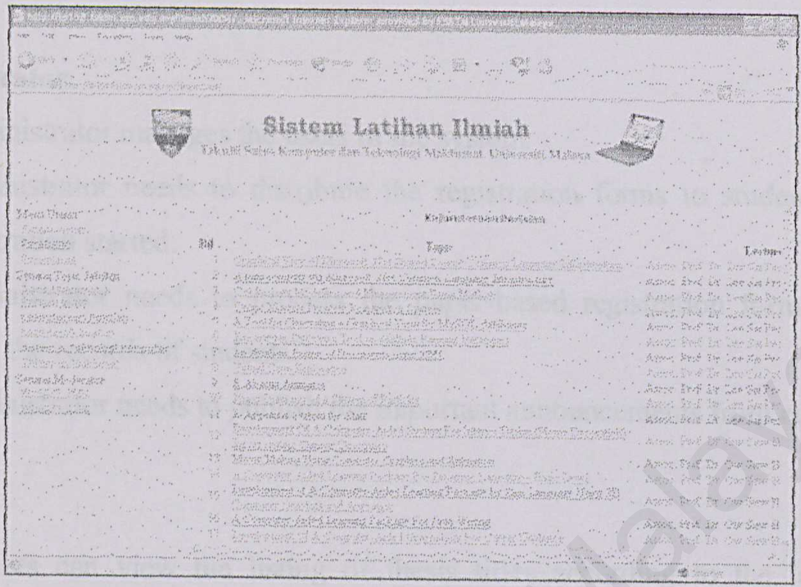
- Students are requested to access the link - http://www.fsktm.um.edu.my/mitpeghm/WXGA0181.cml_0-005.htm so as to view the lists of supervisors.
- The approved titles and descriptions are listed according to the lecturers in .doc or .PDF files. Students need to download it and open it with the Microsoft Word or Acrobat Reader so as to view the titles listed.
- Simultaneously, students have to get a copy of the *Registration Form* and *Application form of Project Supervisor* from the general office.
- Students need to fill in the form completely together with the supervisor signature after getting the approval from their respective supervisors.
- The registration form will have to submit to the administrator in the office.
- For the first time face-to-face meeting with their respective supervisor, students are required to fill in another form- *Borang Penyeliaan Pelajar Ijazah Tinggi* and thus supervisor can record the details of meeting during each discussion.
- Any announcement about master project, the notices will be announced under the student notice section of FSKTM official website.

Drawbacks

- Time consuming – Students have to spend time to look for the lecturer face-to-face in order to obtain their dissertation. They can only meet their lecturer if the lecturer is free.
- Inconvenient – Manual registration is an inconvenient way to manage all the registration processes.
- No standard presentations of information – All the information of registration is not presented systematically in a respective website for students and lecturers. Students may miss out any important notices or announcements. Furthermore, list of titles can not be viewed directly unless the documents are downloaded.
- Paper-based forms and discussion record – All the documents are in the paper-based and students need to get the forms from office or their lecturer in order to complete any processes.

2.2.1.2 Current FSKTM Undergraduate Thesis Registration System (e-Ilmiah)

Figure 2.2 An example static page for students in e-Ilmiah system



Sistem Latihan Ilmiah

Fakulti Sains Komputer dan Teknologi Maklumat, Universiti Malaya

Modules involved

This thesis registration system consists two sections which are static pages for students and dynamic pages for administrator and lecturers. The administration section consists of a few basic administration functions, which include Calendar creating and listing (management of important dates), New Thesis Title Creating, Student Assigning, Student's Mark Assigning and Student Listing. The published website only contains Thesis Title Listing according to the department, Calendar listing (listing of important dates) and publication of announcements. This system involves the parties below:

a. Lecturer

- Lecturer can enter the new project title details, which include project title, description, tools, program, semester, session and number of students. The created details of the titles will be listed in the website.
- Coordinator can enter the important dates for students in the Calendar section.
- Lecturer can assign students after discussion with the students who wish to register the lecturer's title.
- Lecture can assign mark to the students.

- Lecturer need to sign the registration form for their students before the titles are assigned to the students.

b. Administrator

- Administrator manages the users of the system.
- Administrator needs to distribute the registration forms to students when the registration started.
- Administrator needs to process the paper-based registration form in order to make the records of students.
- Administrator needs to publish the important announcements from coordinator.

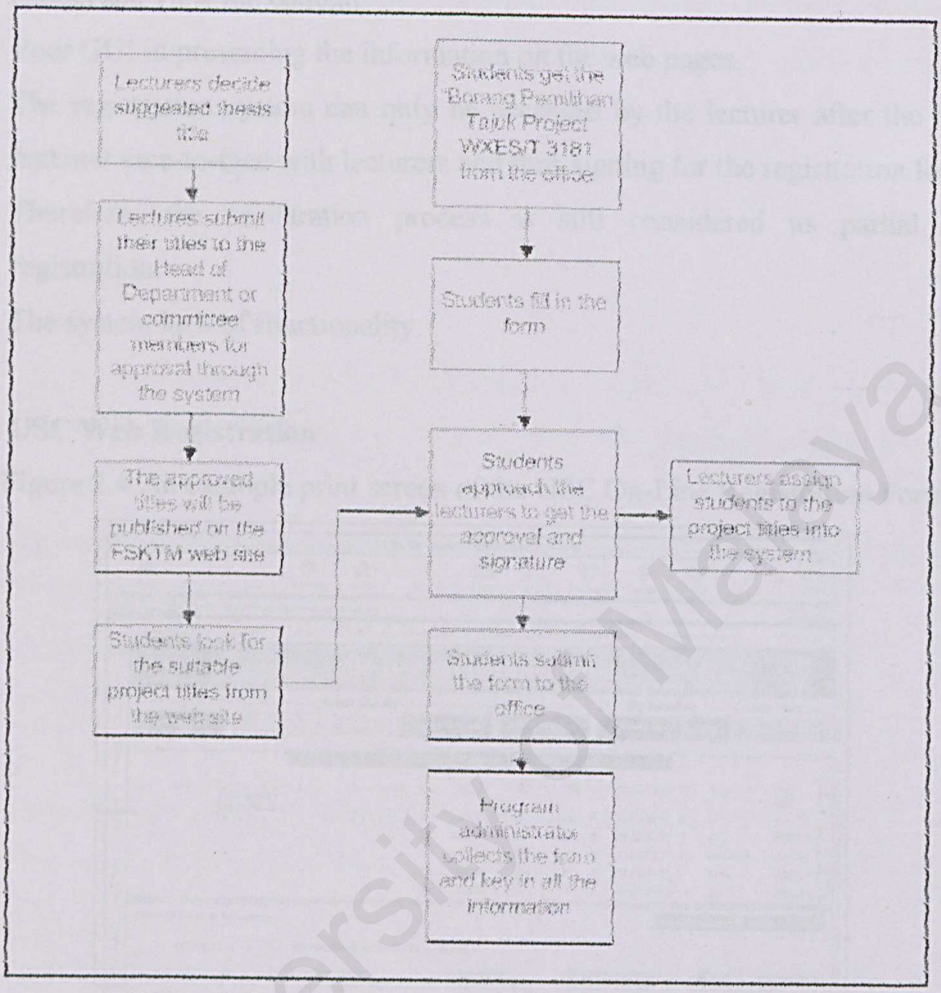
c. Student

- Students can view the listing of thesis titles according to the lecturers and departments.
- Students can view the updated announcements from the website.
- Students can refer the calendar (important dates) from the website.

The flows of the registration process

The procedures involved in the registration process can be referred Figure 2.3.

Figure 2.3 The flows of the registration process in FSKTM


Lecturers decide suggested thesis title
Lectures submit their titles to the Head of Department or committee members for approval through the system
The approved titles will be published on the FSKTM web site
Students look for the suitable project titles from the web site
Students get the 'Borang Pemilihan Tajuk Project WXES/T 3181' from the office
Students fill in the form
Students approach the lecturers to get the approval and signature
Lecturers assign students to the project titles into the system
Students submit the form to the office
Program administrator collects the form and key in all the information

Strengths

- Information about the thesis titles are listed clearly according to the departments and lecturers.
- In addition, the tools and number of students that can be assigned for certain project is stated.
- Calendar which stated important dates to remind students is provided.

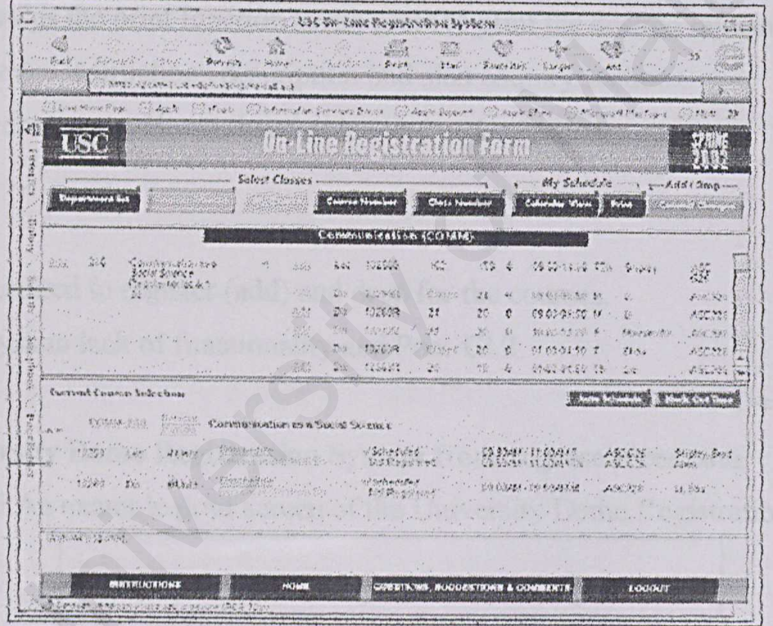
Weaknesses

- The website is designed in the static pages, which has no interaction with users.

- The website provides no authentication and authorization. Anyone also can access and view the content.
- Poor GUI in presenting the information on the web pages.
- The registration system can only be accessed by the lecturer after the students had met face-to-face with lecturers and thus signing for the registration forms.
- Therefore, the registration process is still considered as partial manual registration.
- The system lack of functionality

2.2.1.3 USC Web Registration

Figure 2.4 An example print screen of the USC On-Line Registration Form



Source: https://camel.usc.edu/

This is a system which enables the users to register (add) and drop the courses online. In addition, the information of the schedule for each course is provided to the users to browser through. It has 3 main sections in the system, which includes Select Classes, My Schedules and Add/Drop. The Select Classes has the sub sections of Department list, Course offered, My Degree, Course Number and Class Number. The My Schedule sub section includes Calendar View and print, where as the Add/Drop has only sub section of Commit Changes. The main objective of this On-Line Registration System is to let users view and thus select the suitable courses according to their arrangement of the schedule. Besides that, the instructions descriptions are provided and it is acted as the help to the users. The questions, suggestions and comments section is provided to gain the feedback from users and also give a way to users in order to enquire about any doubts or comments.

Strengths

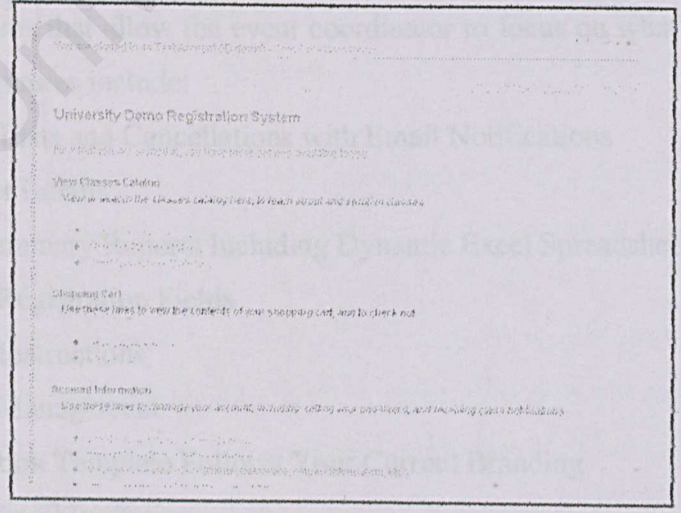
- The system enables users to build their own schedule. There are two ways of building a schedule which are using the auto schedule and building the schedule manually.
- It provides the print function to the users to print the desired information.
- It provides instructions description and also enquiry sections.
- The website provides authentication and authorization.

Weakness

- Only limited to register (add) and drop for the courses.
- The system lack of functionality and Poor GUI.

2.2.1.4 University Demo Registration System from regresources.com

Figure 2.5 An example print screen of the University Demo Registration System



University Demo Registration System

View Classes Catalog

View or search the classes catalog here, to learn about and register classes

Shopping Cart

Use these links to view the contents of your shopping cart, and to check out

Account Information

Use these links to manage your account, including setting your password, and managing class notifications

source: http://www.regresources.com

This demo shows the university online registration system. It is formed of 2 main parts which are official web site for registration and the other one is administrative area. Administrative area provides the various functionalities to let the administrators to manage the online registration process and other related events involved. The functions included manage classes and students, edit greetings and emails, manage reports, manage payment involved and others advanced functions. In the main area of registration area, this system has shop cart registration, event categories, a two-tier event catalog, registration limits, waitlists, prerequisites, event blockers and tiered pricing.

Strengths

- The website provides authentication and authorization.
- It has a full set of complete registration functions involved for users and also administrator to dynamically manage the related events and objects.
- It has the ability to operate as a complete large-scale university-style registration system.

2.2.1.5 The Vérité Registration System (VRS)

VRS is an online turn-key registration system from Vérité, Inc. It's also the easiest way to make the most of the company's big event. VRS can be accessed from anywhere, and easily adapts to enrollment of any kind: tradeshows, product registrations, catalog and collateral requests, contests and lotteries, surveys, entry forms, whatever. VRS contains automated features that allow the event coordinator to focus on what's really important. Some of these features include:

- Waiting Lists and Cancellations with Email Notifications
- Reminder Emails
- Event Summary Reports Including Dynamic Excel Spreadsheets
- Custom Registration Fields
- Driving Instructions
- Identity Management
- Registration Template Follows Your Current Branding
- HTML Email Invitation

The VRS contains two distinct parts; the end-user registration experience and the administration and reporting experience.

2.2.2 Proposed System

Online Dissertation Registration System (ODRS) is a web-based system which is used to assist students and lecturers to complete their registration process in anywhere and anytime. It is also a system created to ease the maintenance of the dissertation titles. This system will provide some basic functionality to lecturers and postgraduate coordinators such as discussion recording, documents uploading and downloading, bulletin board and calendar. These functionalities will increase the interaction between students and lecturers and thus reduce the use of papers and also save students' and lecturers' time in managing all the processes and procedures involved within the both parties.

Benefits of proposed system

- Improve the GUI and information presentation – Good GUI and more dynamic on presentation the information to the users as all the submenus are listed clearly in the Main Menu. Users can just choose from Main menu in order to browse through desired page.
- Time-saving and resources-saving – The manual registration process will be replaced by the online registration system, this will save the students and lecturers time in registering the titles. Furthermore, the use of paper is reduced and thus this will save resources as all the procedures are managed by the computers.
- Efficient in data processing and management – All the data involved in the process of registration will be handled by the system and the data will be saved in the database.
- Facilitate individuals involved – The online registration system facilitates lecturers and students to complete the registration anytime and anywhere. It also facilitates administrator, Postgraduate Coordinator and lecturers in managing the

dissertation tiles, recording the details of discussion, managing announcements and so on.

2.3 Technologies Review

2.3.1 System Architecture

Client-server architectures exist to provide organizations flexible and robust infrastructures that, depending on how they are designed, can cater to specific business needs. The client-server architecture is a versatile, message-based and modular infrastructure that is intended to improve usability, flexibility, interoperability, and scalability as compared to centralized, mainframe, time sharing computing [Sadoski, 2004]. In this client-server environment, a client is any computer requesting services from base or server which plays role as the provider of services. Multiple computer platforms are dedicated to special functions such as database management, printing, communications, and program execution in the client-server architecture.

2.3.1.1 Mainframe architecture

It is also called host-client architecture, which the application and database reside on one center host computer, and the user interacts with the application and data using a "dumb" terminal. Users interact with the host through a terminal that captures keystrokes and sends that information to the host.

Pros:

- It is not tied to a hardware platform, easy client installation and reliable.

Cons:

- It does not easily support graphical user interfaces (GUI).
- It is difficult to maintenance and it is expensive too.
- It does not easily access to multiple databases from geographically dispersed sites.

2.3.1.2 Basic client-server Architecture

It allows a client to send a request to a server through a query, whereby reducing network traffic. The server then queues and processes the request by interpreting; analyzing and serving the information back to the client. The client application then has the ability to present this information to the user via a graphical user interface (GUI) or a host terminal. This process allows a multi-user environment using a shared data source. Remote Procedure Calls (RPCs) or Standard Query Language (SQL) statements are typically used to communicate between the client and server.

Pros:

- It is ideally for small businesses and shared data environment.
- It assists to improve performance.

Cons:

- Software deployment, software control, poor performance.

2.3.1.3 Two tier client-server Architecture

It consists of three components, which are user system interface, processing management and database management distributed in two layers. The user system interface is usually located in the user's desktop environment and the database management services are usually in a server that is a more powerful machine that serves many clients. It splits the processing management between client and server.

Pro:

- It is intended to improve usability by supporting a form-based, user-friendly interface.
- It improves flexibility by allowing data to be shared, usually within a homogeneous environment.
- It is used extensively in non-time critical information processing where operations of the system are not complex.
- It works well in relatively homogeneous environments with less-frequent change processing rules.

Cons:

- It provides limited flexibility in moving or repartitioning the program functionality from one server to another without manually regenerating procedural code.
- It is difficult to administer and maintain because every upgrade must be delivered, installed, and tested on each client if applications reside on the client side.

2.3.1.4 Three-tier client-server Architecture

The three tier architecture is also known as the multi-tier architecture. It emerged to overcome the limitations of the two-tier architecture. The three tier architecture is used when an effective distributed client/server design is needed that provides increased performance, flexibility, maintainability, reusability, and scalability, while hiding the complexity of distributed processing from the user[Sadoski, 2004]. Thus, with these characteristics, three-tier become a popular choice for web-application and web-based information system.

Pros:

- It improves performance, maintainability and flexibility.
- Processing is centralized in the middle tier.
- It facilitates software development because each tier can be built and implemented in a separate platform.

Cons:

- The separation of the three-tier is not always obvious.
- The development environment is more difficult to use.
- It involves complicated infrastructure to build three-tier architecture.

2.3.2 Application platforms

2.3.2.1 UNIX

UNIX is a powerful operating system originally developed by AT&T for minicomputers. It can be used on many computer system types and platforms, from personal computers to mainframe systems. It runs on all sizes of computers using a wide range of microprocessors. [Flynn et al, 2000] It has functions that manage the hardware and the executing of applications separately. In addition, it includes a set of libraries, file system and process control. UNIX is designed to provide a multi-user, multitasking system for the use by programmers. It benefits companies using both small and large computers systems.

Pros:

- High portability because it is compatible with different types of hardware
- High flexibility in choosing the hardware manufacturer.
- Easier to move programs and data among computers and connect mainframes and personal computers to share resources.
- It has very powerful of utilities as its utilities are brief, single-operation commands that can be combined in a single command line to achieve desired result
- It is device independent because it includes the device drivers as part of the Operating System and UNIX can be configured to run any devices.

Cons:

- UNIX's commands are so brief that novice users find it unfriendly.
- It needs very powerful workstations and therefore not cost effective.
- UNIX is a registered trademark and the use of terms requires developers to meet very expensive certification criteria.

2.3.2.2 LINUX

Linux was originally created by Linus Torvalds who wanted developed a system that could maximize the capabilities of the Intel 80386 microprocessor at the University of Helsinki [Flynn et al, 2000]. It was based on a powerful multiplatform operating system, UNIX. It brought the speed, efficiency and flexibility of UNIX to a PC environment and thus taking advantage of all capabilities built into personal computers.[Horvath, 1998] Linux is an open-source program and the source code is freely available to anyone for improvement. It includes true multitasking, virtual memory, shared libraries, demand loading, memory managements, Internet networking, and other features.

Linux is used for a wide variety of purposes including networking, software development, and as an end-user platform. It is often considered as low cost alternative to other more expensive operating system. Furthermore, it can act as a server for most major file serving protocols, and provide all the major internet applications. Linux is becoming widely adopted in commercial and industrial markets around the world nowadays.

Pros:

- Its source code is freely available to everyone as it is developed under the GNU General Public License.
- It is a stable and high-performance operating system for Internet usage with a host of performance enhancements.
- It is easier in installation and has a greater selection of drivers.
- GUI-based front ends for web administration and Window management.

Cons:

- Linux is inherently unsafe because every malicious cracker in the universe has the source code to the web site that develops under Linux.
- Installation will cause problem as it requires some enlightened input from the installer. In addition, disk partitioning and mounting of file systems need relatively advanced concepts.
- It is developed by people worldwide, therefore lack of the proper organized support.
- Talented people in any particular arbitrary combination of Linux / Apache / Jrun / Mod_Perl / PHP / Locomotive and so on are difficult to find than talented people in Microsoft Windows / IIS / COM.

2.3.2.3 Windows XP Professional

Windows XP Professional integrates the strengths of Windows 2000 Professional. It combines the standard-based security, manageability, and reliability and also the best business features of Windows version to create the best desktop operating system for business. This latest version of operating system increases the computing power while lowering the cost of ownership for desktop computers.

Windows XP has the great features and services which can bring benefits to its users, such as new standard for efficient and dependable computing, advanced management, deployment, and support tools to make tasks easier as well as advanced productivity tools

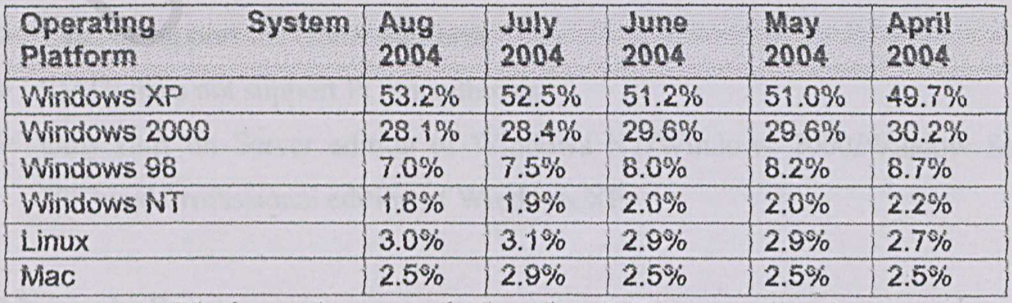
Pros:

- ➢ Administrator can boot the system to a command console and potentially repair the errors of operating system
- ➢ Help to minimize user downtime and increase system stability as it provides windows installer service.
- ➢ Files and folders can be easily published to any Web service that uses the WebDAV protocol
- ➢ Administrator have a greater degree of flexibility in deploying Internet Explorer 6.0.
- ➢ Easily integrate Windows XP Professional into existing peer-to-peer networks.

Cons:

- ➢ The ongoing maintenance and support requirement of Windows XP Professional can make them much more costly to run.

Table 2.1 Operating System Platform Statistic



Operating System Platform	Aug 2004	July 2004	June 2004	May 2004	April 2004
Windows XP	53.2%	52.5%	51.2%	51.0%	49.7%
Windows 2000	28.1%	28.4%	29.6%	29.6%	30.2%
Windows 98	7.0%	7.5%	8.0%	8.2%	8.7%
Windows NT	1.8%	1.9%	2.0%	2.0%	2.2%
Linux	3.0%	3.1%	2.9%	2.9%	2.7%
Mac	2.5%	2.9%	2.5%	2.5%	2.5%

Platforms that count for less than 0.5% are not listed

Source: http://www.w3schools.com/browsers/browsers_stats.asp

2.3.3 Web Servers

2.3.3.1 Internet Information Server (IIS)

Internet Information Services (IIS) is one of the best and powerful Web server that provides a highly reliable, manageable, and scalable Web application infrastructure for the server versions of Window NT/Windows 2000/Windows XP Professional/Windows Server 2003. It is a group of Internet Servers which includes a Web or alternatively a Hypertext Transfer Protocol server and also a File Transfer Protocol Server. IIS can assist organizations to increase the web site and application availability while lowering system administration costs. It is much easier to set up and maintain than many of its UNIX-based competitors. It has the capability of supporting in intranet as well as internet. IIS provides three major services to users, which are World Wide Web server, File Transfer Protocol server and Gopher server. Its Internet Service Manager (ISM) application controls these services on this or any other IIS server on the network.

Pros:

- Site administration is performed using the Microsoft Management Console.
- Free for downloading
- Index Server(Indexing tool) can handle Microsoft Office documents
- It provides integrated search engine capabilities.
- It reinforces security with automatic security patch management. In addition, it reduces attack surface area by not installing the IIS by default when the Operating System is installed on a new computer.

Cons:

- Lack of support for UNIX platform
- SMTP does not support POP mailboxes
- Only runs on Server edition of Windows NT/Windows 2000/Window Server 2003 and Professional edition of Windows XP.

2.3.3.2 Apache Server

Apache is the high-end enterprise-level server for UNIX and Windows platforms. Apache is a free and open source server. According to Netcraft survey August 2004 web server survey, apache is in the lead with 67.70%, followed by IIS at 21.21% and Netscape at 3.14%. Apache users have come to rely on the server's rock-solid reliability, outstanding performance, and rich set of features. It is undeniable that the apache server is a powerful, flexible, HTTP 1.1 compliant Web Server. Furthermore, its popularity is also due to its free-distributed code. Being able to view the source code and also active user support for the server is one of the best parts of it. It is not only can let the users and developers modify and add features to the server, but the vast numbers of developers have tried to make modules to add on to the server. Apache has a built-in search engine and HTML authoring tools and supports File Transfer Protocol (FTP).

Pros:

- Source codes are freely distributed and active user support for the server.
- Reliable, secure, robust, high performance and contains rich set of features.
- It supports for the HTTP 1.1 protocol.
- Quick technical support via Usenet newsgroup

Cons:

- Setup and maintenance of the server are accomplished via command-line scripting tools.
- It has not offered any visual, wizard or browser-based maintenance capabilities and any GUI configuration or administration tools.
- It needs higher deployment and maintenance costs.

2.3.3.3 Netscape Enterprise Server

Netscape Enterprise Server is high-performance, highly scalable web server software for deploying the largest-scale web sites. It can run on a representative collection of operating system such as AIX, Digital UNIX, HP-UX, Iris, Solaris and Windows. It was voted the Best of 1998 by PC Magazine and is running some of the largest e-commerce, ISP, and portal web sites on the Internet.

Netscape Enterprise Server provides support for the HTTP 1.1 protocol, a built-in search engine with document attributes and custom views, advanced content publishing and management for end users through an approach called "Netshare", server clustering and administrative rights delegation, and Java integration with support for JavaBeans, JDBC, and servlets. It has the capabilities as failover, automatic recovery, dynamic log rotation as well as high performance and scalability for dynamic and secure content. It has a powerful development environment that supports Web-based applications development that can be run on the Intranet, extranet or Internet.

Netshare is an innovative approach which provides content management services. The services include Web publishing, access and version control, agent services, and link management. Netshare also facilitates users who may publish pages to a server without to learn FTP to perform uploading and downloading operations.

Pros:

- It supports several of platforms.
- It contains integrated search engines, SMTP support, centralized server management and support dynamic application development.
- It is reliable as it has process-failover protection to eliminate service interruptions.
- Its browser-based administration interface provides ability to remotely manage all servers from any location, using any browser.

Cons:

- The costs needed is high especially compare to IIS
- Technical Support is needed.

2.3.3.4 Internet Information Server (IIS) versus Apache Server

Table 2.2 Difference between IIS and Apache

Features	Apache	Internet Information Services
Server type	WEB	WEB
Latest version	2.0	6.0
Price details	Freeware	Included with all Windows Server 2003 version
Vendor	Apache Software Foundation	Microsoft Corporation
Description	The predominant open source web server	Web server that works in conjunction with Windows Server

		operating systems.
Administration	GUI configuration, GUI setup, Remote administration, SNMP configurable/monitorable.	GUI configuration, GUI setup, Remote administration, SNMP configurable/monitorable.
Futureproofing / scalability	64-bit port, cluster support, IPv6 support.	Net compliant, 64-bit port, cluster support, IPv6 support.
Other features	Multiple logs, supports Microsoft ISAPI, Virtual servers.	Multiple logs, supports Microsoft ISAPI, Virtual servers, web-based user-interface.
Programming / Scripting	Includes source, own API, support external scripting / batch language.	Own API, own scripting / batch language, support external scripting / batch language.
Security	ActiveDirectory authentication, built-in proxy capabilities, Internal user access scheme, LDAP authentication, SSL (hardware)	ActiveDirectory authentication, antivirus features, built-in firewall capabilities, built-in proxy capabilities, LDAP authentication, SSL (hardware and software)
Support	Commercial support available, forum support, mailing list support.	Commercial support available, forum support, GSA scheduled, Service-level agreement.

Source: http://www.serverwatch.com/stypes/compare/servercompare.htm

2.3.4 Web Browsers

2.3.4.1 Microsoft Internet Explorer

Internet Explorer 6.0 is the standard browser in the Microsoft Windows XP Home Edition and Windows XP Professional. It was released in August 2001. It provides enhanced privacy features as well as a flexible and reliable browsing experience for users. Many new and enhanced features are included in Internet Explorer 6.0. The features are listed below:

- Flexibility - New and innovative features are added such as Image Toolbar, Print Preview, Favorites, auto complete, search companion and so on.
- Web Privacy - IE 6.0 contains the security tools to support the Platform for Privacy Preferences (P3P), a technology under development by the World Wide Web Consortium (W3C).
- Reliability - New fault collection services assist to identify problems that need to be fixed in future updates to Windows Internet technologies.

2.3.4.2 Microsoft Internet Explorer versus Netscape Navigator

In order to encompass the ability to access other Internet services, the features of browser software have been expanded. Table 2.3 shows the features compared of web browser.

Table 2.3 Microsoft Internet Explorer versus Netscape

Features	Internet Explorer	Netscape Navigator
Speed	- Faster at loading large graphics files as it is more efficient at managing large images.	- 35 percent faster than IE especially its cache is emptied.
Security	- Support Secure Sockets Layer (SSL), digital signature, 128 bits keys for their security codes, and Authenticode. - Offer support for parental controls and ratings systems (PICS and RSAC)	- Support Secure Sockets Layer (SSL), digital signature, 128 bits keys for their security codes. - Not offer any support for parental controls and ratings systems.
Platform support	- Only available on specific platform such as Macintosh, Unix and Windows.	- Cross-platform, supports all versions of Windows, Macintosh, and Unix variants such as Solaris and AIX.
Multimedia	- Support various multimedia types such as AVI movies, AU and AIFF sound files and MPEG video. - Contains ActiveX control called ActiveMovie that handles all of its multimedia files.	- Support various multimedia types such as AVI movies, AU and AIFF sound files and MPEG video. - Comes pre-configured with plug-ins, including QuickTime and Live3D VRML.

Table 2.4 Browser Statistic

Browser	Aug 2004	May 2004	Feb 2004	Nov 2003	July 2003
Internet Explorer 6.X	70.5%	72.6%	71.5%	71.2%	66.9%
Internet Explorer 5.X	7.3%	9.2%	11.5%	13.7%	20.3%
Opera 7	2.2%	2.2%	2.2%	1.9%	1.7%
Mozilla	14.6%	14.6%	9.0%	7.2%	5.7%
Netscape Navigator 7	1.4%	1.4%	1.5%	1.6%	1.5%
Netscape Navigator 4	0.3%	0.3%	0.4%	0.5%	0.6%

Source: http://www.w3schools.com/browsers/browsers_stats.asp

2.3.5 Programming Languages

2.3.5.1 ASP

Active Server Pages (ASP) is a server-side scripting environment that can be used to create and run dynamic as well as interactive web server applications. ASP can be combined with HTML pages, script commands and COM components inclusively ACtiveX controls to create interactive web pages or web-based applications. Active Server Pages enables server side scripting for IIS with native support for both Visual Basic Script and Java Script.

With ASP, the code can be simply written in the HTML page. The HTML tags and code are side by side. No compiling and no complex interfacing needed. ASP has made it much quicker and easier to create highly interactive Web sites. It also makes your pages easier to maintain and update in the future.

Furthermore, it always allows user to easily access data and thus displays it on a web page by using an ODBC-compliant database. It also has an important feature which is the ability to use cookies to store and retrieve information. The Request object has a Cookie collection, and user can use this in data processing

Pros:

- Primarily supports two scripting languages, vbScript and JavaScript
- Simple scripting to access the server and its extensions, an efficient way to returm dynamic content.
- It is easy to learn.
- More efficient than CGI because it runs as a service and can take advantage of multithreaded architectures.
- An open, compile-free application environment in which can combine HTML, scripts and reusable Active X server components.
- Browser independent.

Cons:

- Limits only to Microsoft platforms.

- The simplest of scripting mistakes can cause the server to crash or hang, effectively bringing down the website.

2.3.5.2 JSP

Java Server Pages (JSP) is a technology produced by Sun Microsystems. It is entirely based on Java programming language. Internally, JSP pages are dynamically converted into Servlets, which are simply Java classes. This means JSP enjoys all the capabilities that Java programming supports. JSP runs in server-side component which is known as JSP container, which translates them into equivalent Java servlets.

JSP is a great deal more efficient than many other scripting languages, such as CGI and ASP. Tags can be defined in tag libraries and then used within any JSP page. This makes for a better separation of page content from its code, which leads to less scattered code and thus the site is easier to maintain. Global changes need to be made only to the tags defined in these tag libraries, making time-consuming, page-by-page fixes things of the past. Java code inside a JSP page is more script-like because it doesn't require Java class and package definitions.

Pros:

- Primarily support real java code.
- Dynamically assembled into Servlets, it supports the same modularity, reusability, platform-independence, and access to Java APIs that Java programming supports.
- Efficient than other scripting languages as its threading model and error handling prevent server hangs and crashes.
- Can be written in well formed, valid XML by using XSLT.

Cons:

- It needs a skillful knowledge in Java programming.

2.3.5.3 ASP versus JSP

JSP and ASP have some basic concepts in common. [Karan, 2003]

- Both make use of simple sever-side scripting to provide access to Web server information and functionality.
- Both have similar styles of delimiting this scripting from a page's content. [DaySite Web Hosting, 2003]

JSP is Java, so it is case sensitive while VBScript of ASP is not.

2.3.5.4 ASP.NET

ASP.NET is a new programming framework from Microsoft for developing next generation web Applications. It is a framework built on the Common Language Runtime and introduces a new paradigm to server-side Web development. It provides the easiest and most scalable way to build, deploy and run distributed web applications that can target any browser or device. It is a complete architecture for developing Web sites and Internet-distributed objects using managed code.

ASP.NET is broken into 3 layers which are the ASP.NET Application, ASP.NET Pages and ASP.NET Web Services. The ASP.NET application layer represents the entire solution, with ASP.NET Pages representing the user interface, and Web Services can expose selected functionality of the application.

ASP.NET currently offers built-in support for three languages which are C#, Visual Basic, and JScript. A key feature of ASP.NET is the separation of code into a separate file from the HTML page that calls it. The code behind concept helps clarify the roles of designers and developers, and neatly accommodates another important .NET technology, namely XML Web Services.

ASP.NET is different from ASP in two major ways: [Ramesh, 2003]

- ASP.NET offers several programmatic enhancements over ASP, such as, Compiled ASP.NET Scripts, Post-Back in caching, Server Controls in the web forms and deployment.

- ASP.NET offers a change in programming fundamentals while ASP followed procedural pattern of creating pages.

Pros:

- Powerful database-driven functionality in ASP.NET
- ASP.NET is object-oriented and has multiple language support.
- A new language, C#, has been introduced, which combines the efficiency of C++ with some of the ease of development of Visual Basic.
- The main programming languages have been moved far closer together, so code written in VB, C++ and C# can be intermixed. Freely step between the languages in the debugger.
- Compiled code and caching make processing of web requests much more efficient and faster in web applications.
- It includes a large number of pre-written components that can generate commonly used HTML form and user-interface items.
- It contains memory leak and crash protection.

Cons:

- It needs to be hosted on a Microsoft web server.

2.3.5.5 JavaScript

JavaScript is designed by Netscape and based roughly on C. It enables authors to design interactive sites. It was rapidly retargeted to look more like Java when Java came out, however JavaScript is different from Java. Although it shares many of the features and structures of the full Java language, it was developed independently.

JavaScript is unable to stand on its own like Java. It is a text-based language that must be placed within HTML, to be read by the browser and interpreted so the instructions can be performed. Hence, JavaScript is also very easy to edit. JavaScript can interact with HTML source code to enable web authors to spice up their sites with dynamic content. It carries with it a smaller command set and a much simpler structure, though it remains an OOP (Object Oriented Programming Language).

JavaScript is primarily a browser extension language. It is not only interpreted by the user's web browser but also be interpreted by some web servers to add server-based scripting.

2.3.5.6 PHP

PHP Hypertext Preprocessor is an open-source server-side; HTML embedded scripting language used to create dynamic Web pages for e-commerce and other Web applications. When the PHP preprocessor in Web server notices a PHP language tag, the PHP engine is invoked to execute the code.

PHP can use the ODBC to access to the data in the database. PHP offers excellent connectivity to most of the common databases including Oracle, Sybase, MySQL, ODBC and others. PHP also offers integration with various external libraries, which allow the developer to do anything from generating PDF documents to parsing XML

PHP will be familiar to any programmers who have worked with imperative programming languages. PHP is the natural choice for developers on Linux machines running Apache server software. This is because it can be compiled as a module or directly into the Apache binary.

Pros:

- Open source and work native with Apache.
- More efficient memory and usage and faster execution.
- PHP code will run faster as there is no overhead of communicating with different COM objects in different processes.

Cons:

- Lack of exceptions, event-based error-handling instances that interrupt the normal flow of a program.
- Structure of exception handling is not standardized and causes inconsistency.

2.3.6 Web Application Development Tools

2.3.6.1 Microsoft FrontPage XP

Microsoft FrontPage is focused on making basic Web document publishing and site management easy for business professionals and end users who are not full-time Web publishing professionals. It is known as a WYSIWYG (what you see is what you get) editor which allows to forego actual coding and simply design web pages within the document window. FrontPage is usually the editor of choice of novice web builders because of it is easy to learn and use.

FrontPage is just like Internet Assistants which allows users to create multimedia Web sites with just a few mouse clicks. It allows users developing pages to control the way their pages look and work, and allows Web professionals to quickly code in either Normal View or HTML View. It can also enable users and third parties to create rich add-on and custom Web applications.

FrontPage enables non-expert user to create and maintain sophisticated, interactive site from their desktop with a powerful yet simple "web top publishing" system. From the aspect of integration with Microsoft Office, there is add-ins for Office and it is easy to update Office documents.

2.3.6.2 Macromedia Dreamweaver MX

Macromedia Dreamweaver MX 2004 provides the most complete set of tools available for building, editing, and maintaining accessible websites and web applications. It combines its renowned visual layout tools with the rapid web application development features of Dreamweaver UltraDev and the extensive code-editing support of Macromedia HomeSite.

By using Macromedia in handling website, HTML codes are generated concisely and it is always editable. It also includes advanced features that take advantage of the latest innovations on the web, such as dynamic HTML, CSS, PHP, Web service and so forth, while still ensuring that web pages work well in different web browsers. Other features in the Macromedia include easy integration of Active X components, Java applets, plug-ins for improved web page interactivity.

Dreamweaver comes with a complete HTML debugger and it has a Cleanup HTML command for files created in other programs like FrontPage. Furthermore, it provides three types of view to ease users which are full design view, full code view and design with code view. It has a complete set of unique built-in Web objects including rollover button interface, tabular-data table builder and so on. This is essential to provide an easy way to user to create web pages. It also can integrate the effect with other components of Macromedia like Flash, Movies, Shockwave and Fireworks in order to develop interactive web pages.

2.3.6.3 Microsoft Visual Studio .NET

Microsoft Visual Studio .NET is Microsoft's integrated development environment (IDE) for creating, documenting, running and debugging programs written in a variety of .NET programming languages. It is a visionary and practical development tool for creating the next generation of console application, windows application web applications and even web services. It is also a XML-based programming model and tool; it is fully supported by MSDN and Windows DNA 2000 servers.

Microsoft Visual studio .NET offers editing tools for manipulating several types of files. It is a powerful yet sophisticated tool for creating business-critical and mission-critical applications. Furthermore, it also provides users with window for exploring files and customizing controls. The window can be accessed using the toolbar icons in the menu bar and also toolbar. Besides, it has provided an extensive help mechanism for users too.

Microsoft Visual studio .NET desires to achieve the goal of providing a single, unified environment that adapts to whichever environment they are working in and is largely device-independent. Thus, it offers users:

- *Natural Interface* - A collection of technologies that enable the interactions between humans and computer such as natural language to input via a new type in box. Technologies can be combined for multimodel user interface.
- *Universal Canvas* - The universal canvas builds upon XML schema to transform the Internet from a read-only environment into a read/write platform, enabling users to interactively create, browse, edit, annotate and analyze information.
- *Information Agent* - Manages history, preferences, context over the internet and provides greater control on the interactions with the web sites and services.
- *SmartTags* - Extensible architecture allows anyone to create adaptive user experience and data handlers.

2.3.6.4 Adobe Photoshop

Adobe Photoshop has become the most popular image-managing software that is available for Macintosh and Windows-based computers. Photoshop delivers a comprehensive environment for professional designers and graphics producers to create sophisticated images for print, the Web, wireless devices, and other media. Moreover, Photoshop provides a consistent work environment with other Adobe applications.

All the functions provided in Photoshop are included under a set of user-friendly editing tools. It contains graphical icons to represent every functions of each button. Besides that, it also provides many shortcut keys that is easier and save time for users and for those who do not like to use mouse.

2.3.7 Database Management System

2.3.7.1 Microsoft SQL Server 2000

Microsoft SQL Server 2000 is a full-featured relational database management system (RDBMS) that offers a variety of administrative tools to ease the burdens of database development, maintenance and administration. It is a very powerful tool for turning information into opportunity. It is based on the client/server architecture, which divides processing into two components: a front-end, or client component, that run on a local workstation and a back-end, or server component, which runs on a remote computer.

Tools that are frequently used include Enterprise Manager, Query Analyzer, SQL Profiler, Service Manager, Data Transformation Services and Books Online. SQL Server 2000 supports the rapid development of enterprise-class business applications that can give the companies in the market a critical competitive advantage. Microsoft SQL Server 2000 has the features such as full web-enabled, high scalable and reliable, faster time-to-market.

Pros:

- Industry-leading support for XML
- Enhanced tools for system management and tuning
- Lower implementation and maintenance costs
- Exceptional scalability and reliability

Cons

- Only works on Windows-based platforms.

2.3.7.2 Oracle 9i

The Oracle9i Database is the current release of Oracle's information management solution. It is designed to support and leverage the capabilities of the Internet.

It provides extensive functionality to support businesses running on the World Wide Web, as well as traditional mission critical OLTP and Data Warehousing applications. It has been designed to provide the most complete and low cost solution for any business information management requirement.

The Oracle9i Database is capable to handle all types of information for all types of applications and provides outstanding, across-the-board, and transparent scalability from low-end uniprocessor to high-end symmetric multi-processor systems and multi-node clustered configurations. [Cheevers, 2002]

Oracle 9i has the features listed below:

- Consolidate and manage all the Internet content.
- Reduce the time it takes for a business to make better business decisions by analyzing more data faster.
- Integrate business information from disparate sources, including third-party databases and file systems.
- Secure and protect the privacy of sensitive business information. It provides an end-to-end security infrastructure especially in the industry.
- High availability as it guarantee that critical business information is available when needed.
- Provide a powerful online reorganization and redefinition architecture.
- The Oracle database can resize the various memory structures dynamically, whereby no shutdown and restart of the system is necessary.
- Prevention and improved handling of disk corruption are included. Oracle provides a precision database repair so as to store the important data efficiently and effectively.
- Oracle9i Streams exists to meet all the information sharing needs. It enables the propagation of data, transactions and events in a data stream.
- Oracle 9i supports a number of languages for developing applications such as C, C++, Java, COBOL, PL/SQL and Visual Basic.
- Oracle also includes the technology of summary management. It provides a complete solution for the creation; use and on-going management of summary tables through an intelligent query rewrite mechanism.

2.3.7.3 MySQL

The MySQL database server is the most popular open source database tools. It is a true multi-user, multi-threaded SQL database server. Its architecture makes it extremely fast and easy to customize. MySQL has the features shown below:

- It can handle large databases with no memory leaks.
- It supports a broad subset of the ANSI SQL syntax.
- Cross-platform support with standard thread safe client library.

- Unique independent storage engine according to users' need.
- It contains a privilege and password system which is very flexible and secure (allow host-based verification), including SSL support.
- Passwords are secure as all password traffic is encrypted when connecting to a server.
- Query caching which can increase the performance of commonly issued queries.
- A service of full-text indexing and searching is provided.
- Powerful and useful embedded database library.

The unique separation of the core server from the storage engine makes it possible to run with strict transaction control or with ultra-fast transactionless disk access, whichever is most appropriate for the situation. There are four versions of the database server available which are MySQL Standard, MySQL Max, MySQL Pro and MySQL Classic. [MySQL Database Server, 2004]

Pros:

- Faster than Access 2000, DB2, Informix, MS-SQL, Solid, Sybase, and Oracle 8.0.3, on Windows platforms.
- High performance and simple to download and install.
- Fully networked and full access control of users.
- Portable as it is cross-platform support.

2.3.7.4 Comparison of SQL Server 2000 with MySQL v4.1

Table 2.5 Comparison of SQL Server 2000 with MySQL v4.1

Feature	SQL Server 2000	MySQL v4.1
Platform	Windows-based platforms	Support all known platforms, including Windows-based platforms, AIX-based systems, HP-UX systems, Linux Intel, Sun Solaris and so on.
Hardware Requirements	Intel or compatible platforms.[Chigrik, 2003]	Uses less hardware resources. (Minimum 32 Mb RAM and 60 Mb hard disk

		space)
Editions	6 editions: Enterprise, Standard, Personal, Developer, Desktop Engine and SQL Server CE.	2 editions: Standard and Max.
License	Available under two licensing: LProcessor license and Server/per-seat client access license (CAL)	Dual Licensed. Users can choose to use: i. An Open Source/Free Software product under the terms of theGNU General Public License or ii. Purchase a standard commercial license from MySQL AB.(per database server)
Advantage	i. Holds the top TPC-C performance and price/performance results. ii. Easier to install, use and manage. iii. Transact-SQL is a more powerful language than MySQL dialect.	i. Supports all known platforms, not only the Windows-based platforms. ii. Requires less hardware resources. iii. Can be used without any payment under the terms of the GNU General Public License.

2.3.8 Data Access Technology

2.3.8.1 Microsoft Open Database Connectivity (ODBC)

Open Database Connectivity (ODBC) is an open standard Application Programming Interface (API) for accessing a database. A separate module or driver is needed for each database to be accessed. Microsoft supports the main proponent and supplier of ODBC programming. The goal of ODBC is to make it possible to access any data from any application, regardless of which database management system (DBMS) is handling the data.

In order to achieve its goal, a middle layer which is called a database driver is inserted between an application and the DBMS. The purpose of this layer is to translate the application's data queries into commands that the DBMS understands, both the application and the DBMS must be ODBC-compliant. This means the application must be capable of issuing ODBC commands and the DBMS must be capable of responding to them.

ODBC is based on and closely aligned with The Open Group standard Structured Query Language (SQL) Call-Level Interface. It allows programs to use SQL requests that will access databases without having to know the proprietary interfaces to the databases. ODBC handles the SQL request and converts it into a request the individual database system understands.

2.3.8.2 ActiveX Data Objects (ADO)

ActiveX Data Objects is a Microsoft Active-X component which is automatically installed with Microsoft Internet Information Server (IIS). These objects can be used to both client and server information for building dynamic content web pages. ADO is easy to use and is language-independent. It provides extensible interfaces for programmatic access to all types of data.

Unlike the Remote Data Objects (RDO) and Data Access Objects (DAO), ADO is designed to be used to access all sorts of different types of data, including web pages, spreadsheets, and other types of documents. ADO is the single data interface for developers creating 1 to n-tier client/server and Web-based data-driven applications. It provides consistent access to data for creating a front-end database client or middle-tier business object using an application, tool, language, or even an Internet browser.

ADO has provided an efficient, robust interface for COM programmers to work with data. ADO is widely used as an interface to a variety of different stores because it can be called from any automation language including Microsoft Visual Basic 6.0, Microsoft Visual C++, and a variety of scripting interfaces. The primary advantage with ADO was improved support for disconnected data and the ability for a disconnected recordset to marshall over the network.

2.3.8.3 ADO.NET

ADO.NET is introduced by Microsoft in the .NET Framework. It is an evolution of the data access architecture provided by the Microsoft ActiveX Data Objects (ADO) programming model. It can provide better platform interoperability and scalable data access than ADO. ADO.NET has different functionality from ADO, and it introduces a few objects used for data access in various manners and conditions.

ADO.NET is an expansion of ADO with some of the key concepts retained. ADO.NET has greatly extended to provide access to structured data from diverse sources, which are all accommodated in a consistent, standardized programming model

ADO.NET is broken down into 3 namespaces in the .NET framework. Namespaces are used to organize components into groups based on organizational and logical reasons. Before connecting to a database, firstly all the necessary namespaces are imported to allow users to work with the objects required. Creating a new set data access APIs in ADO.NET brings a number of advantages listed below:

- Improved integration with XML
    - ADO.NET was designed from the ground up to integrate with XML, and leverages XML in a fundamental way. It relies on XML for remoting data between tiers or clients. XML tools are used to perform validation, hierarchical queries, and data transformations on relational data by ADO.NET.
- Integration with the .NET Framework
    - In-memory data can be exposed through common .NET Framework structures, including arrays and collections, providing users with the common access methods when working with relational data.
- Improved support for the disconnected business model
    - A new object, the DataSet, which serves as a common, in-memory representation of relational data was introduced in the ADO.NET. It is ideal for packaging, exchanging, caching, persisting, and loading data.
- Explicit control of data access behaviors
    - ADO.NET provides well-defined, factored components with predictable behavior, performance, and semantics that enable users to address common scenarios in a highly optimized manner.

- Improved design-time support
  - It leverages known metadata at design time in order to provide better run-time performance and more consistent run-time behavior.

2.3.8.4 ADO.NET versus ADO

Table 2.6 ADO.NET versus ADO

Feature	**ADO**	**ADO.NET**
Memory-resident data representation	Uses the **RecordSet** object, which looks like a single table.	Uses the **DataSet** object, which can contain one or more tables represented by **DataTable** objects.
Relationships between multiple tables	Requires the JOIN query to assemble data from multiple database tables in a single result table.	Supports the **DataRelation** object to associate rows in one **DataTable** object with rows in another **DataTable** object.
Data visitation	Scans **RecordSet** rows sequentially.	Uses a navigation paradigm for non-sequential access to rows in a table. Follows relationships to navigate from rows in one table to corresponding rows in another table.
Disconnected access	Provided by the **RecordSet** but typically supports connected access, represented by the **Connection** object. You communicate to a database with calls to an OLE DB provider.	Communicates to a database with standardized calls to the **DataAdapter** object, which communicates to an OLE DB provider, or directly to SQL Server.
Cursors	Utilizes server-side and client-side cursors.	The architecture is disconnected so cursors are not applicable.
Programmability	Uses the **Connection** object to transmit commands that address underlying data structure of a data source.	Uses the strongly typed programming characteristic of XML. Data is self-describing because names for code items correspond to the "real world" problem solved by the code. Underlying data constructs such as tables, rows, and columns do not appear, making code easier to read

		and to write.
Sharing disconnected data between tiers or components	Uses COM marshalling to transmit a disconnected record set. This supports only those data types defined by the COM standard. Requires type conversions, which demand system resources.	Transmits a **DataSet** as XML. The XML format places no restrictions on data types and requires no type conversions.
Transmitting data through firewalls	Problematic, because firewalls are typically configured to prevent system-level requests such as COM marshalling.	Supported, because ADO.NET **DataSet** objects use XML, which can pass through firewalls.
Scalability	Database locks and active database connections for long durations contend for limited database resources.	Disconnected access to database data without retaining database locks or active database connections for lengthy periods limits contention for limited database resources.

2.4 Chapter Summary

This chapter indicates findings, summarization, and analysis of what have been found and studied about the similar system of registration system. The materials obtained can be in the form of books, journals, conference papers proceeding, symposium technical reports or articles available from Internet.

The purpose of this literature review is to get a better understanding and also knowledge on the development tools that can be used to develop a project. Various suitable tools and languages are studied in order to choose the suitable tools and languages for the project. In Chapter 3, all the chosen tools in the project will be mentioned.

Chapter 3 Methodology

3.1 Methodology

Methodology is a term that always been used in the context of software engineering. In the 1970s, the word methodology began to be used in the sense of "a way of developing software"; the word actually means the science of methods [Schach, 2005]. A methodology is actually a set of activities, methods, deliverables and automated tools that the developer teams use to develop and enhance the information system. It consists of a set of phases where by each phase comprises a collection of sub phases.

3.1.1 Development Model

3.1.1.1 Waterfall Model

Waterfall model is the earliest method of structured system development but yet it is a popular development model [The Waterfall Model, 2004]. The waterfall model describes a development method that is linear and sequential. It breaks the complex tasks of development into several steps, starting from requirements gathering and ending with release. It is checked for proper execution and quality through validation. It is easy to use and one phase has to be finished before the next phase.

Normally, Waterfall model is used if a system's requirements and problems are understood clearly and well defined. This is because a system is difficult to alter once the model is used. For example, if a client requests to change the requirements suddenly, the system developer has to start from the scratch as there's no fast way to implement the new requirement.

3.1.1.2 Prototyping Model

Prototyping approach is an iterative process involving designers and users to clarify the requirement of the system. When using the Prototyping Model, the developer builds a simplified version of the proposed system or external interfaces and thus presents it to the customer for consideration as part of the development process. The customer will then provide the feedback after reviewing the prototype.

The Prototype model is a good model for the project which has unambiguous user requirements. The model will help users to understand what they actually want. Prototyping includes the steps of requirements definition/collection, design, prototype creation/modification, assessment, prototype refinement and system implementation [Prototyping Model, 2004]. The iterative process finally produces a fully functional system.

3.1.1.3 The Unified Process

The Unified Process is the primarily object-oriented methodology nowadays. The Unified Process is not a specific series of steps that, if followed, result in the construction of a software product. Instead, it is modified for the specific software product to be developed. There are three key words which describe the distinguishing aspects of the Unified Process: use-case driven, architecture-centric, and iterative and incremental. Furthermore, the unified process uses the terms Inception, Elaboration, Construction and Transition as the phases of the lifecycle. These phases contain activities in varying degrees [Schach, 2005].

3.1.1.4 Rapid Application Development (RAD)

Rapid Application Development (RAD) is designed to give faster development and better results and to take maximum advantage of recent advances in development software [Schach, 2005]. It is associated with a wide range of approaches to software development: from hacking away in a GUI builder with little in the way of analysis and design to complete methodologies expanding on an information engineering framework.

3.1.1.5 V-shaped Model

V-shaped model is an improved version of waterfall model. It is different from waterfall model that each test phase matches each development phase: requirements with system testing, high-level design with integration testing, and detailed design with unit testing. It does not run into the problem that the software is impossible to be tested because system test, integration test, and unit test are planed ahead. For example, when we plan the requirement, we also plan for system testing. Therefore, when the system is built, we have a whole set of test cases for system testing. By that way, the system does not meet user requirements.

3.1.1.6 Spiral Model

Spiral Model is an iterative approach. The underlying concept of the model is that for each portion of the product and for each of its levels elaboration, the same sequence of steps or cycle is involved. Risk analysis and the risk-driven approach are key characteristics of the spiral [Kan, 1995].If a project with a very high risk is developed, spiral model should be used. In every iteration, the risks are evaluated and forecasted to determine the development of the project. The spiral model is made up of the steps of project objectives determining, risk assessment, engineering and production, planning and management.

3.1.2 Conclusion on chosen methodology

The Unified Process will be used as the methodology to develop and enhance the system that is going to be built. This model is the closest to the way that software or application is produced in the real world.

The Unified Process provides a disciplined approach to assigning tasks and responsibilities within a development organization. Its goal is to ensure the production of high-quality software that meets the needs of its end-users, within a predictable schedule and budget .As we know, the Unified Process is the primarily object-oriented methodology nowadays, and it has three main aspects involved which are use-case driven, architecture centric as well as iterative and incremental.

In this project, the Unified Modeling Language is used to prepare the use-case diagram to elicit the functional requirements of the system. This is because the Unified Process mainly uses the Unified Modeling Language (UML) when preparing all blueprints of the software system. Initially, the interaction of users with the system being developed is illustrated as use cases. Each use case identified in the project is a piece of functionality that gives the user results or values. All use cases then together make up use case models. Finally, the system's functional requirements in this project are captured. These use case models can drive developer the design, implementation and test of the system. This means the development process will follow a series of workflows and developed in parallel with the system architecture and both influence each other.

Iteration and incrementation are the basic aspect in developing a project. Iteration and incrementation are used in conjunction with one another practically in this project. That is, the project will be divided the work into several modules and these modules are constructed piece by piece (incrementation), each incrementation will go through a series of workflows with iteration. Each iteration consists of the following workflow which are requirement workflow, analysis workflow, design workflow, implementation workflow and test workflow. The developers do not assume that all requirements are known at the beginning of the lifecycle; indeed change is anticipated throughout all phases This Online Dissertation Registration System (ODRS) defined its workflows of the Unified Process as following:

(a) Requirement workflow – A basic understanding of the application domain, which is the specific environment in which the target system is to operate, is acquired. The requirement gathering is also carried out in this stage. The constraints like deadline, reliability and feasibility of the system are identified too.

(b) Analysis workflow – Analyze and refine the requirements that had been gathered to achieve the detailed understanding of the requirements of ODRS as it is essential for developing a feasible and reliable system. The UML diagram (use case model) is used to describe all the requirements of the ODRS in such a way that resulting design and implementation are easy to maintain.

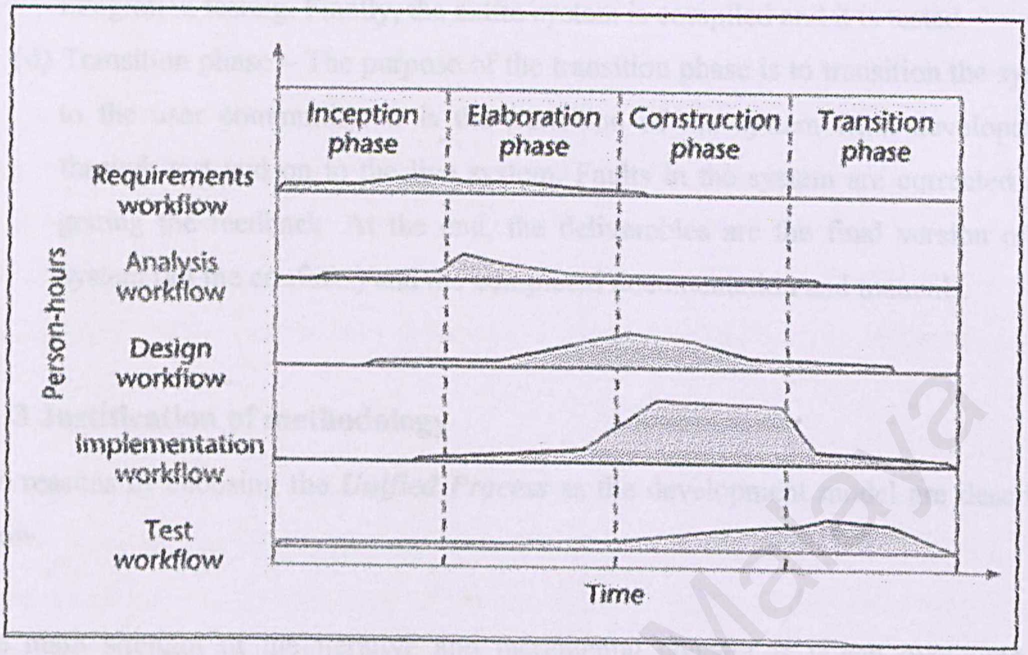
(c) Design workflow - This involves refining the artifacts of the analysis workflows until the results obtained from the analysis workflow is in a form that can be implemented. The classes and objects involved are extracted during the analysis workflow and designed during the design workflow.

(d) Implementation workflow – The designed system is implemented in the chosen implementation language, which is ASP.NET, which the .NET framework 1.1 is needed. The work of the project has been divided into several modules, and all the design and codes are implemented and thus are integrated into the system which can work correctly.

(e) Test workflow – The testing is carried out in parallel with the other workflows, starting from the beginning. Unit testing and integration testing are involved in this stage. The testing is a major activity during each iteration and particularly at the end of each iteration. This is to ensure that the errors are identified early and can be corrected. It is the verification of the entire system.

Figure 3.1 shows the workflows and the phases of the Unified Process. The Unified Process repeats over a series of cycles. Each cycle consists of four phases: inception, elaboration, construction, and transition. Planning and documentation are performed throughout the development of the project as both are essential parts of every phase.

Figure 3.1 The workflows and the phases of the Unified Process



(a) Inception phase – It covers the early work of domain analysis, gathering requirements, analyzing them and beginning the design. Simplified use-case model and tentative architecture are presented. Furthermore, the risks which consist of technical risks, not getting the requirements and not getting the architecture right are identified and ranked so that the critical risks are mitigated first. Then, testing is commenced to ensure the requirements of ODRS are accurately determined.

(b) Elaboration phase – It is the improvement of requirements through analysis, and into design. The major activity involved is refinements of the previous phase. In addition, it involved specifying the features and designing the architecture too. Architectural decisions have to be made with an understanding of the whole system: its scope, major functionality and nonfunctional requirements such as performance requirements. It is the most critical of four phases.

(c) Construction phase – This phase covers finalising the design, writing codes, and drawing related pictures. It produces the first operational-quality version of the system being developed. The emphasis in this phase is on the implementation and testing of the system. All the components are coded and units are tested.

Then, it will be compiled and integrated to form the sub-systems, which are integration testing. Finally, the entire system is compiled and it is tested.

(d) Transition phase – The purpose of the transition phase is to transition the system to the user community. It is the transition of the system from development, through test and on to the live system. Faults in the system are corrected after getting the feedback. At the end, the deliverables are the final version of the system (all the artefacts) and the completed documentation and manuals.

3.1.3 Justification of methodology

The reasons of choosing the ***Unified Process*** as the development model are described below.

The main strength of the iterative and incremental concept is it can overcome the moving-target problem (requirements always change). Therefore, its iterative process can help developer to identify the requirements clearly and thus to build a feasible system. Furthermore, it always can capture and present the best practices in the system as the Unified Process is an ongoing improvement model.

Besides, the Unified Process is the best solution found to date for treating large problems as a set of smaller, largely independent sub problems as it provides a framework for iteration and incrementation to cope with the complexity of the entire system. It helps developer to correct the inevitable mistakes made while the system is being developed. It also allows resolving problem and issues discovered early.

The iterative approach is risk-driven. Normally, the Unified Process addresses risk in first two phases, inception and elaboration. Early rather than late identification and mitigation of risks is important to ensure the risks may not lead to the failure of the system and thus increase the predictability.

The use-case driven development in the Unified Process assists developer to rectify the problem and capture the requirements of the system. This is because the use case diagram provides a systematic and intuitive means of capturing functional requirements. The requirements of the system are modeled visually via the Unified Modeling Language (UML). It has also driven the whole development process from analyzing, designing, and implementing even to testing. For example, the identified use cases in the project help in identifying classes and verifying the instances of classes. Moreover, it also helps in developing the user interfaces. It could also provide the starting point for user manual too.

The testing activity is an integral part and it will be implemented particularly at the end of each iteration, not only at the end of the completion of the entire system. It will help to identify the errors early during the implementation and hence the errors made can be corrected. This will then reduce the errors which may cause the system failure at the end of the development.

The concept of this model helps the developer to complete the project in a structured and organized way. The development task has to be divided into increments (phases), within each increment, iteration is performed until the task is completed. As humans, we are limited by Miller's Law [Schach, 2005], which states that humans can actively handle seven concepts at one time. Therefore, the tasks can be handled easily by developer as the tasks have broken into increments, not the entire system in a whole.

Furthermore, it emphasizes in planning rather than rapid prototyping as the planning phase is carried out throughout the development of the system. The documentation that is also being carried out throughout the project will always help the developer to keep track on the progress and information of the system. Finally, the project can be completed with this model implied.

3.2 Information gathering methods

Information gathering methods always vary to suit the various needs appear in the different types of projects. The methods include Internet research, brainstorming, sample thesis review, questionnaire, interview and so on. It is very important to define and elicit users' requirements and also collect the essential information related to the project in order to produce a complete system. So, several appropriate information gathering methods have been used in the process of collecting information in order to complete the whole project.

3.2.1 Internet research

Internet is a very useful, quick and effective resource to help in searching for any relevant information during the development of the system by just a few clicking within seconds. In addition to that, the search engines are playing the vital role in searching the appropriate sites related to the project. Search engines such as Google, Yahoo!, dmoz, searching.net and others. The information like existing system, attractive web design, tools and technologies for system design and development, project methodology and so on has gained via Internet.

3.2.2 References

Some materials like books and magazines has been reviewed and referred to obtain the information about the project methodology, tools and technologies of system development and the concepts of system development. Some books and documents had also helped in preparing the project objectives, project scope, functional requirements and non-functional requirements of the system.

3.2.3 Observation and informal interviews

By observing the current situation of dissertation registration, the processes and procedures involved had been clearly understood and identified. Furthermore, informal interview with office staff had also brought me some useful information about the policies and procedures involved in the registration as well as the existing system of thesis registration (undergraduate) in FSKTM.

3.2.4 Review sample thesis documents

The review of the sample thesis has brought me the clear picture of the structures in the thesis full report. The techniques to write the well-structured thesis report can be learnt from sample thesis documents too.

3.2.5 Discussion with supervisor

Discussions with supervisor have been held from time to time in order to obtain advices and guidance during the development of the project.

3.3 Conclusion on tools and technology

After all the tools and technologies have been reviewed and analyzed in the previous chapter, the most suitable and appropriate tools for developing the system are identified and chosen.

3.3.1 Chosen System Architecture

For this project, Online Dissertation Registration System (ODRS) is designed to be ***3-tier architecture*** to perform its functionality. The third tier will provide the database management functionality whereas the middle tier provides process management services. The middle process management tier controls transactions and asynchronous queuing to ensure reliable completion of transactions. The first tier and also the upper tier is the user interface for the users. Advantages of three-tier architecture are listed below:-

- It improves performance, maintainability and flexibility.
- Processing is centralized in the middle tier.

- ➢ It facilitates software development because each tier can be built and implemented in a separate platform.
- ➢ It allows for the parallel development of individual tiers by application specialists.

3.3.2 Chosen Application Platform

After the review is done on application platform, ***Windows XP*** is chosen to be the application platform for this project. This is because it is the most suitable operating system that can support all the development tools which is chosen to complete the project. Furthermore, Windows XP is a common-used Microsoft Operating System nowadays. This is also same with FSKTM in UM; most of the computers are currently installed with Windows XP. Therefore, the development of the project can be done easily and efficiently. Advantages of using Windows XP:

- ➢ Manageable and easy to use.
- ➢ Intelligent user interfaces than previous versions of Windows.
- ➢ Built-in application such as Microsoft IIS 5.0 and Internet Explorer 6.0 which will provide users with a fast and richer Internet experience.
- ➢ Easily connect their computer systems to the Internet with a single Internet connection.
- ➢ Always support the latest hardware standards.

3.3.3 Chosen Web Servers

Microsoft Internet Information Server (IIS) is chosen to be the web server for this project. This is because it is fully supported by Microsoft Windows XP. It also provides the powerful security, administration and development functionality. IIS is considered by experts to be just as powerful as and much easier to set up and maintain than many of its UNIX-based competitors. Advantages of using IIS:

- ➢ Reliable, it allows an administrator to restart Web services without rebooting the computer.
- ➢ It is easier to install, manage and maintain.

- ➢ Scalability and well performance.
- ➢ It offers a variety collection of server tools.

3.3.4 Chosen Web Browsers

Microsoft Internet Explorer 6.0 is chosen to be the main browser during the development of this project so as to preview all the built applications. This is suitable as the platform for this project is Microsoft Windows XP. Internet Explorer is the standard browser in the Microsoft Windows version. Furthermore, most of the tools and technologies chosen in this project are suited to Microsoft Windows environment. It is also the most common used browser.

3.3.5 Chosen Programming Languages

In this project, ***ASP.NET*** is the main programming language which is chosen to develop the entire system. It provides the easiest and most scalable way to build, deploy and run web applications. Furthermore, it is object oriented and offers built-in support for three languages which can be chosen according to the developer's familiarity. ASP.NET is programming framework from Microsoft and it has a rich set of libraries to help the developer in building the web applications. Thus, it will increase the productivity and efficiency.

3.3.6 Chosen Authoring Tools

Microsoft Visual Studio .NET will be chosen as the authoring tool for building, deploying and running the entire system. Mainly, this is because the ASP .NET is chosen to be the main programming language to develop this project. Visual Studio .NET is the best Microsoft's integrated development environment (IDE) for running and debugging programs written in ASP.NET. It has advanced compilation and caching techniques to increase the performance. Furthermore, it can integrate with existing system where the applications can connect with each other regardless of their underlying platform and object model. It has rich toolbox and powerful debugger to ease the creation of applications too.

Adobe Photoshop will be chosen to create and edit all the images of this project. This is because it is the most popular image-managing software nowadays. It provides a rich toolbox and great effect for processing images. It is very suitable to create sophisticated images for the Web, print and others.

3.3.7 Chosen Database Management System

Microsoft SQL Server 2000 is chosen to be the database management system for this project. This is because it can works well with databases of any sizes. Furthermore, it offers a variety of administration tools to ease the management of the database. Furthermore, it provides great features to handle a large amount of data during transactions simultaneously without affecting performance. Thus, it is suitable to be the database server for this project.

3.3.8 Chosen Data Access Technology

ADO.NET is chosen to be the data access technology in this project. It is the integral part of the .NET Framework of Microsoft and it is also the Microsoft's latest data access technology. ADO.NET provides an extensive set of .NET classes that facilitate efficient access to data from a large variety of sources. In addition, it enables sophisticated manipulation and sorting of data too. ADO .NET can be used in any application which connects to and communicate with data source such as Microsoft SQL Server.

3.4 Chapter Summary

In conclusion, this project will be developed by implementing the Unified Process in the methodology. Initially, the requirement elicitation, requirement analysis and system design will be started. Careful elicitation, analysis and research have been done to determine the feasibility of this project. However, the changing of the requirements can be overcome as the features of iterative and incremental. The implementation of this model assists in organizing the task of the project well especially the system design and implementation.

The chosen tools and technologies will be shown in the Table 3.1.

Table 3.1 The chosen tools and technologies

Tools and Technologies	
System architecture	Three-tier Architecture
Application platform	Windows XP Professional
Web Server	Microsoft Internet Information Server
Web Browsers	Microsoft Internet Explorer 6.0
Programming Languages	ASP.NET
Authoring Tools	Microsoft Visual Studio .NET, Adobe Photoshop
Database Management System	Microsoft SQL Server 2000
Data Access Technology	ADO.NET

Chapter 4 System Analysis

System analysis is a significant stage of systems development during which the problems and opportunities of the existing system are understood and defined. In addition, it is also important to determine the requirements thoroughly and clearly before proceeding to the next stage. Technique analysis in this context 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 objective [Whitten et al, 2004].

4.1 System Requirement Analysis

To build a system which is fault-free production, complete on time and satisfies its stakeholders, it is important to identify the requirements of the system clearly. Typically, system requirements can be divided into two categories. There are functional requirement and non-functional requirement. Functional requirement specifies the actions or services that the system must be able to perform, whereas non-functional requirement specifies properties of the system itself [Whitten et al, 2004]. The properties include reliability, security, scalability and others.

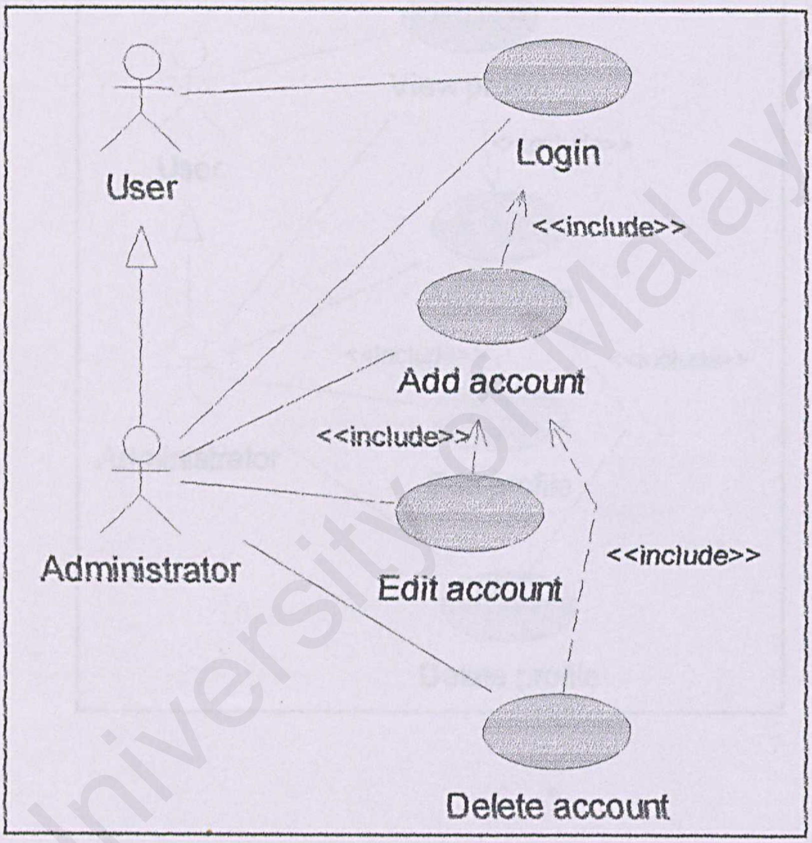
4.1.1 Functional Requirements

For the Online Dissertation Registration System (ODRS), it is divided into several sub-modules, whereby the functional requirements are described as the following:

4.1.1.1 Authentication Module

This module includes the functions of adding, editing and deleting the users of the system. Administrators are able to manage the users of the system and each user will be provided the login ID and password so as to log into the system to perform any operations.

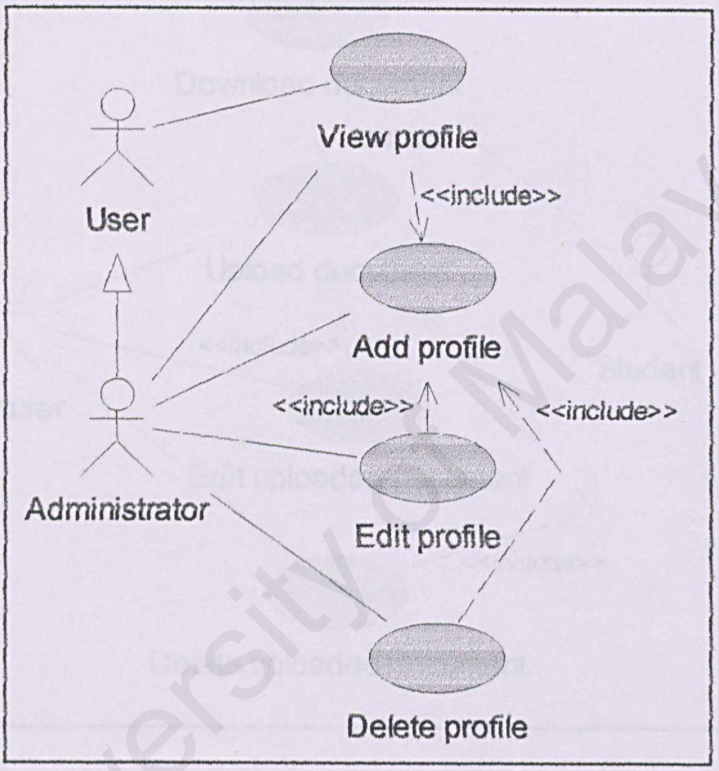
Figure 4.1 Use Case diagram for Authentication Module



4.1.1.2 Profile Module

This module includes the recording of the lecturers and students' profile. All the details will only be added, edited or deleted by administrator. All the users are only allowed to view their details through the system.

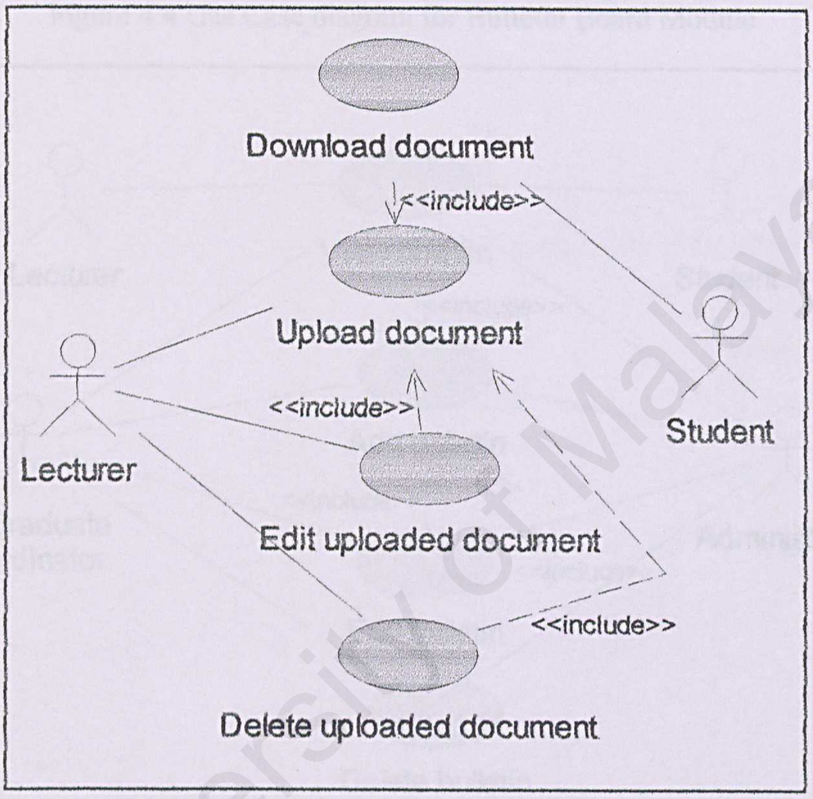
Figure 4.2 Use Case diagram for Profile Module



4.1.1.3 Documents Downloading Module

This module enables the students to view and download the forms, reports or articles uploaded by the Postgraduate Coordinator and lecturers.

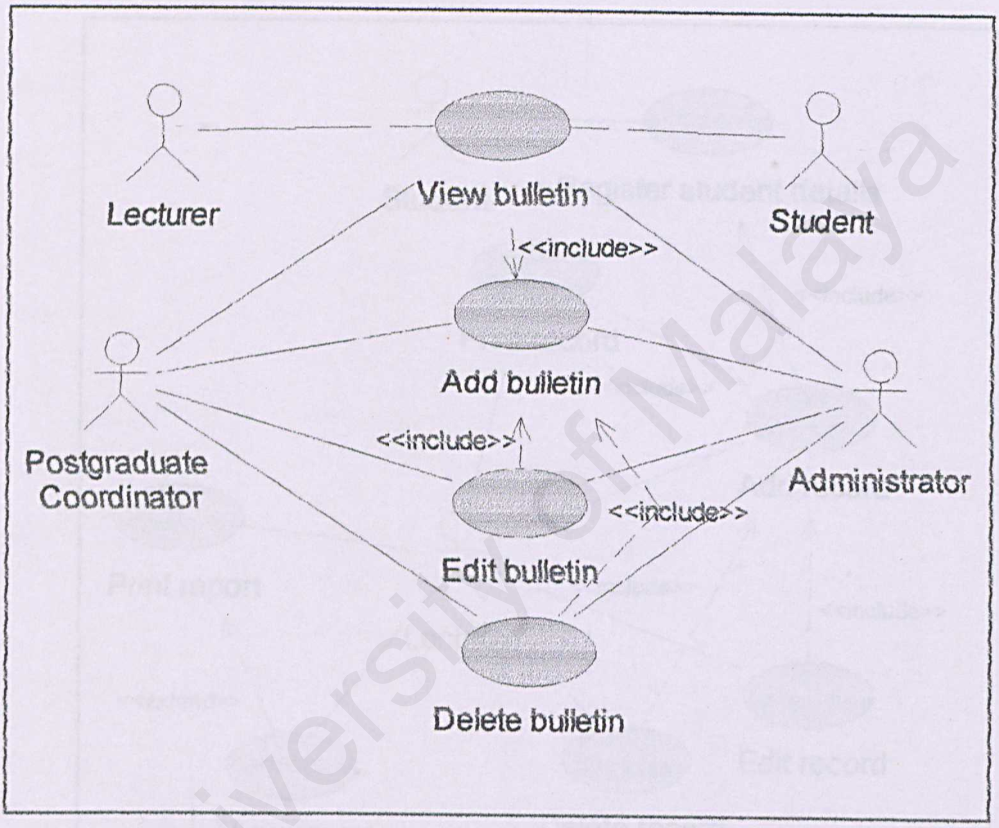
Figure 4.3 Use Case diagram for Documents Downloading Module



4.1.1.4 Bulletin Board Module

This module provides an informative board for the Postgraduate Coordinator and administrators to publish news, notices, announcements and any information to inform the students.

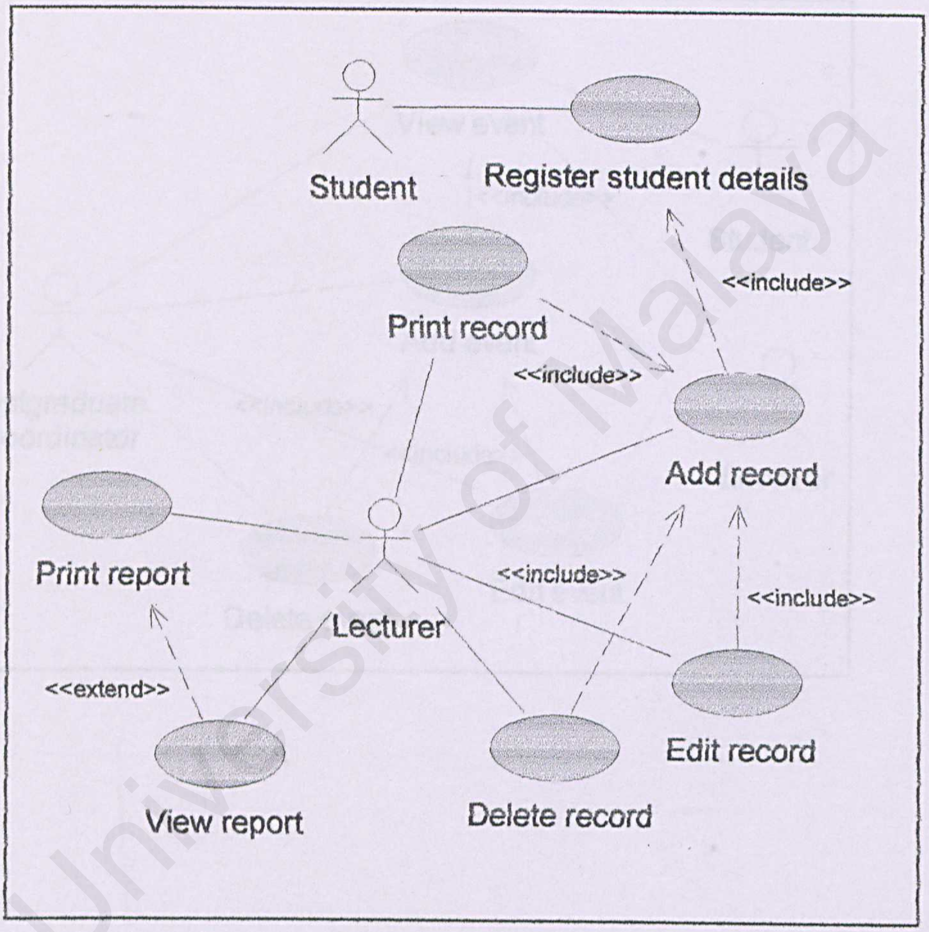
Figure 4.4 Use Case diagram for Bulletin Board Module



4.1.1.5 Discussion Record Module

This module gives an easy way to lecturers to record the details of face-to-face discussion and thus indicate the performance status of the students.

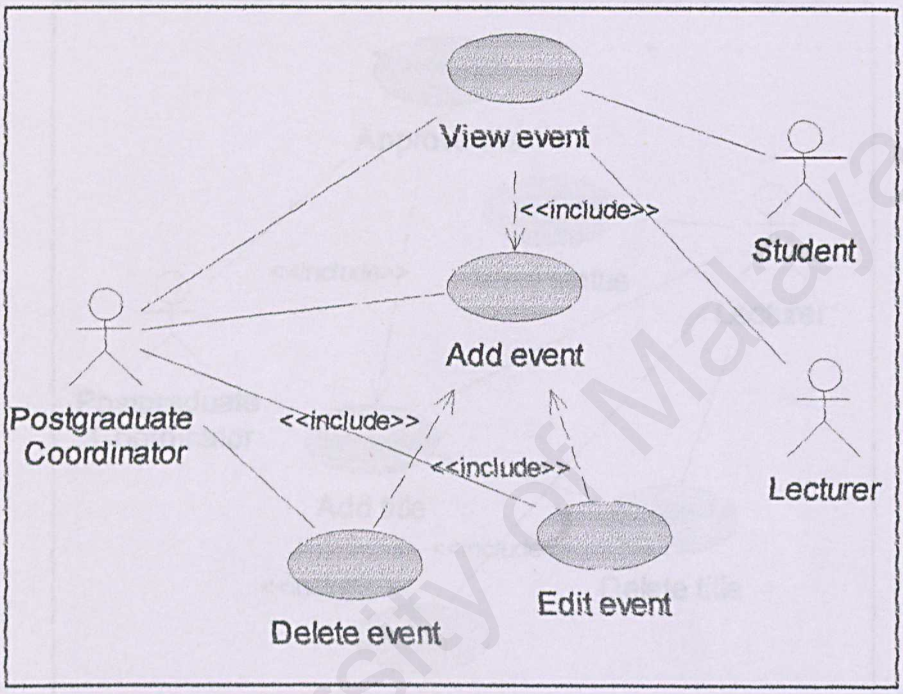
Figure 4.5 Use Case diagram for Discussion Record Module



4.1.1.6 Event Calendar Module

This module enables the Postgraduate Coordinator to state the important related event calendar to all the lecturers and students according to the academic calendar of the faculty.

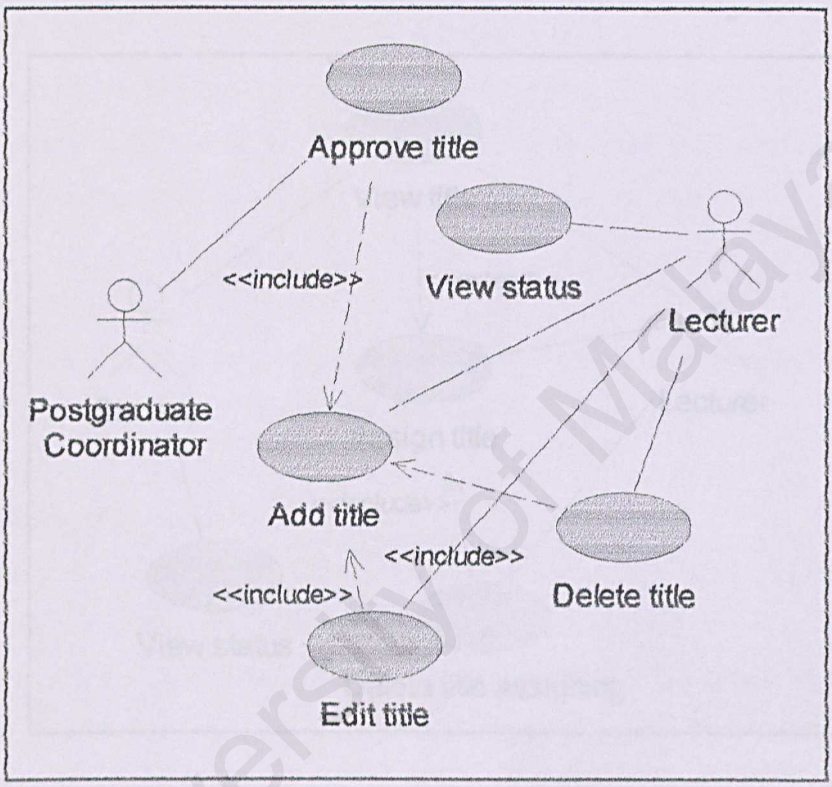
Figure 4.6 Use Case diagram for Event Calendar Module



4.1.1.7 Project Title Proposal Module

This module includes the submission of proposed titles and also the approval of the titles. The lecturers will propose the master project titles to the Postgraduate Coordinator and thus obtain the approval from the Postgraduate Coordinator.

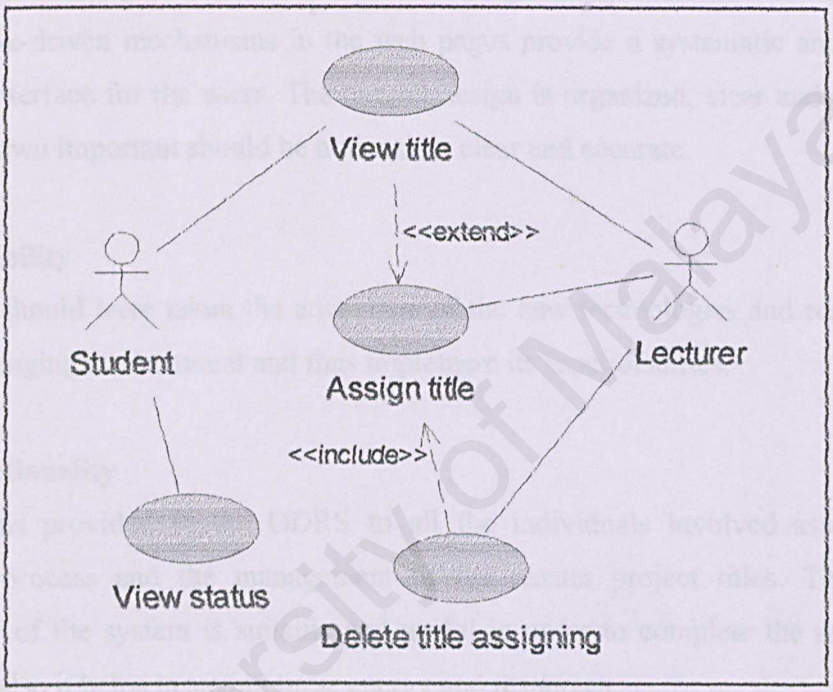
Figure 4.7 Use Case diagram for Project Title Proposal Module



4.1.1.8 Project Title Assigning Module

This module enables the students to view all project titles and thus they can approach their preferred lecturers (supervisors) to obtain their desired titles. Lecturers will register the students to the respective titles after the consensus has been met.

Figure 4.8 Use Case diagram for Project Title Assigning Module



4.1.2 Non-functional Requirements

The non-functional requirements for the Online Dissertation Registration System (ODRS) are described as following:

4.1.2.1 GUI

The Online Dissertation Registration System (ODRS) provides an easy-to-use web interface with WIMP (Window, Icon, Menu, Pointing device) features. The menu driven and hyperlink-driven mechanisms in the web pages provide a systematic and standard navigation interface for the users. The overall design is organized, clear and consistent while the shown important should be noticeable, clear and accurate.

4.1.2.2 Flexibility

The system should have taken the advantage of the new technologies and resources to adapt the changing environment and thus implement its functionalities.

4.1.2.3 Functionality

The functions provided in the ODRS to all the individuals involved assist in the registration process and the management of the master project titles. The overall functionality of the system is suitable and useful in order to complete the registration process and also it helps in saving time, energy and resources.

4.1.2.4 Portability

The system is developed with Windows platform. It is important to achieve this component so that the web pages can be viewed and accessed by the users thoroughly without any defects. Users still can access the web site regardless of the platform as long as their computers are connected to the Internet.

4.1.2.5 Usability

The system is useful to students, lecturers and the Postgraduate Coordinator in managing the registration system through the online system. The information and collections are readily available and it can be retrieved at anytime and anywhere. In addition to that, the system provides the notification messages and informative error messages if any failure of information entered or retrieving are limited or restricted towards the users.

4.1.2.6 Reliability

The system can be trusted in performing its functions and operations. Thorough and careful testing is carried out to ensure the system reliability while performing continuous testing to further improve the overall trustworthiness. The system is always available for services when requested by end-users as long as there's no any server down or crowded users problems occurred. Thus, the system should be able to perform its daily functions and operations correctly.

4.1.2.7 Modularity

Program in the system is broken into several modules that isolated from one another that each performs distinct functions. This characteristic will make testing and maintenance much easier. The errors will be easily detected and corrected.

4.1.2.8 Manageability

The ODRS is built by using Object-oriented approach, it is believed that bugs or system faults can be detected and fixed in the shortest time. Besides that, the system should be easy to manage and handle, this is to ensure that maintenance can de carried out easily and regularly.

4.1.2.9 Expandability

The system needs to be extendable and adaptable. It should be able to extend to accommodate more functionality in the future. This will allow the more advance technology to take part in the future of the system so as to produce a more usable and flexible system to the users.

4.1.2.10 Security

The system will ensure that data is protected from authorized access. Each user of the system will be given different level of authorized access to enter the system. The login

ID and password are used in this situation. Furthermore, certain important details and information can only be accessed by the administrator of the system.

4.2 Run Time Requirements

Run Time Requirement refers to the specification that the system needs to function. Run time requirement includes hardware requirements and software requirements.

4.2.1 Hardware Requirements

In order to develop the full system of Online Dissertation Registration System (ODRS), the hardware requirements below need to be fulfilled by the developers.

Table 4.1 Hardware Requirements for Developers

Type of Hardware	Requirements
Processor	450-megahertz (MHz) Pentium II-class processor 600-MHz Pentium III-class processor recommended
RAM	192MB or greater (256 MB recommended)
Hard disk	900MB of system drive; 3.3GB of installation drive Additional 1.9 GB for optional MSDN Library documentation
Drive	CD-ROM or DVD-ROM drive
Video	1024x768 or higher resolution display with 256 colour
Others	Keyboard and Microsoft Mouse or compatible pointing device, Modem 56K

In order to use or browse this Online Dissertation Registration System (ODRS) successfully, the following hardware requirements must be fulfilled by the users.

Table 4.2 Hardware Requirements for Users

Type of Hardware	Requirements
Processor	300-megahertz (MHz) Pentium II-class processor 450-MHz Pentium III-class processor recommended
RAM	128MB or higher (192 MB recommended)
Hard disk	660MB of system drive; 190MB of installation drive (850MB total)
Drive	CD-ROM or DVD-ROM drive
Video	800x600 or higher resolution display with 256 colour
Others	Keyboard and Microsoft Mouse or compatible pointing device, Modem 56K

4.2.2 Software Requirements

In order to develop the full system of Online Dissertation Registration System (ODRS), the software requirements below need to be fulfilled by the developers.

Table 4.3 Software Requirements for Developers

Type of Software	Requirements
Operating System	Windows XP Professional
Browser	Internet Explorer 5.01 or later
Web server	Microsoft Internet Information Server(IIS)
Database Server	Microsoft SQL Server 2000
Authoring Tools	Visual Studio .NET, Adobe Photoshop 7.0, .NET Framework 1.1 SDK
Documentation	Microsoft Office XP (Microsoft Word, Microsoft Office Project), Rational Rose Enterprise Edition,

The users need to fulfill the following software requirements in their computer in order to use or browse this Online Dissertation Registration System (ODRS) successfully.

Table 4.4 Software Requirements for Users

Type of Software	Requirements
Operating System	Windows XP Proffesional
Browser	Internet Explorer 5.01 or later
Database Server	Microsoft SQL Server 2000
Authoring Tools	*.NET Framework 1.1 SDK*

4.3 Chapter Summary

This chapter describes analysis stage of system requirement and run-time requirement. The system requirement includes of functional requirements, which is the actions or services provided in the system, as well as non-functional requirements, which is the properties of the system. Furthermore, the run-time requirement consists of hardware requirements and software requirements for developers (servers) and also users (clients).

The functional requirements are expressed using use-case diagram. The use-case modeling is one of the Unified Modeling Language (UML). The functional requirements are presented according to the several modules of the project. The non-functional requirements of this project include GUI, flexibility, functionality, portability, usability, reliability, modularity, manageability, expandability and security. Generally, the hardware and software requirements for developers and users are specified in order to achieve the project's objectives.

The next chapter will present the designs of various aspects in the system. These include system architecture, data design, process design, database design and user interface design. It will give a clearer picture of the system.

Chapter 5 System Design

System design is the creative process in which requirements are translated into representation of software and transforming the problem into a solution. It mainly emphasizes on the technical or implementation concerns of the system. System design is a very important factor in system development as it determines the success of a system. The design phase addresses how technology will be used in the system and how the system functionality is to be provided by the different components of the system. The information collected earlier was used to accomplish the design of the system. In this project, the Model-Driven Approach is implemented. [Whitten et al, 2004]

5.1 System Architecture

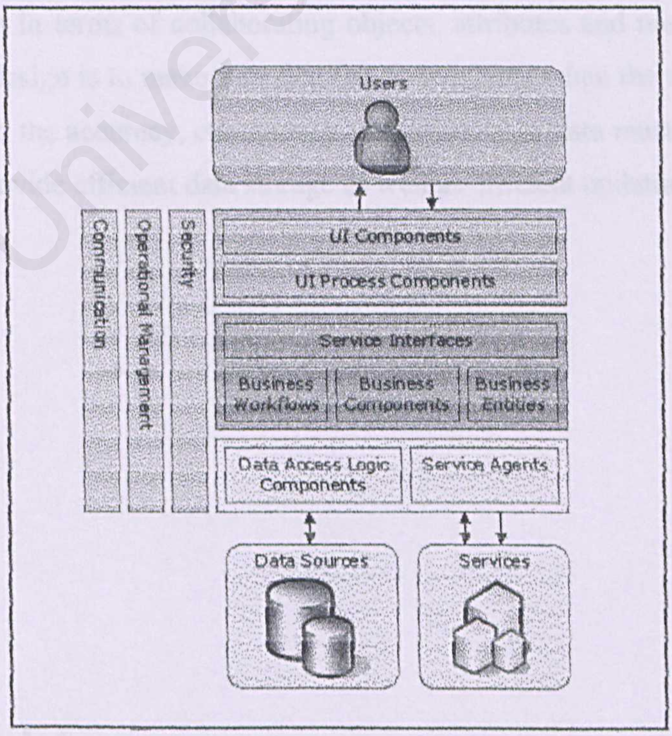
System Architecture is the technologies to be used by one or more information systems in terms of their data, processes, interfaces, and network components. Thus, designing the system architecture involves considering network technologies and making decision on how the systems' data, processes and interfaces are to be distributed. The system architecture for Online Dissertation Registration System (ODRS) is chosen based on the scope and complexity of this project. Thus, this ODRS functions in three-tier client server architecture. The three-tier client server architecture is chosen because it allows the information transfer between the web server and the database server to be optimized. Efficient middleware that supports database queries in SQL (Structural Query Language) is used to handle information retrieval from the database. [Sommerville, 2001]

The three tier client server architecture consists of three tiers which are Presentation (User Interface), Application Process Management (Business Logic) and Database Management (Data Access). These three tiers are logically separate. The description of three tiers is listed in Table 5.1. In addition, Figure 5.1 is shown about the component layers found in distributed applications and services built with .NET

Table 5.1 The description of Three-tier Client Server Architecture

Layer (Tier)	General description	Apply in the ODRS
Presentation	This layer concern about presenting information to the user and with all user interaction. It displays information, collects user input and communicates with Application Process Management Layer.	ASP.NET Web Forms and Server Controls, UI Components, UI Process Components.
Application Process Management	This layer concern about implementing the logical of the application. It involves manipulation of inputs and stored data, validation of any data that comes in from the presentation, and figuring out exactly what data source logic to dispatch depending on commands received from the presentation	Implementation in both C# and Visual Basic .NET
Database Management	This layer concern about all the database operation. It will communicate with database.	ADO.NET DataSets using SQL provider, Microsoft SQL Server 2000 database using stored procedures, Database transactions

Figure 5.1 The component layers found in distributed applications and services built with .NET



The main focus of the design for this project is to use User Interface components and ASP.NET Web Forms for the presentation tier which communicate to C# and Visual Basic .NET business components in the application process management middle tier. Then, the logical business components access a back end database, SQL Server 2000 through ADO.NET. By using ASP .NET and layered architecture, it gives advantages include:

- Separation of Presentation layer, Application Process Management and Database Management layer, no more mixing of html and ASP script
- Can be used with any database
- Better reusability and maintainability

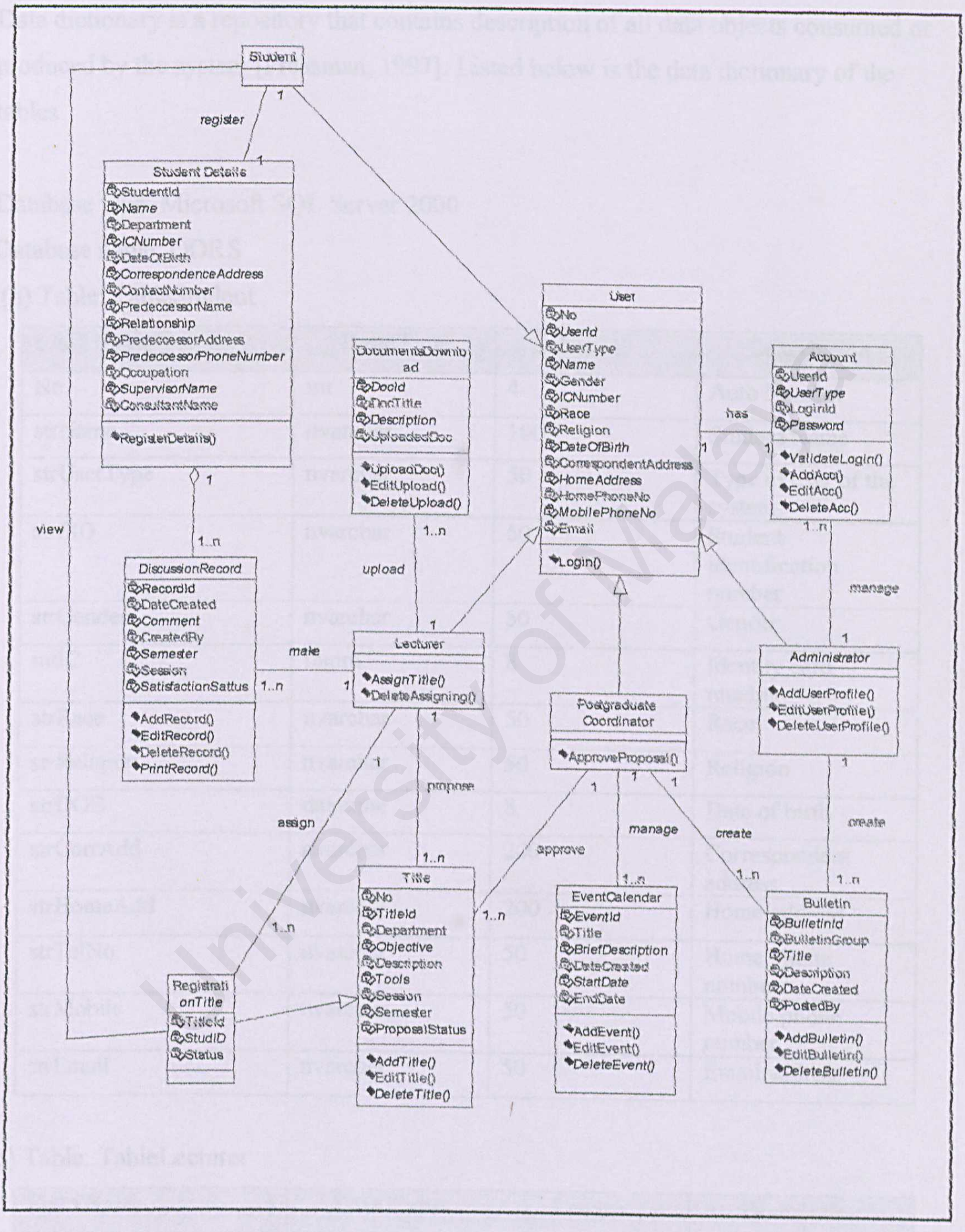
5.2 Data Design

The data design is needed to describe how the data involved in the system will be flowed to complete the related processes. Previously, the object-oriented approach is used in the system analysis. Use case diagram in the Unified Modeling Language is used to model the functional requirements of the system. Thus, the object-oriented design is used to specify the system in terms of collaborating objects, attributes and methods. The main objective of data design is to make sure that data is available when the user wants to use it. Apart from that, the accuracy, consistency and integrity of data must be assured from time to time, to provide efficient data storage as well as efficient updating and retrieval.

5.2.1 Class diagram

Class diagram is a diagram that shows class and the various relationships in which they are involved. The class diagram primary shows the structure of the system that is being developed. The relationships among the classes in the diagram provide the stepping stone for the structure of the system.

Figure 5.2 Class Diagram for Online Dissertation Registration System



5.2.2 Data dictionary

Data dictionary is a repository that contains description of all data objects consumed or produced by the system [Pressman, 1997]. Listed below is the data dictionary of the tables.

Database type: Microsoft SQL Server 2000

Database name: ODRS

(a) Table: TableStudent

Field Name	Data Type	Length	Description
No.	int	4	Auto Number
strName	nvarchar	100	Student Name
strUserType	nvarchar	50	Type of user of the system
strSID	nvarchar	50	Student identification number
strGender	nvarchar	50	Gender
intIC	bigint	8	Identity card number
strRace	nvarchar	50	Race
strReligion	nvarchar	50	Religion
strDOB	datetime	8	Date of birth
strCorrAdd	nvarchar	200	Correspondent address
strHomeAdd	nvarchar	200	Home address
strTelNo	nvarchar	50	Home phone number
strMobile	nvarchar	50	Mobile phone number
strEmail	nvarchar	50	Email address

(b) Table: TableLecturer

Field Name	Data Type	Length	Description
No.	int	4	Auto Number
strName	nvarchar	100	Lecturer Name
strUserType	nvarchar	50	Type of user of the system

strID	nvarchar	50	Lecturer identification number
strGender	nvarchar	50	Gender
intIC	bigint	8	Identity card number
strRace	nvarchar	50	Race
strReligion	nvarchar	50	Religion
strDOB	datetime	8	Date of birth
strCorrAdd	nvarchar	200	Correspondent address
strHomeAdd	nvarchar	200	Home address
strTelNo	nvarchar	50	Home phone number
strMobile	nvarchar	50	Mobile phone number
strEmail	nvarchar	50	Email address

(c) Table: TableAccount

Field Name	Data Type	Length	Description
autonumber	int	4	Unique code generated for each acount
strAID	nvarchar	50	User ID
strUserType	nvarchar	50	User type
strLoginID	nvarchar	50	User login Id
strPassword	nvarchar	50	Password used to login

(d) Table: TableBulletin

Field Name	Data Type	Length	Description
autonumber	Int	4	Unique code generated for each record as bulletin ID
BulletinGroup	nvarchar	50	Type to identify the bulletin's currency.
BulletinTitle	nvarchar	50	Bulletin title
Description	nvarchar	1000	Content for the bulletin

DatePosted	datetime	8	Date for the bulletin posted
PostedBy	nvarchar	50	User who posted the bulletin

(e) Table: TableEvent

Field Name	Data Type	Length	Description
autonumber	int	4	Unique code generated for each event as event ID
dateCreated	datetime	8	Date for the event created
strTitle	nvarchar	100	Event title
strBriefDes	nvarchar	100	Brief description for each event
strDeDes	nvarchar	500	Full description for each event
dateEventStart	datetime	8	Event start date
dateEventEnd	datetime	8	Event end date

(f) Table: TableUpload

Field Name	Data Type	Length	Description
intDocID	int	4	Unique code generated for each document
strDocTitle	nvarchar	50	Document title
strDescription	nvarchar	100	Description for the uploaded document
strID	nvarchar	50	User ID who upload the document
dateUplDate	datetime	8	Date the document is uploaded
strFileName	nvarchar	50	Filename of the uploaded document
intFileLength	int	4	File's length
strFileType	nvarchar	50	File's type

(g) Table: TableTitle

Field Name	Data Type	Length	Description
intTitleID	int	4	Unique code generated for each title
TitleId	nvarchar	30	Unique Id to identify proposed title
strSession	nvarchar	50	Session
intSemester	int	4	Semester
strId	nvarchar	50	User who propose the title
strTitle	nvarchar	150	Title
strDepartment	nvarchar	50	Department
strObjective	nvarchar	500	Title objective
strTitleDesc	nvarchar	1000	Title description
strTools	nvarchar	500	Suggested tools for the project
strStatus	nvarchar	50	The status of the proposal

(h) Table: TableRegistrationOnTitle

Field Name	Data Type	Length	Description
intTitleID	int	4	Title ID
strID	nvarchar	50	Student ID who get the title
strStatusAssign	nvarchar	50	The status of the student assigning

(i) Table: TableStudentDetail

Field Name	Data Type	Length	Description
strID	navrchar	50	Student ID
strPreName	nvarchar	50	Student predecessor name
strRelationship	nvarchar	50	Relationship
strPreAdd	nvarchar	50	Predecessor Address

strPreTel	nvarchar	50	Predecessor contact number
strEmail	nvarchar	50	Predecessor's Email
strOccupation	nvarchar	50	Occupation
strSupervisor	nvarchar	50	Supervisor name
strConsultant	nvarchar	50	Consultant name

(j) Table: TableRecord

Field Name	Data Type	Length	Description
intRecordID	int	4	Unique code generated for each record
strStudID	nvarchar	50	Student ID
strLedID	nvarchar	50	Lecturer ID
dateCreated	datetime	8	Date for each record created
strSession	nvarchar	50	Session
intSem	int	4	Semester
strComment	nvarchar	1000	Comment for each discussion

5.3 Process Design

Because of the object-oriented approach is implemented in system design, sequence diagram is used to model the logic use cases involved in the system by depicting the interaction of messages between objects in the time sequence. Once the objects' behavior and responsibilities had been determined, the sequence diagram is created to depict how the objects will interact with each other to provide the functionality specified in each designed use case of the system. Based on the uses cases built in the previous chapter, sequence diagrams have been created as shown from Figure 5.3 to Figure 5.36.

5.3.1 Authentication Module

Figure 5.3 Sequence Diagram for Login

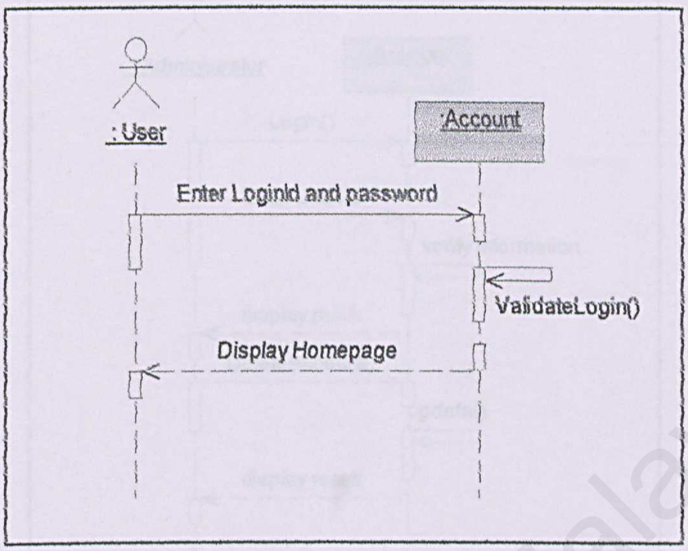


Figure 5.4 Sequence Diagram for Edit account

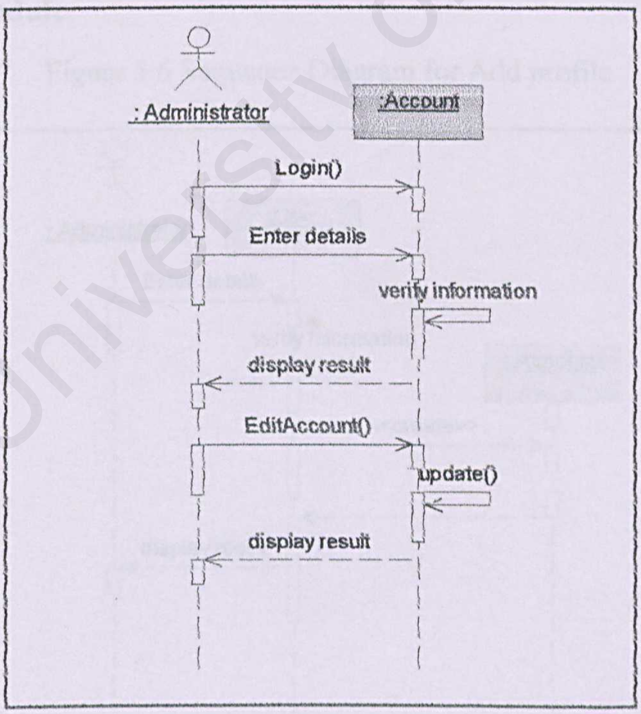
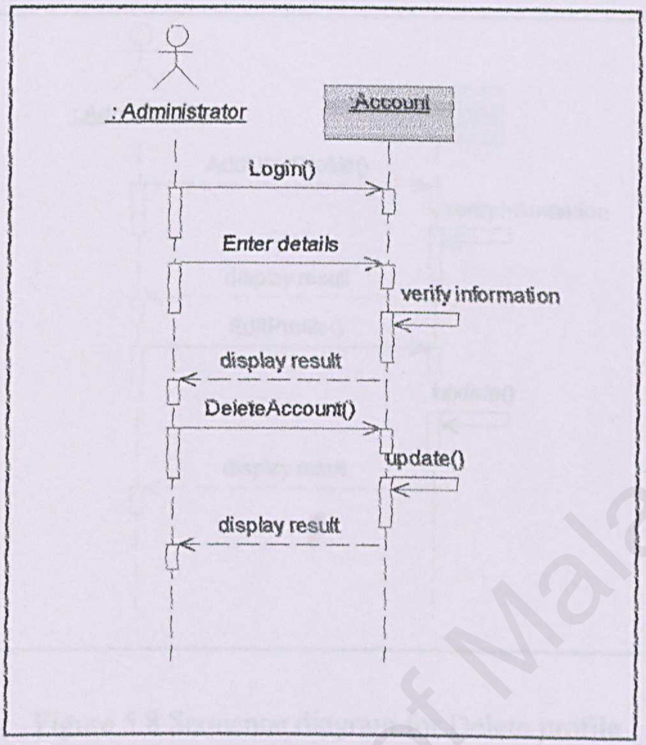


Figure 5.5 Sequence Diagram for Delete account



5.3.2 Profile Module

Figure 5.6 Sequence Diagram for Add profile

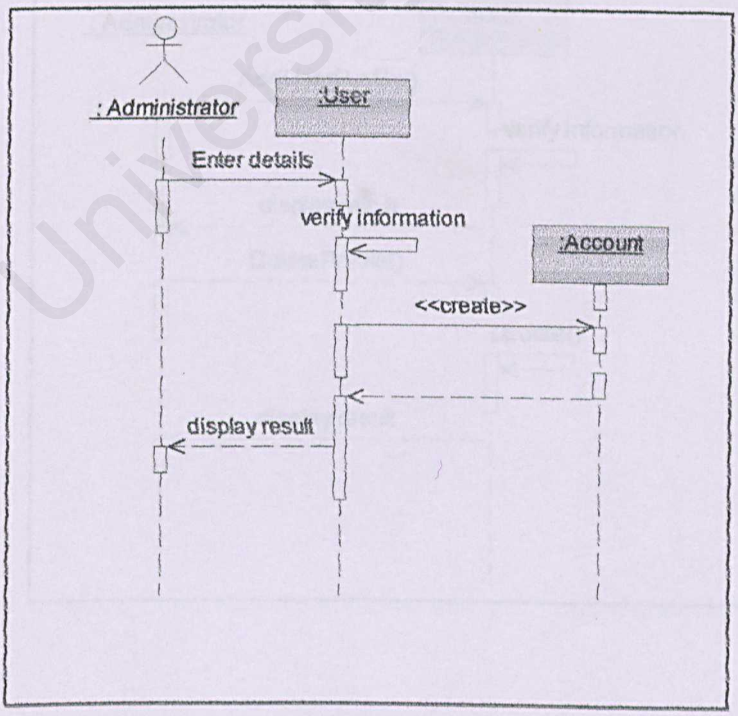


Figure 5.7 Sequence Diagram for Edit profile

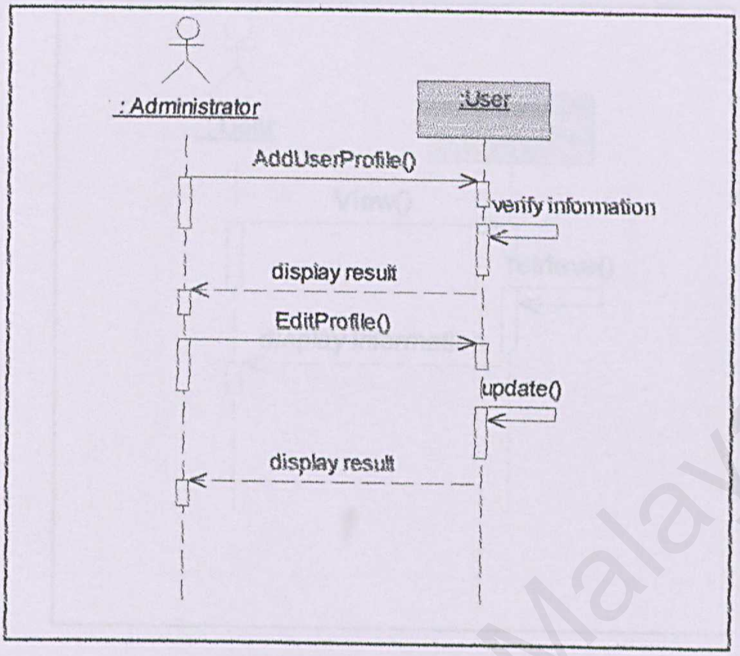


Figure 5.8 Sequence diagram for Delete profile

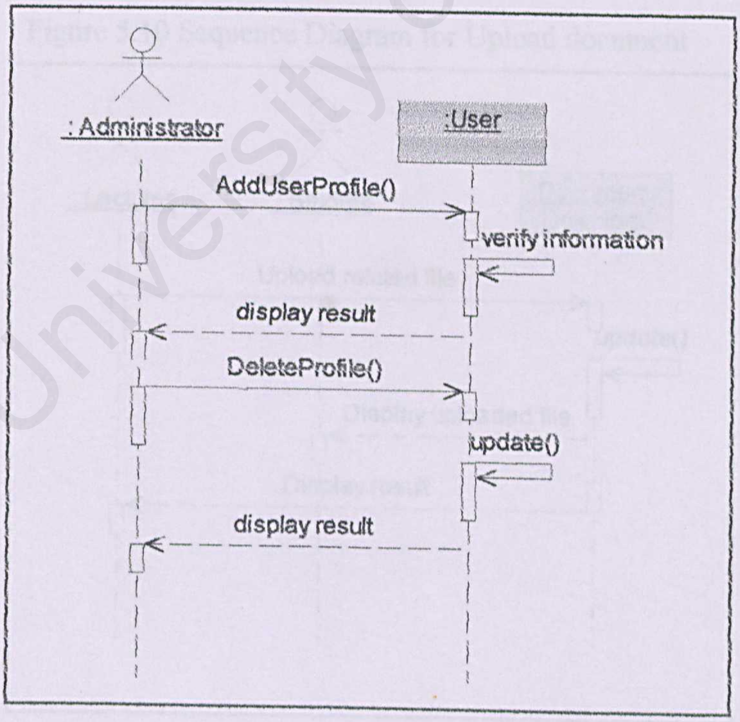
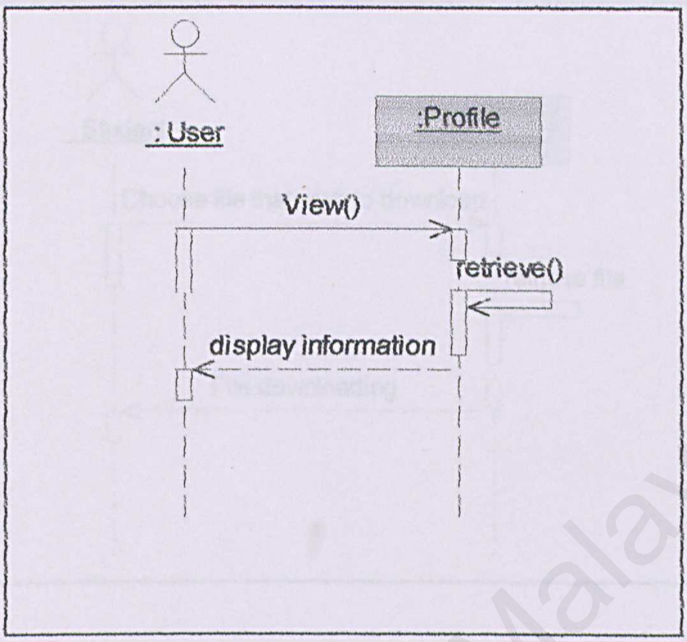


Figure 5.9 Sequence diagram for View profile



5.3.3 Documents Downloading Module

Figure 5.10 Sequence Diagram for Upload document

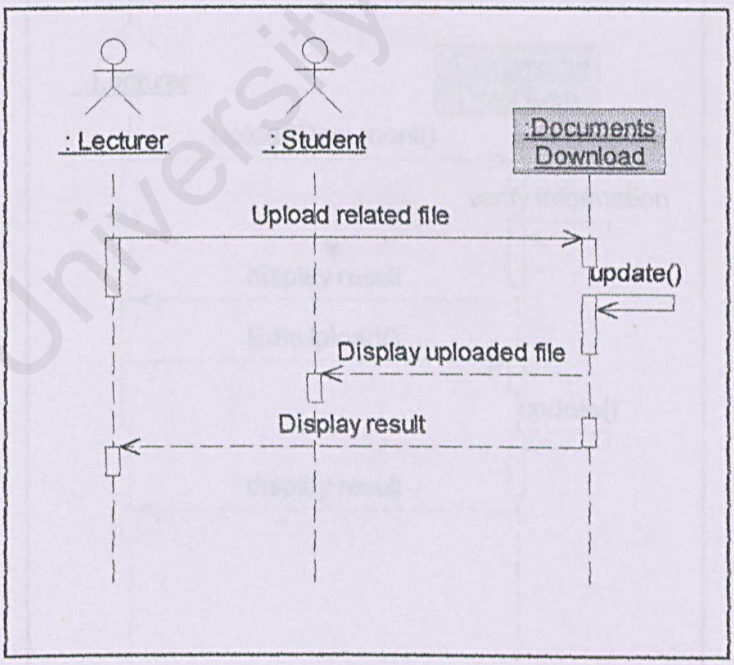


Figure 5.11 Sequence Diagram for Download document

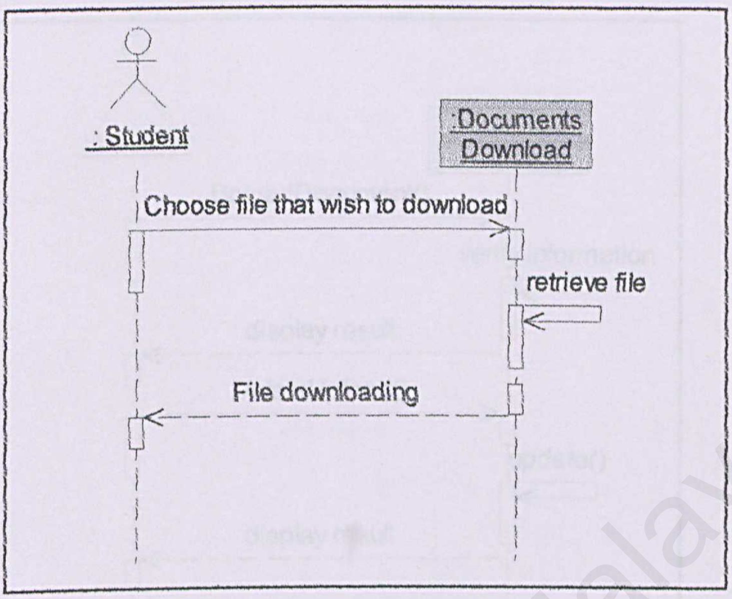


Figure 5.12 Sequence Diagram for Edit upload

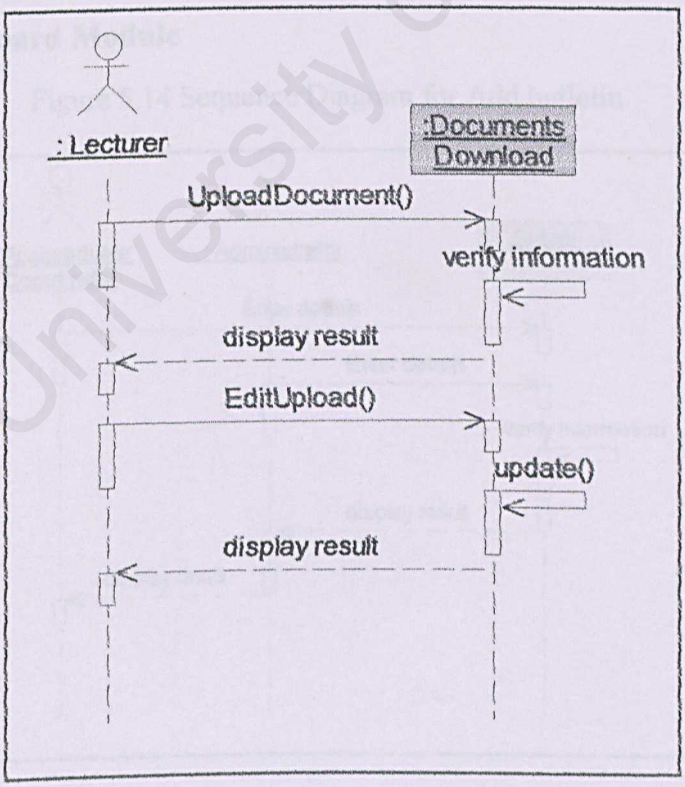
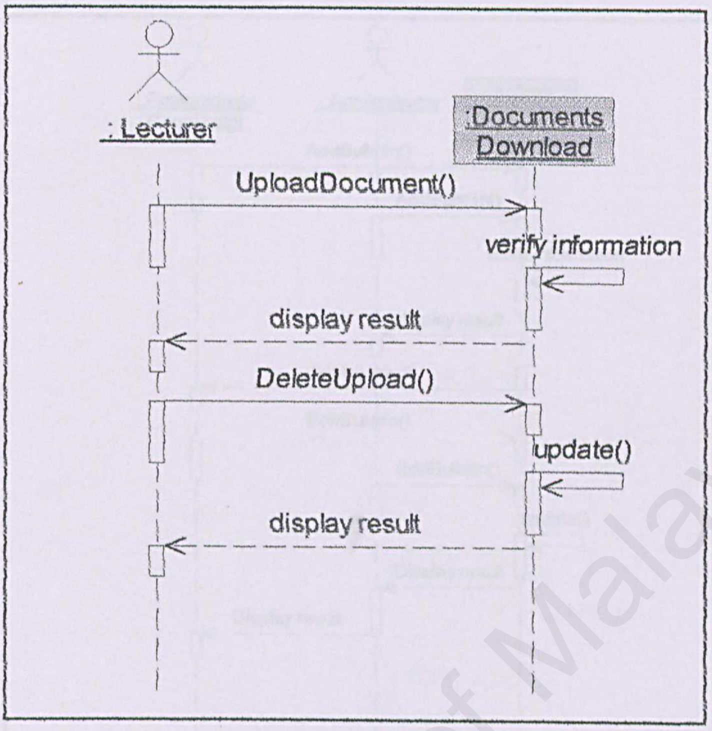


Figure 5.13 Sequence Diagram for Delete upload



5.3.4 Bulletin Board Module

Figure 5.14 Sequence Diagram for Add bulletin

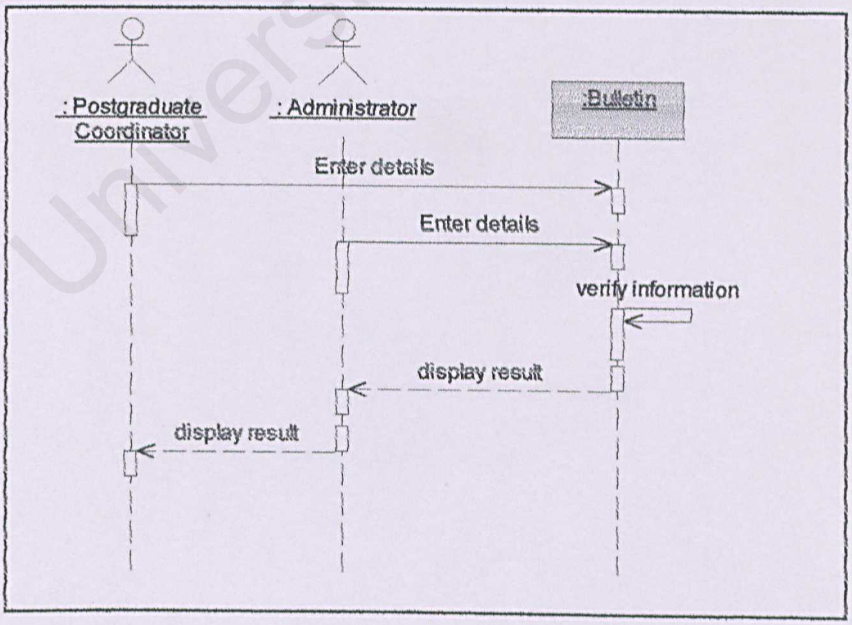


Figure 5.15 Sequence Diagram for Edit bulletin

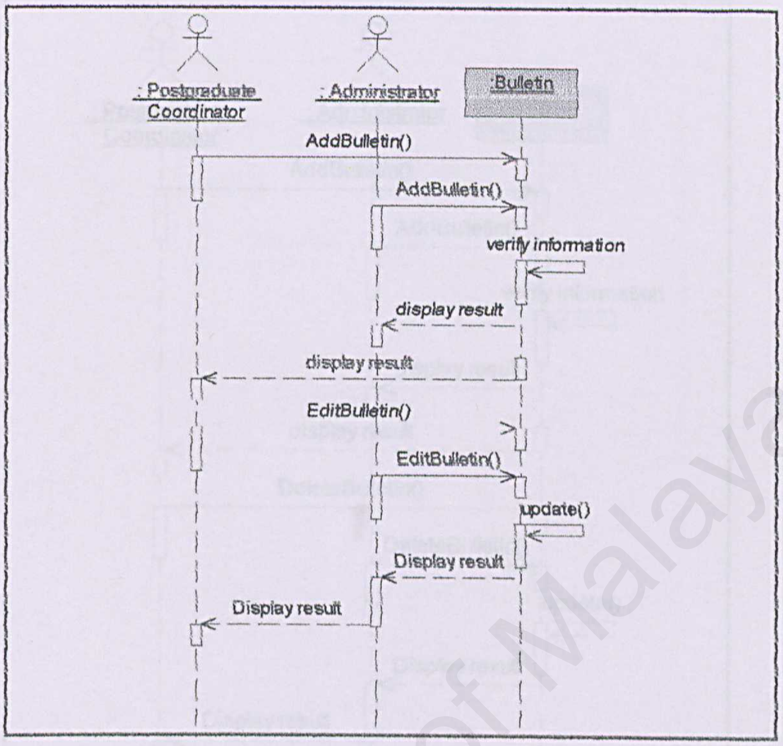


Figure 5.16 Sequence Diagram for Delete bulletin

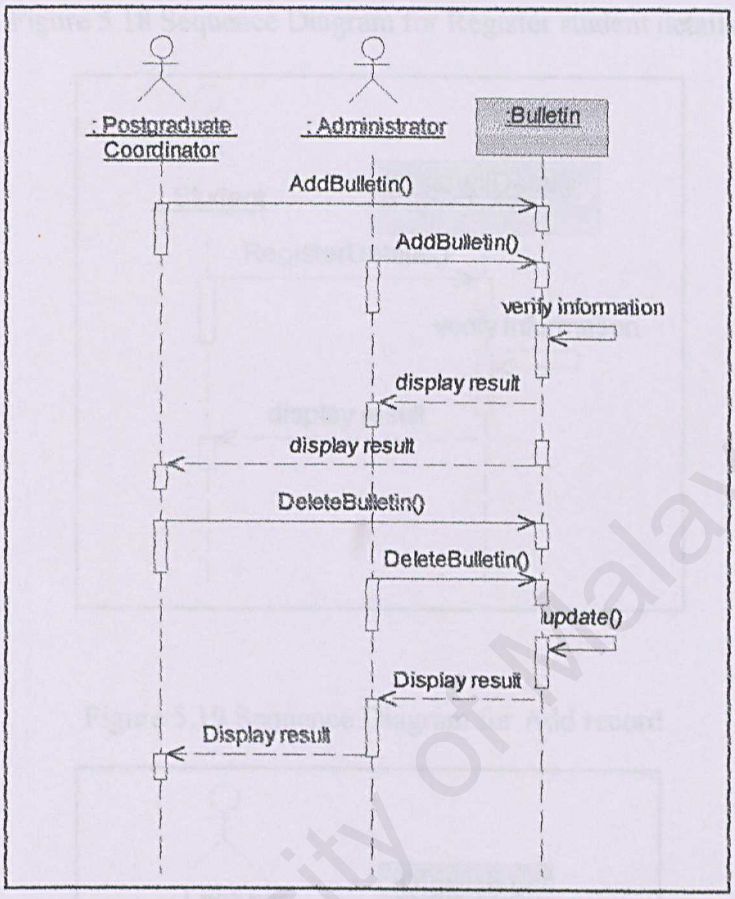
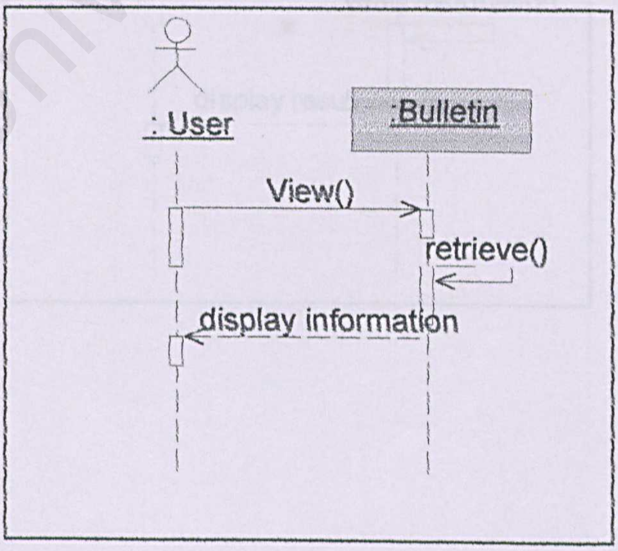


Figure 5.17 Sequence Diagram for View bulletin



5.3.5 Discussion Record Module

Figure 5.18 Sequence Diagram for Register student details

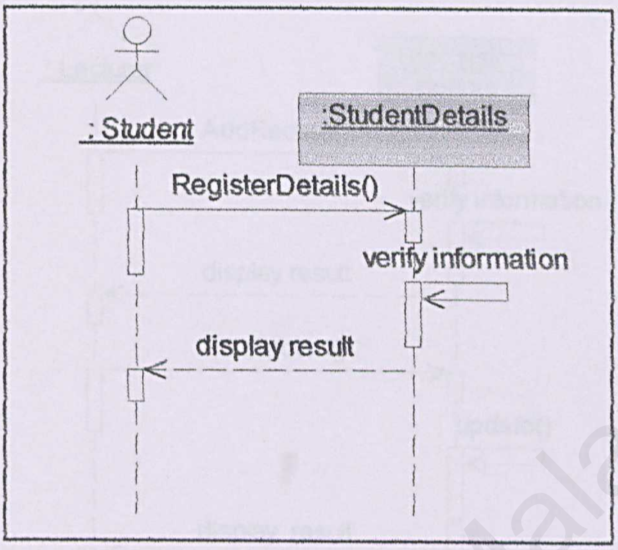


Figure 5.19 Sequence Diagram for Add record

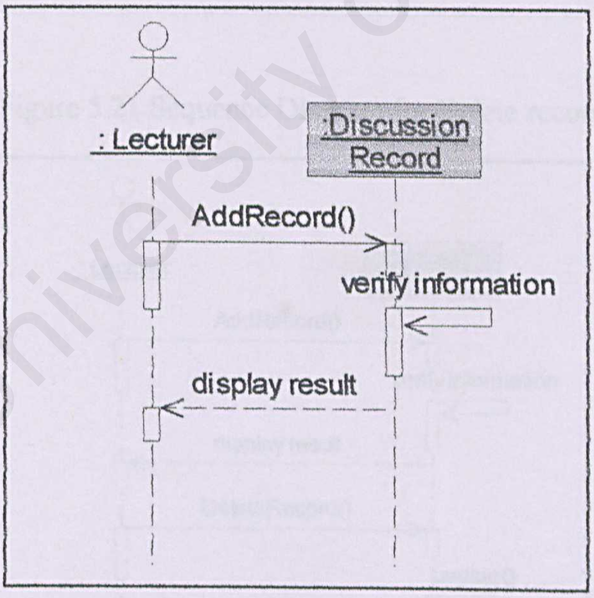


Figure 5.20 Sequence Diagram for Edit record

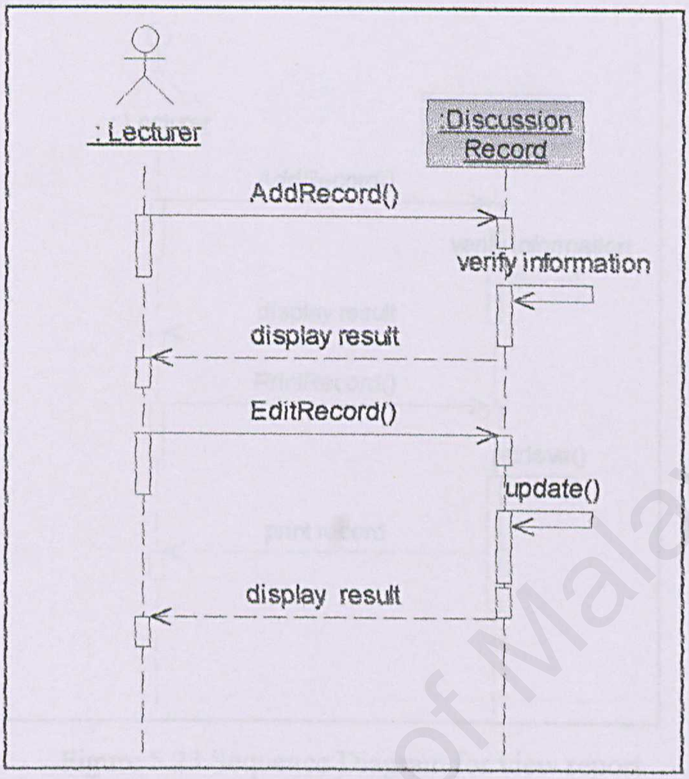


Figure 5.21 Sequence Diagram for Delete record

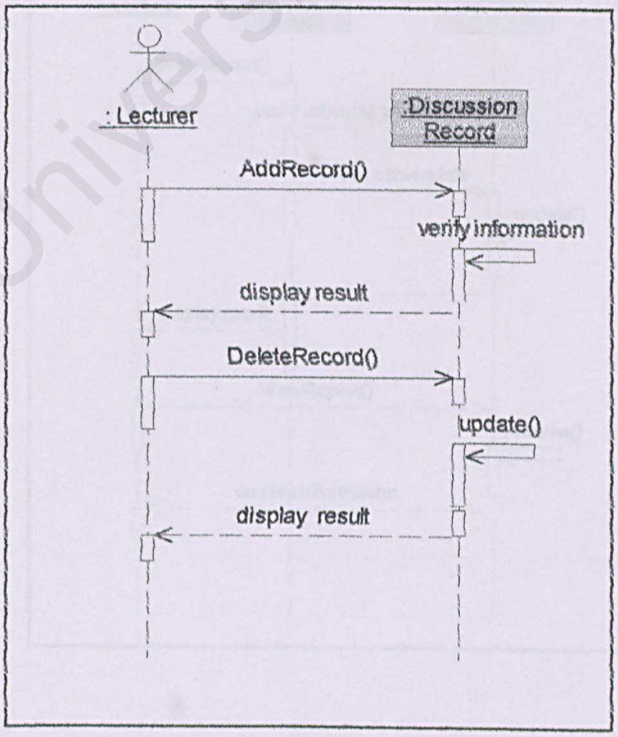


Figure 5.22 Sequence Diagram for Print record

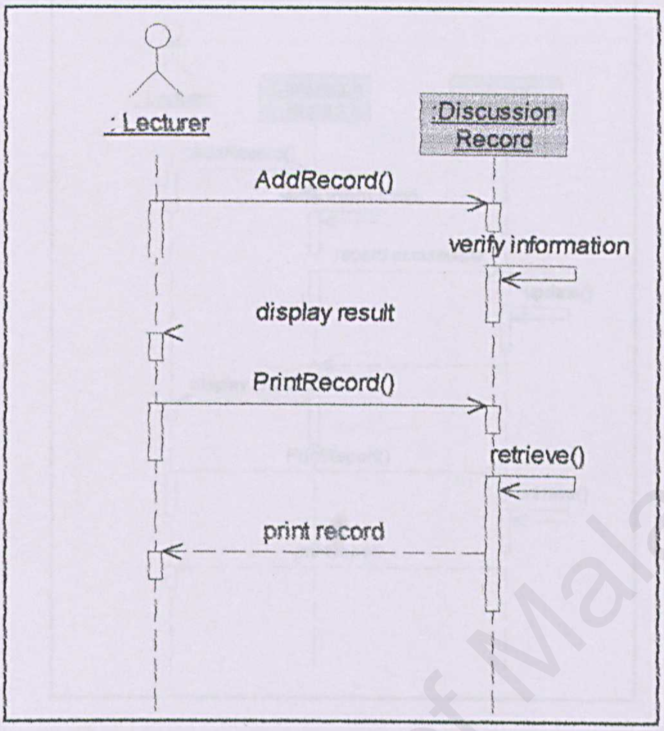


Figure 5.23 Sequence Diagram for view report

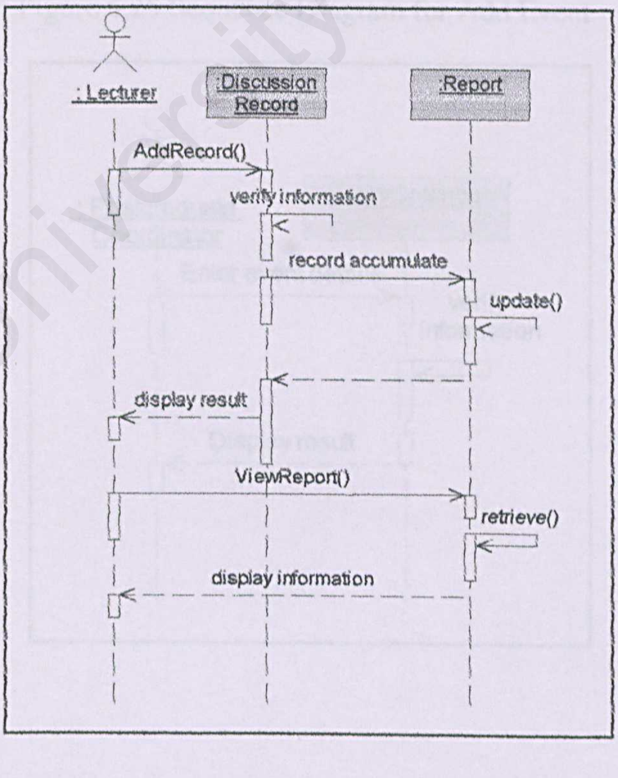
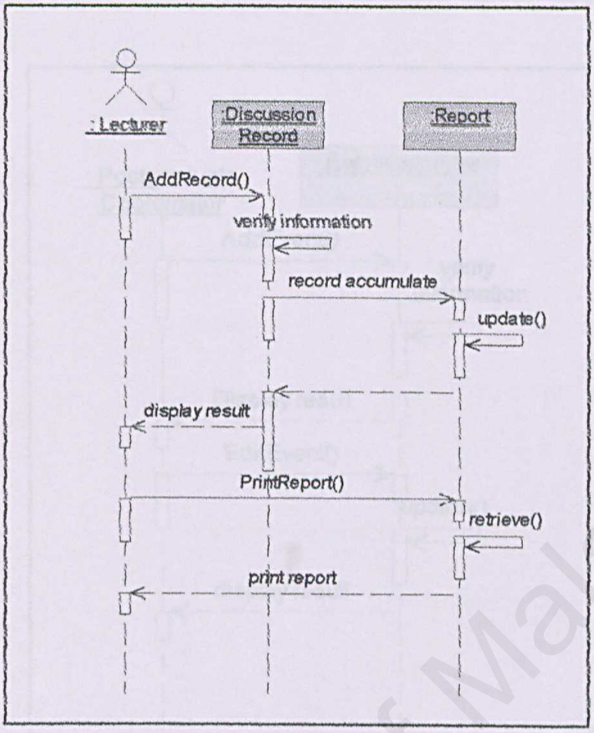


Figure 5.24 Sequence Diagram for print report



5.3.6 Event Calendar Module

Figure 5.25 Sequence Diagram for Add Event

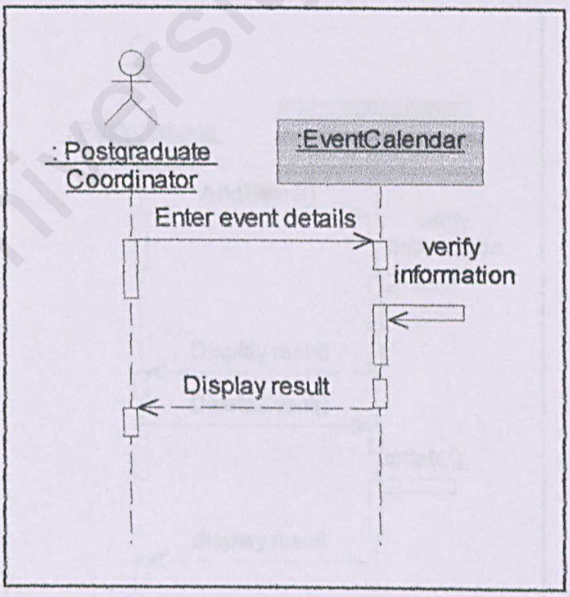


Figure 5.26 Sequence Diagram for Edit Event

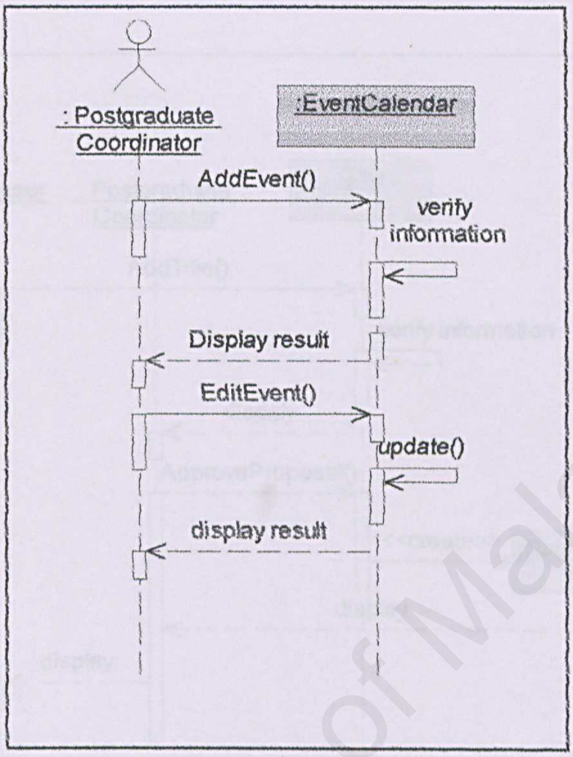
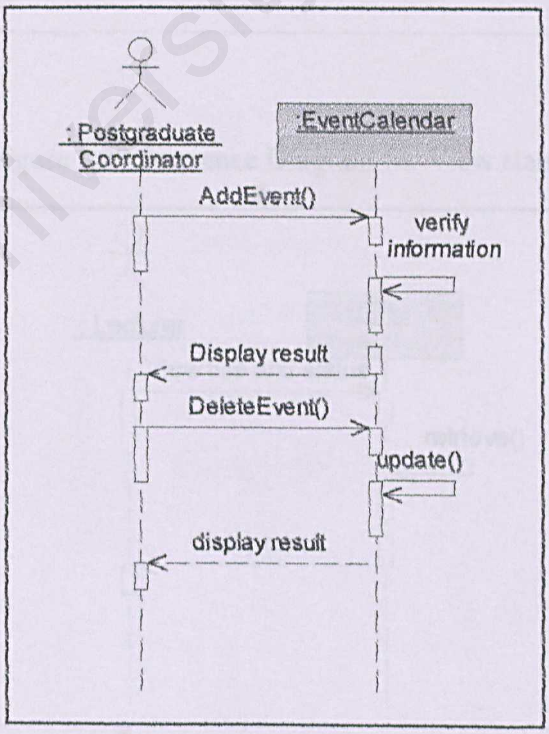


Figure 5.27 Sequence Diagram for Delete Event



5.3.7 Project Title Proposal Module

Figure 5.28 Sequence Diagram for Add title and Approve title

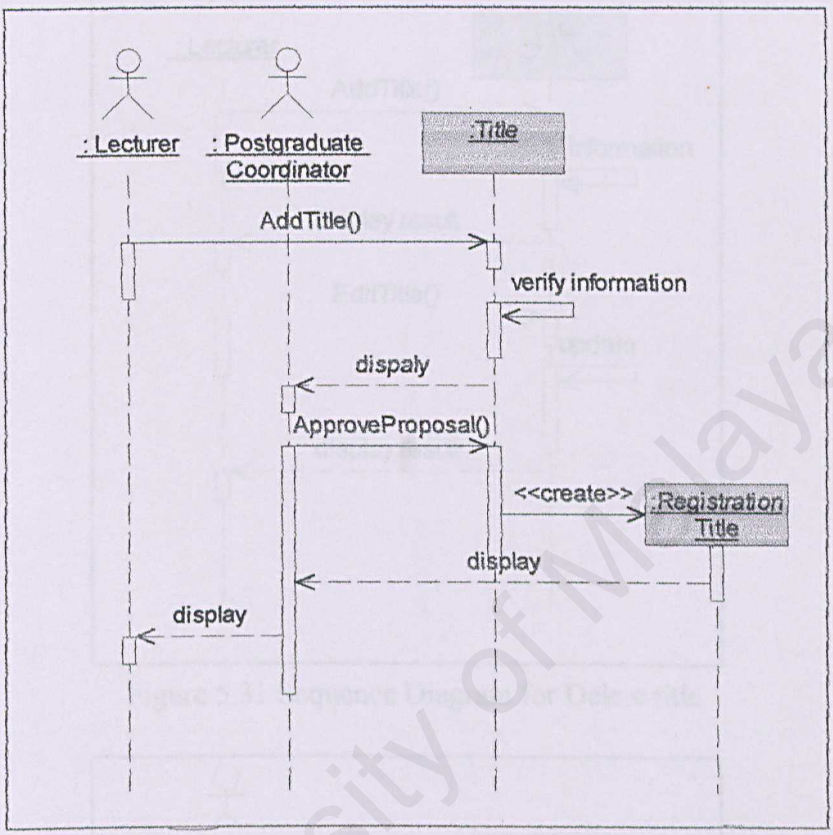


Figure 5.29 Sequence Diagram for View status

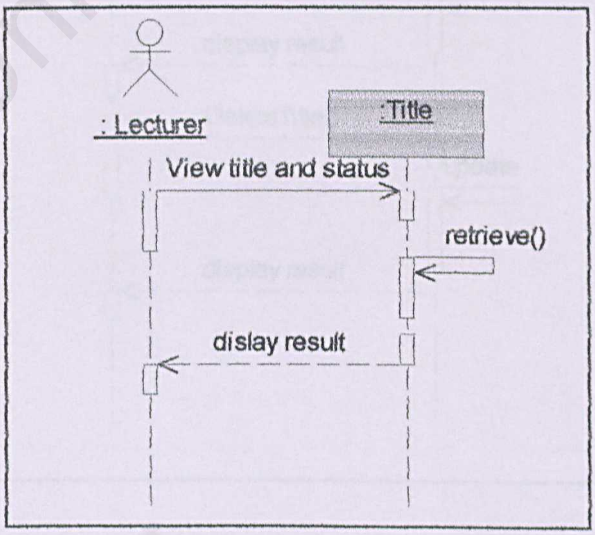


Figure 5.30 Sequence Diagram for Edit title

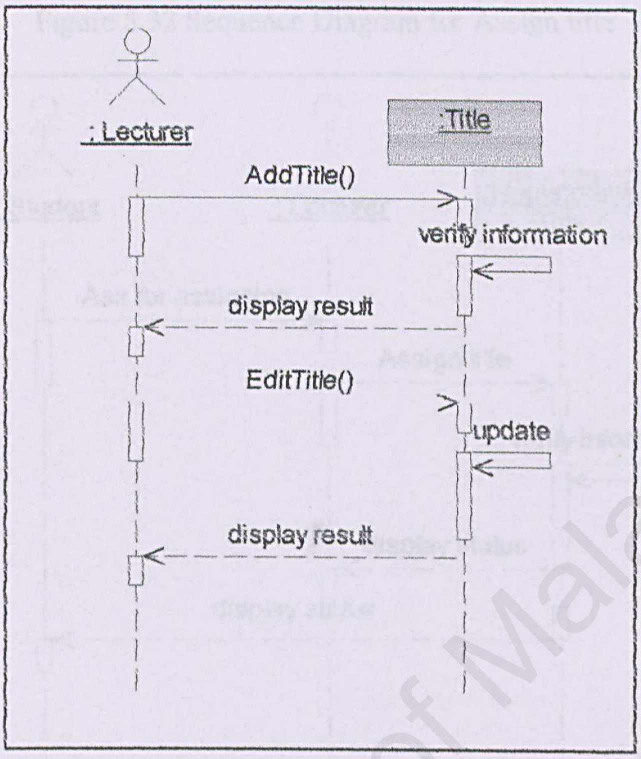
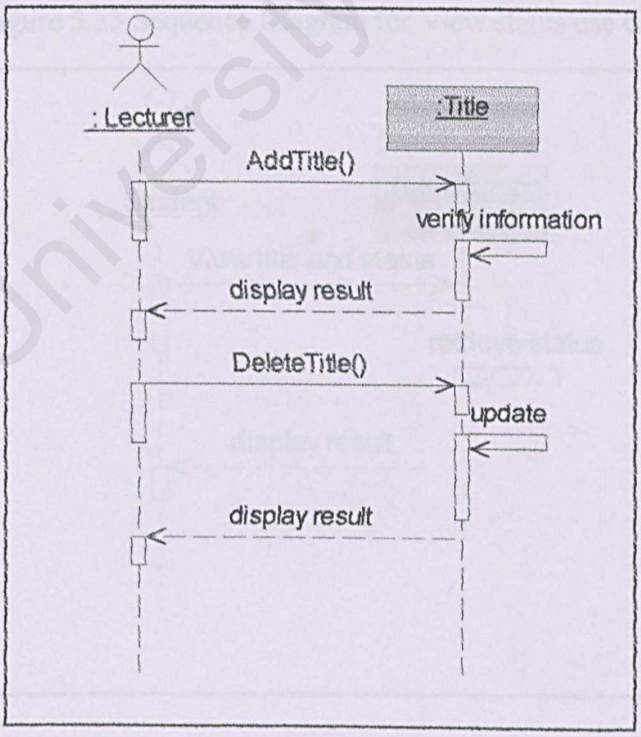


Figure 5.31 Sequence Diagram for Delete title



5.3.8 Project Title Assigning Module

Figure 5.32 Sequence Diagram for Assign title

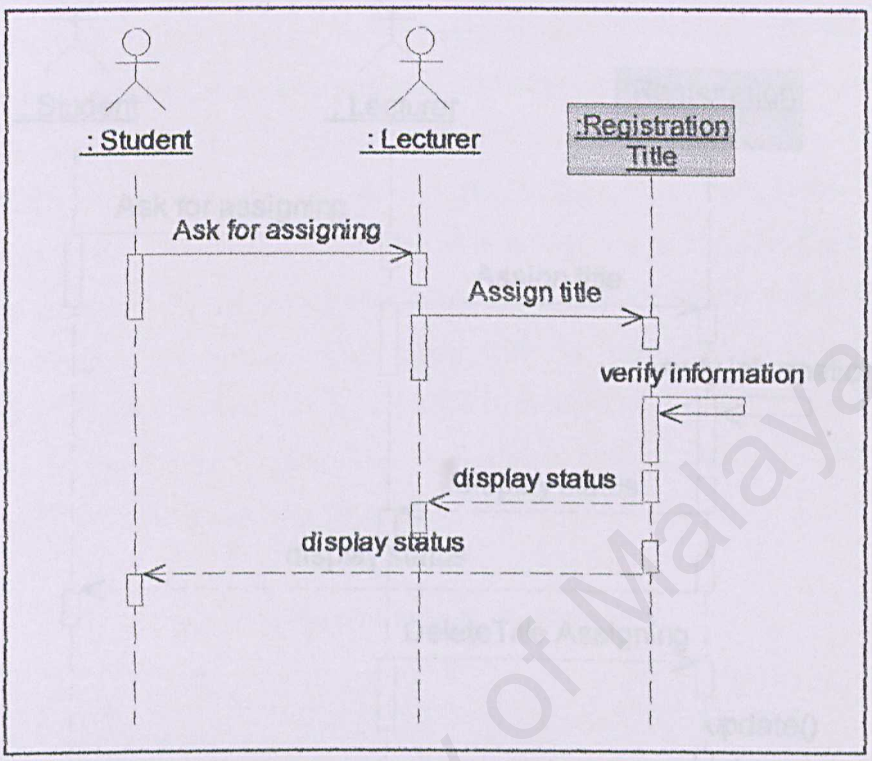


Figure 5.33 Sequence Diagram for View status use case

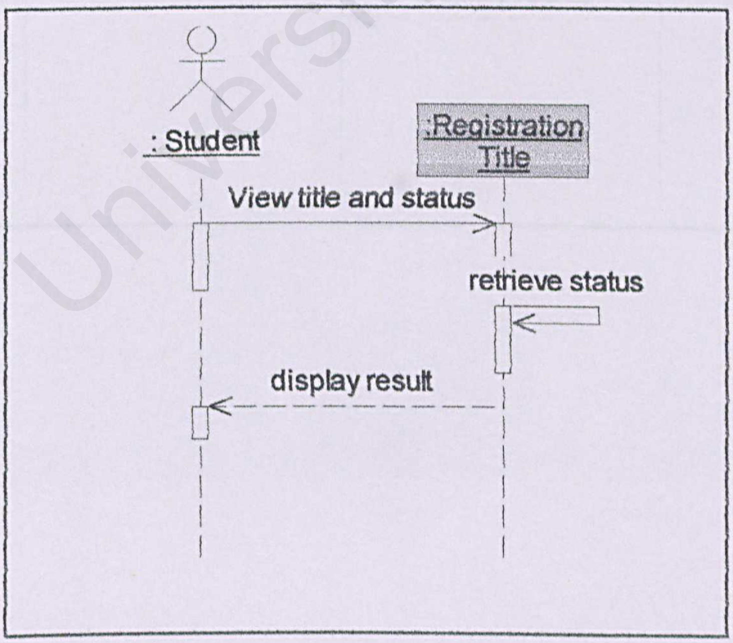
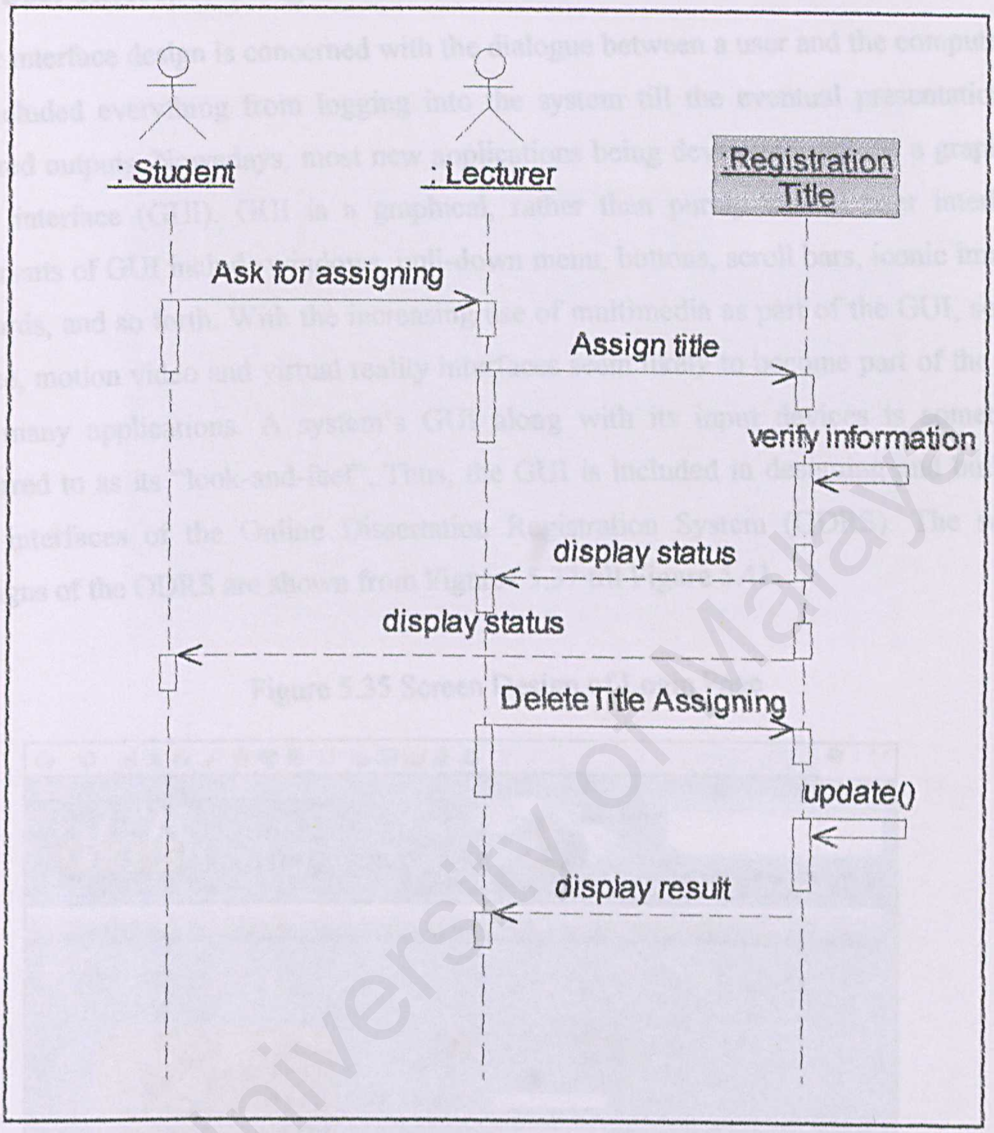


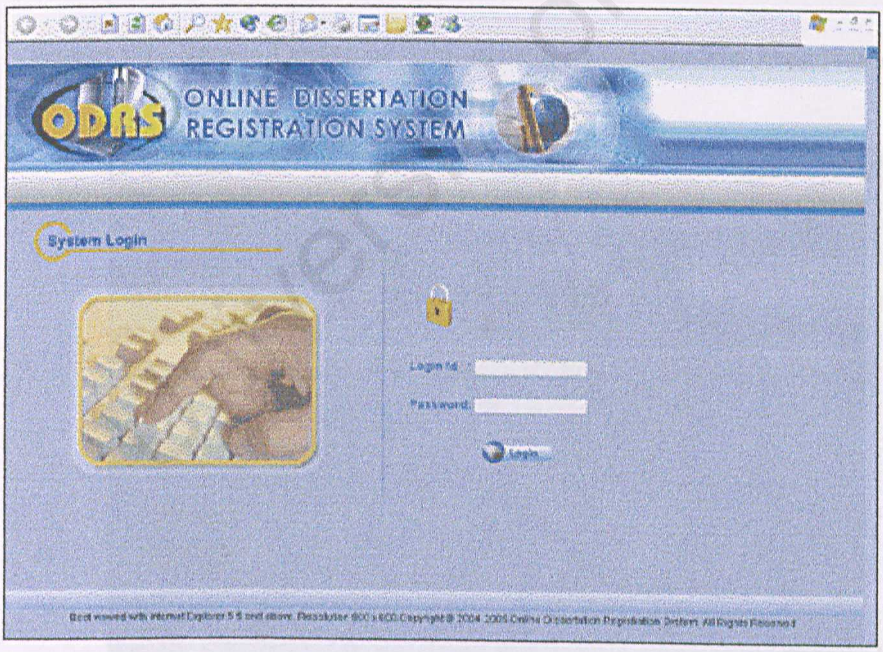
Figure 5.34 Sequence Diagram for Delete title assigning



5.4 User Interface design

User interface design is concerned with the dialogue between a user and the computer. It is included everything from logging into the system till the eventual presentation of desired outputs. Nowadays, most new applications being developed include a graphical user interface (GUI). GUI is a graphical, rather than purely textual user interface. Elements of GUI include windows, pull-down menu, buttons, scroll bars, iconic images, wizards, and so forth. With the increasing use of multimedia as part of the GUI, sound, voice, motion video and virtual reality interfaces seem likely to become part of the GUI for many applications. A system's GUI along with its input devices is sometimes referred to as its "look-and-feel". Thus, the GUI is included in designing and building the interfaces of the Online Dissertation Registration System (ODRS). The screen designs of the ODRS are shown from Figures 5.37 till Figure 5.41.

Figure 5.35 Screen Design of Login Page



Best viewed with Internet Explorer 5.5 and above. Resolution 800 x 600 Copyright © 2004-2005 Online Dissertation Registration System. All Rights Reserved

Figure 5.36 Screen Design of Student Main Home Page

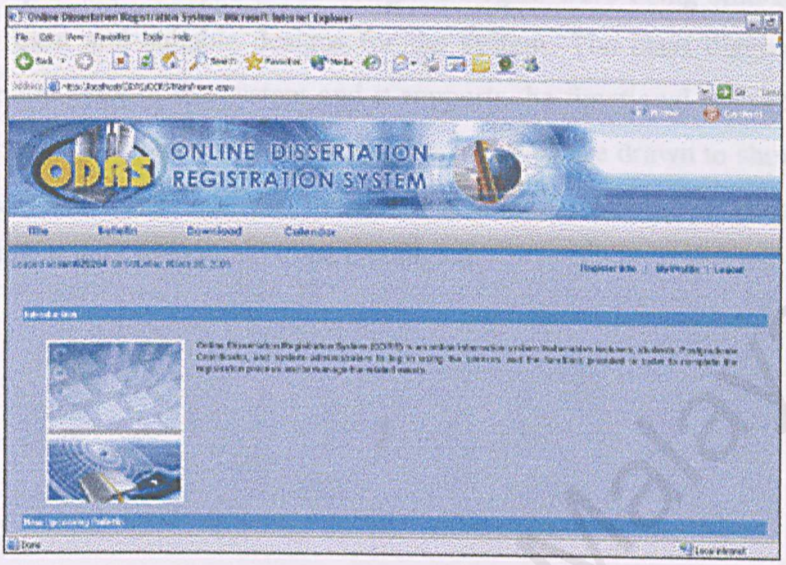
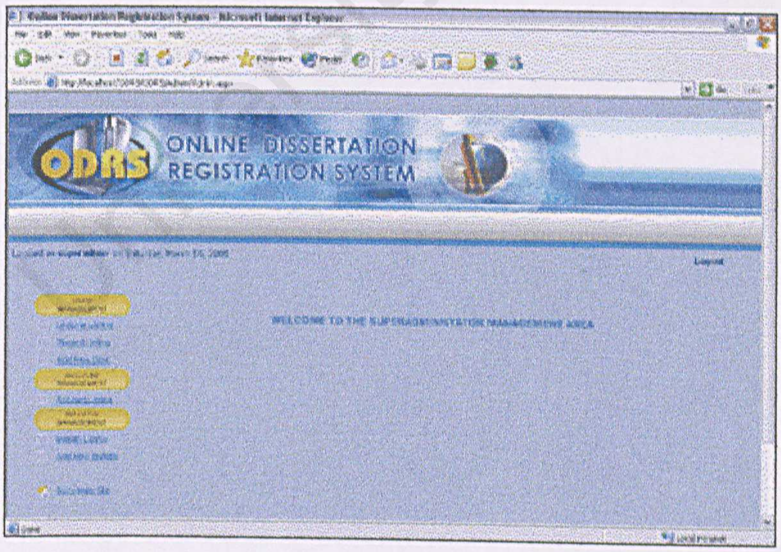


Figure 5.37 Screen Design of Administrator Management Area Home Page



5.5 Chapter Summary

This chapter has described about the physical design of the Online Dissertation Registration System (ODRS). Several design strategies were being studied in order to achieve the finest and suitable designs for the system. The class diagram is used to model the design view of the system and it supports the functional requirements of the system. The sequence diagrams of the use cases involved are drawn to show the flow of the related functions. Together with the interface design, this chapter aims to present a better understanding of the system to be developed.

Chapter 6 System Implementation

After the system designing phase on how the system should be functioning, the next process will involves the implementation phase. System implementation is a process that converts the system requirements and designs into program codes. It focuses on implementing the solution as software. This stage involves both application and database implementation where database will be created, coding of the system will be written and debugging step takes place. Generally, the requirements analysis, system design and implementation phases do not have a clear boundary. Each phase tends to overlap one another. This phase at times involves some modifications to the previous design.

6.1 Development Environment

Development environment plays an essential role in deciding the success of the project and also speed up the system development. The appropriate hardware and software should be chosen in order to obtain the good impact on the development of a system.

6.1.1 Hardware

Below listed are the hardware used to develop the Online Dissertation Registration System:

- 1.2GHz Intel Pentium IV Processor
- 256MB RAM
- 40GB Hard Disk
- Network Connection
- Other standard desktop PC components

6.1.2 Software

The table shown below is the software tools utilized in the development of this project:

Table 6.1 Software tools utilized in the development of ODRS

Software tools	Description
Microsoft Windows XP Professional	Operating System (OS)
Internet Information Server (IIS) 5.1	Web Server Host
Microsoft SQL Server 2000	Database. Build the database to store and manipulate the data.
Microsoft Visual Studio .NET	Authoring Tool. Coding the web pages
Programming Languages ASP.NET, C#, Java script	Coding the web pages
Internet Explorer 6.0	Web browser. Viewing the web pages
Adobe Photoshop 7.0	Create the buttons, headers, images and graphics of the system.

6.2 Platform Development

The platform development include setting up the development environment, create project in Visual Studio .NET, create a database in SQL 2000 Server and configure the IIS server. Servers and development tools installations are the very first step before starting off with any development work like design and coding.

6.2.1 Database Management System

The system database of the ODRS is created in Microsoft SQL Server 2000 which is the Database Management System (DBMS). It facilitates the various data retrieving, storing, deleting and other information manipulation activities. The installation process of the Microsoft SQL Server 2000 is relatively simple compare to the configuration and setup as the user just need to follow some instructions.

The Enterprise Manager, a graphical interface tool, which is used to administer the SQL Server. The Enterprise Manager can be used to configure SQL Server options, create, edit and view databases as well as perform maintenance and backups. The ODRS database is created by using the Enterprise Manager tool. This database will become the database storage for the system. After creating the database, all the tables involved in the system are identified and created. Thus, each table's fields and its property are identified and set accordingly. After all the tables have been created in the database, the relationships between the tables are established in order to enforce referential integrity. The referential integrity is an important constant on a relationship that ensures the consistency between the related tables.

6.2.2 Web Server

The Web Server chosen for the development of Online Dissertation Registration System is the Internet Information Services 5.1 (IIS). The installation of IIS does not require much manual configuration from user as the default configuration for the web server is enough for the normal system development environment. IIS enable the user to maps local directory to virtual directory and creates local web site.

6.3 Coding and Project Development

Firstly, the coding development is basically the construction and implementation of the system utilizing the necessary means to achieve the proposed specification. Mainly, Online Dissertation Registration System is created by using Microsoft Visual Studio. ASP.NET C# is used to code all the web forms and pages involved in the project as The Microsoft Visual Studio .NET is the most apparent tool for working with ASP.NET. This tool has the Integrated Development Environment (IDE) which enables the user to develop a system via a Graphical User Interface (GUI). Furthermore, it enables easy performance of the many complex programming and database tasks required in the creation of a web site, as well as the incorporation of HTML formatting and layouts, graphics and other multimedia components.

Microsoft Visual Studio .NET development platform will replicate a copy of the project files directly into Inetpub and the folder wwwroot which enable the user to run the system. Besides that, it also gives the user to view the web pages through browser in order for the user to debug the system in a runtime environment.

As mentioned before, ASP.NET C# is chosen as the server side scripting language. The server side scripting is executed through the web server of the host of the system and enables the system to invoke user defined functions and perform data interaction between the user and the database system. Besides, JavaScript is used as the client script together with HTML and Cascading Style Sheets (CSS) are used to design web forms and create the interesting interface for the convenience of the user.

6.3.1 Review Program Documentation

The initial step to be taken in project development phase is to review the program documentation that was prepared during the earlier stages. The program documentation prepared in the system design phase consists of architectural view, concepts and controls, module flow diagram, data dictionary and also the sample layout of the interface. The documentation provides a guide and an understanding of the works that need to be done in the coding phase. After reviewing the program documentation, designing the program is the next following process after that. The designing of program determines how the program can accomplish the features and functions that are described in the program documentation and developing a logical solution to the programming problem.

6.3.2 Prototype Development

The prototype development is basically the construction of the system utilizing the necessary means to achieve the proposed specification. ODRS is mainly consists of the ASP.NET. This is the core tool of prototype development that facilities tasks such as editing, debugging and coding. Besides that, in order to create an interesting and user-friendly web pages for this system. The various techniques included in the Photoshop are used to create the attractive images and icons. Various techniques like blending, masking, cropping, smoothing and others powerful techniques are utilised.

6.3.3 System Coding

System coding is referred as converting the prior system design into a working and full functional system. This mainly involves software programming and preliminary testing of the prototype model. Programming is the process of transforming the structure shorts, logical and physical data flow diagrams, and also interface designs into a working prototype model. The testing that involves here is the development testing. Preliminary testing that focuses on the correctness of individual modules and integration of multiple modules of the prototype model will be conducted.

6.3.4 Coding Approach

Coding the program is the process of writing the program instructions that implement the program design which translate the design specifications into a machine-readable format. There are two types of coding approach which are top-down and bottom-up. The bottom-up coding is based on coding the lower-level modules initially and leaving the high-level modules merely as skeletons that are used to call the lower modules, whereas the top-down approach is the reverse of the bottom-up approach. In ODRS, the bottom-up approach is chosen in the development. Initially, each lower-level function and procedure was developed individually which are then integrated into appropriate high-level modules accordingly. Bottom-up approach offers the advantages listed below:

- Testing can be conduct on some of the modules while the others are still under construction.
- Critical functions can be coded initially to test their efficiency.
- Increase the development process as the lower-level modules or functions can be built independently and simultaneously without waiting or delaying the others.
- Faults are easier to be detected.

6.4 Coding Styles

The coding style is a very important attribute to determine the readability and maintainability of the source codes. With a clear and systematic coding style, it helps the programmer to see the codes easier in order to help the programmer on maintaining and also debugging the system. Elements taken into considerations while coding an easy to maintain and enhance system are internal documentation, standard naming convention and etc.

An ASP.NET is a primarily a scripting environment; while languages used to develop an ASP.NET are HTML, C# and JavaScript. Web Forms page consists of two parts: visual elements (HTML, server controls, and static text) and the page's programming logic. Visual Studio .NET stores each of these components into separate files. The visual elements are created in an .aspx, where as the logical code is in a separate class file, called the code-behind class file (.aspx.cs). It is written in C#.

Because the .aspx file is compiled dynamically when a user browses the page, its relationship to the class file is established with script directives at the top of the page. Specifically, the Inherits attribute of the @ Page directive is used to specify the class file from which the .aspx file is derived. For examples,

Table 6.2 Example codes for the Inherits attribute of the @ Page directive in .aspx

Login Page	<%@ Page language="c#" Codebehind="Login.aspx.cs" AutoEventWireup="false" Inherits="ODRS.Login" %>
AddNewUser Page	<%@ Page language="c#" Codebehind="AddNewUser.aspx.cs" AutoEventWireup="false" Inherits="ODRS.Admin.AddNewUser" %>

Server-side includes is a mechanism which use to insert information into a file prior to processing. C# implements only the '*using*' pre-processing directive of this mechanism. This directive can be used to insert the content of another file into a .cs file before the framework processes the .cs file.

6.4.1 Include File

The syntax: *using namespace.filename* is used in the *included file* coding style of ASP.NET. In this context, *using* is the keyword that indicates the type of path which using to include the file whereas *filename* is the path and file name of the file which wants to include. Included files do not require a special file-name extension. For example, using *ODRS.classes* is used in the project.

6.4.2 Standard Naming Convention

The coding style is a very important attribute to determine the readability and maintainability of the source codes. With a clear and systematic coding style, it helps the programmer to see the codes easier in order to help the programmer in maintaining and debugging the system. One of the elements that take into consideration is the naming conversion standards. Naming convention is an abbreviation of the control name or the object name. Standard naming convention provides programmers with easy identification of variables. In the system, the naming convention is used to ensure the uniformity of the control and command names. By using these standards, the consistency, standardization and readability of the codes are increased. The following shown are some of the standard conventions used in the codes of the system.

Table 6.3 Examples of the standard naming conventions used in ODRS

Related control/object	Naming in the codes	Description
Form	FrmAddNewEvent	Add new event form control in the web form
Text Box	TxtbTitle	Text box to input the title
Label	lblErrorMsg	Label to display the error message.
Button	BtnAdd	Button 'Add' to submit the form.

Server-side scripting requires the RUNAT attribute set to Server so that the script should be executed on the server rather than the client (browser). For example,

Table 6.4 Examples server-side scripting(ASP.NET) used in ODRS

Label for displaying error message	<asp:label id="lblErrorMsg" runat="server" BackColor="#A5BBD1" CssClass="alertFont" ForeColor="Red"></asp:label>
Button to submit the form	<asp:button id="btnAdd" runat="server" BackColor="#2199D3" CssClass="TitleFont" ForeColor="White" ToolTip="Add" BorderStyle="Ridge" Text="Add">
Text box to input the title	<asp:textbox id="txtbTitle" runat="server" CssClass="basicFont" Height="18px" Width="300px"></asp:textbox>

6.4.3 Spacing, Functional Segments and Comments

Indent the code according to the functional segments (for example, the *for* loop or the *if...else* statement). Spacing or line brake in between different section of the codes to enhance readability. In addition, each specific functions and statements of codes are including the comments. These comments are important and useful for the future maintaining by the programmers. It helps in increasing the maintainability and readability of the codes. Different programming languages have the different syntax for the comment. In asp.net c#, simply precede the line with two backslashes for single line comments. For multi-line comment blocks, begin the comment with /* and close with */. This is also same with the client script, JavaScript.

For example,

```
/*checks the existence of cookie, if user is not logged in/after logout and shouldn't be allow to
access this page anyhow - send user away */
                    HttpCookie objckLoginId = Request.Cookies["ckloginId"];

                    if (objckLoginId == null)
                    {
                              Response.Redirect("../FailLogin.aspx");
                    }
                    //manipulate the client script checking when button clicks
                    btnSubmit.Attributes.Add("onclick","javascript:return chkFields();");
```

6.4.4 User Control

ASP.NET has a reusable user control, which is a custom control saved in a file with an extension .ascx. User controls are compiled the first time they are accessed and then saved in client memory until any changes are made to the control. The fundamentals of reusability can be applied through the user controls. In the ODRS, the user controls is used to customize the header, footer and easy navigation which contains links and images. User controls can be built once and reused any number of times as required in the same web page, web site or even across multiple web sites. The first and most obvious advantage of user controls is the encapsulation of a logical unit of a web page into reusable controls. The user control can be configured by setting properties and also handling events raised for the user control. After building the desired user control, the user controls can be used by calling the following statements below in the codes.

Table 6.5 User control codes

@Register directive	User Control statement used in the codes
<%@ Register Tagprefix="UserControl" Tagname="MenuLec" Src="..\classes\UserControl\MenuLec.ascx" %>	<USERCONTROL:MenuLec id="MenuLec" runat="Server"></USERCONTROL:MenuLec>
<%@ Register Tagprefix="UserControl" Tagname="Header" Src="..\classes\UserControl\Header.ascx" %>	<USERCONTROL:HEADER id="Header" runat="Server"></USERCONTROL:HEADER>

6.4.5 Configuration Setting

Whenever an application is built, normally the storage of details describing the behaviour and settings for the application is required. This is also known as configuration information. This configuration information includes details such as database connection strings, timeout values and so forth. In ASP.NET, unlike ASP, does not require extensive use of the IIS metabase. Instead, ASP.NET uses an XML-based configuration system. In this system, the ASP.NET application configuration is used. The application configuration information is stored in a file named *web.config*. In the system, the <appSettings/> is used to store application configuration details within the web.config file. For example,

```
<configuration>
<!-- APPLICATION SETTINGS-->
  <appSettings>
    <add key="connectionString" value="data source={LOCAL}; User ID=sa;
Password=; database=ODRS;"/>
    </appSettings>
  <system.web>
```

The use of the key settings simply populates a hashtable that we can access within the application created.

6.4.6 Cascading Style Sheet (CSS)

In coding the web pages in the ODRS system, a W3C technology called Cascading Style Sheet is used to specify the presentation of the elements involved in all the web pages separately from the structure of the document. This separation of structure from presentation simplifies maintaining and modifying the whole layout of the web pages. The elements that can be controlled by CSS include fonts, spacing, margins and whatnots. The structures of the web pages include section header, body text, scroll bar, links and so forth. In coding the CSS in the ODRS, the **external style sheet** is used. The style sheet is coded separately from the web pages. The linking is used to apply the style in the web pages. Different web pages in the ODRS can use the same style sheet and use the specific style by calling the CssClass. When changes are required in the styles, only the single file of the CSS file is needed to be modified in order to make style changes across the entire web sites. The codes used in the external style sheet which is in the *Styles.css* file are shown as following.

```
BODY
{
  Background-color: white;
  Font-Family: Verdana, Helvetica, sans-serif;
  Font-size: .8em;
  Font-weight: normal;
  Letter-spacing: normal;
  Text-transform: none;
  Word-spacing: normal;
  Scrollbar-face-color: #3399cc;
  scrollbar-highlight-color: White;
  scrollbar-shadow-color: #336699;
  scrollbar-arrow-color: #ffffff;
  scrollbar-track-color: #99ccff;
  scrollbar-dark-shadow-color: #A5bbd1;
}

.alertFont {
        font-weight: normal;
        font-size: 12px;
        color: #FF0000;
        font-family: Arial, Helvetica, sans-serif;
        text-align: Justify;
}
```

6.4.7 JavaScript

Mainly, the JavaScript is used as the client scripting to check for the required fields and manipulate the buttons that need the actions which are controlled by JavaScript in coding the ODRS web pages. Error checking is important as well to make sure that the system runs smoothly and without showing unnecessary error messages. The JavaScript is used to check for the errors too. JavaScript codes to check the date format are shown below.

```
function isValidDate(dateStr) {
// Checks for the following valid date formats:
// MM/DD/YY   MM/DD/YYYY   MM-DD-YY   MM-DD-YYYY
// Also separates date into month, day, and year variables

var datePat = /^(\d{1,2})(\/|-)(\d{1,2})\2(\d{2}|\d{4})$/;
// To require a 4 digit year entry, use this line instead:
// var datePat = /^(\d{1,2})(\/|-)(\d{1,2})\2(\d{4})$/;
var matchArray = dateStr.match(datePat); // is the format ok?
if (matchArray == null) {
```

```
            alert("Date is not in a valid format.")
            return false;
    }
    month = matchArray[1]; // parse date into variables
    day = matchArray[3];
    year = matchArray[4];

    if (month < 1 || month > 12) { // check month range
            alert("Month must be between 1 and 12.");
    return false;
    }

    if (day < 1 || day > 31) {
            alert("Day must be between 1 and 31.");
            return false;
    }
    if ((month==4 || month==6 || month==9 || month==11) && day==31) {
            alert("Month "+month+" doesn't have 31 days!")
            return false
    }
    if (month == 2) { // check for february 29th
            var isleap = (year % 4 == 0 && (year % 100 != 0 || year % 400 == 0));
    if (day>29 || (day==29 && !isleap)) {
            alert("February " + year + " doesn't have " + day + " days!");
            return false;
            }
    }
            return true; // date is valid
    }
```

6.4.8 C# Functional Codes

As mentioned before, C# is adopted as the main page language in the code-behind of the ASP.NET web forms. It is used to code all functions and programming logic of the pages. For example, the c# codes shown are the method of retrieving the related data from the ODRS database by using the Stored Procedures and thus bind the data into the Datagrid in order to display it.

```
            //retrieve information from database by openning database connection
            SqlConnection Conn = new
            SqlConnection(ConfigurationSettings.AppSettings["connectionString"]);
            SqlCommand cmdSel = new SqlCommand();
            SqlDataAdapter da = new SqlDataAdapter();

            cmdSel.Connection = Conn;
```

```
        da.SelectCommand = cmdSel;

        //Call the stored procedures
        cmdSel.CommandType = CommandType.StoredProcedure;
        cmdSel.CommandText = "GetStudent";

        DataSet ds = new DataSet();
        da.Fill(ds);

        //Bind the data into datagrid -- dg
        dg.DataSource = ds.Tables[0];
        dg.DataBind();
```

6.4.9 Stored Procedures

Stored procedures are an advanced feature in SQL server that offers the users to create, compile and run SQL statements in the server itself. This is to isolate the business logic from data logic and thus improves the performance of the application. In ODRS, stored procedures are used. Input parameters are taken, operate in a black box fashion and thus return appropriate information. This means that getting information into and out of stored procedures is done through ADO.NET.

```
        CREATE PROCEDURE [ApproveTitle]
        (@intTitleID_1 [int],
        @strStatus_20 [nvarchar](50))

                AS UPDATE [ODRS].[dbo].[TableTitle]
                SET
                [strStatus]         = @strStatus_20
        WHERE
                [intTitleID] = @intTitleID_1
        GO
```

6.5 The Modules Implementation

There are eight main modules in the Online Dissertation Registration System (ODRS).

6.5.1 Authentication Module

- Login – All the users must enter the authenticated login ID and password in order to utilize the system. For each entering of the system, system will check for the validation so as to ensure the users are authorized.
- Logout - When users intend to leave the system, they can click on the logout button available in each page. This is for the user's convenience to logout the system whenever they want.
- Session Timeout – Whenever the session of the system is expired, the users are required to enter the login ID and password again in order to enter the system.
- Account Management – This system has a super administrator who responsible to manage all the accounts of the system. Only those users whose information is provided and entered by the super administrator in order to create accounts, has the account for the system. The super administrator can view, edit and delete the specific accounts. The account is created based on the user type.

6.5.2 Profile Module

- Add/Edit/Delete Users' Profiles – The super administrator will enter the information to the system and thus the account will then be created for the users as authorization. The super administrator can view, edit and delete the users' profiles. The users include the Postgraduate Coordinator, lecturers and students.
- View Profiles – All the users' information entered can be viewed by the users whenever they login with the authenticated login ID and password. They can only view their own particulars.

6.5.3 Documents Downloading Module

- Upload/Edit/Delete documents – This section allows the lecturers to upload the articles, forms, reports and any related documents to the main site together with the documents' title and description.

- Download documents – All the uploaded documents by the lecturer are allowed downloaded by the authorized students. The entered documents' title and description are displayed for the understanding of the students too.

6.5.4 Bulletin Board Module

- Add/Edit/Delete New Bulletins – This section allows the administrator and lecturers to publish the updated news, announcements and notices pertaining to the dissertation. All the entered bulletins will be grouped into the current or archives bulletin. The information can be edited or deleted.
- Set the bulletin to Archives - Due to the status of the bulletin is important during the display to the lecturers and students, administrator and the Postgraduate Coordinator who responsible in managing the bulletin need to always make sure the right status of the bulletin. The current bulletin will be displayed in the home page of the student section in order to get their attention on latest bulletin. Thus, this facility is provided to ease the Administrator and Postgraduate Coordinator to manage the status of the bulletin with multiple selections at one time.
- Bulletin Listing – The entered bulletins will be displayed in the administrator, lecturers and students' section. The current bulletin will be given the priority to be displayed in the home page.

6.5.5 Discussion Record Module

- Add/Edit/Delete Record – This section only allows the lecturers to add the record of face-to-face discussion for those students who have registered the dissertation title. No record can be added to the unregistered students. The records can be chosen to be edited or deleted.
- Record Listing - All the recorded comments will be displayed according to the recorded date.
- Record Report - The printed version of the record in report format is available for lecturers to print out the summary of the discussion as references.

- Register Info – This section will need students to register themselves with some particulars to their supervisors. Thus, the lecturers can view the full registered information of the students when having the discussion.

6.5.6 Event Calendar Module

- Add/Edit/Delete Event – This section is designed particularly for the Postgraduate Coordinator to enter the events related to the dissertation. This event calendar can give a clear time schedule to the lecturers and students about the dissertation.
- Event Listing – All the events can only managed by the Postgraduate Coordinator. Lecturers and Students are only allowed to view the event listing.

6.5.7 Project Title Proposal Module

- Propose/Edit/Delete Title – This section is specifically created for the lecturers to propose the dissertation titles online. Lecturers are allowed to edit and deleted the proposed title before the titles are approved. No changes can be done on the approved titles.
- Approve Title – This part allows the Postgraduate Coordinator to approve the titles that are proposed by the lecturers. All title listings can be sorted according to the choices given. It can be sorted by department, by lecturers, by status or by titles. This provides a convenient mean to read and approve the titles easily.
- Title Listing – The titles listing in the lecturer's management area will be listed only the titles proposed by the specific lecturers, whereas the Postgraduate Coordinator is given the highest priority to view all the proposed titles.

6.5.8 Project Title Assigning Module

- Assign Student – This section allows the lecturers to assign the titles to the students after the titles are approved by the Postgraduate Coordinator. The status of the titles will inform the lecturers whether the titles can be assigned to students. The students with no profile in the system are not allowed to be assigned. One title is only assigned to a student and a student is only allowed to be assigned once.

- Delete Assigning – This section allow the lecturers to delete the assigning of student that had been done previously.
- Title Listings – All the approved titles will be listed out in the students' main site and let the students to view the titles' details. The status of assigning will be stated so as to inform students on the availability of the titles.

6.6 Chapter Summary

Mainly, the implementation of the Online Dissertation Registration System focuses on various aspects which govern the functionality of the system to fulfill the functional and non functional requirements of the system. During the system implementation, system requirements and designs are converted into the programming logical codes. The implementation also involves the setting of the Microsoft .NET Framework, database server and operating system as well. In order to deploy the requirements and design into machine-readable language and thus to produce a usable applications, several software tools are used. These software tools include Microsoft Visual Studio .NET and The Photoshop 7.0.

By referring this chapter, the most important sections in this chapter are about the coding and project development. The prototype development and coding approach are discussed and examples are given based on the ODRS application. All the coding styles adopted in the application of the ODRS are described, these include the include file in ASP.NET, configuration setting, standard naming convention, c# coding and so on. The design of the algorithm and implementation is important to make sure a stable system to be developed and minimized the problems occur in the future enhancement and maintenance.

Chapter 7 System Testing

7.1 Introduction

After coding the program components, we usually examine the codes to spot faults and eliminate them right away. Testing is a process that focused on finding errors and faults. The main objective of testing is to discover different types of errors that exist while executing the system. However, testing cannot show the absence of defects, it can only show that software defects are present. Therefore, a test is considered successful only when a fault is discovered or a failure occurs as a result of our testing procedures. A failure is the observed incorrect behaviour of the software product as a consequence of a fault. Hence, software testing is crucial for software quality assurance and represents the ultimate review of specification, design, and code generation.

7.2 Stages of Testing

Testing is a process of exercising or evaluating a system by manual or automatic means to verify that it has satisfied requirements or to identify differences expected and actual results. A number of different types of testing have to be performed during the implementation, such as unit testing, integration testing, product testing and acceptance testing. In developing the Online Dissertation Registration System (ODRS), 4 stages of the testing are involved in order to uncover different classes of errors in a minimum amount of time with the minimum effort needed. The four stages of testing are unit testing, module testing, integration testing and system testing.

7.2.1 Unit Testing

Unit testing is the first approach in system testing where each program component is tested on its own, isolated from the other components in the system. It verifies that the component functions work properly with the types of input and output expected from studying the component's design. In this stage, testing can be started by examining the program code by reading through it, trying to spot algorithm, data and syntax faults. Test also can be performed by comparing the code with the predefine specifications and design to ensure that all relevant cases have been considered.

Test data for unit testing can be constructed systematically in two basic ways. The first way is known as functional testing or data-driven testing. In this approach, the code itself is ignored; the only information used in drawing up test cases is the specification document. The other extreme is to test to code and ignore the specification document when selecting test cases. It is known as logic-driven testing.

In the Online Dissertation Registration System (ODRS), the following steps are taken in order to carry out the unit testing successfully.

- The codes of the application are reviewed by reading through so as to spot the possible algorithm, data and syntax faults.
- The control objects are traced and tested to ensure its functionality.
- The different data types are examined and thus the data types are used to test the error handling functions in the application.
- Related test cases are designed and developed to ensure the input of the users are suitable and thus provide the desired output.

7.2.1.1 Review the code

Before the source code is deploying, codes are examined by reviewing them line by line to identify any syntax or semantic errors. Faults that have been identified are corrected immediately. Because of the Visual Studio .NET compiler and debugger are provided together with the Microsoft Visual Studio .NET, the detecting faults is faster and more efficient. During the compilation where the solution project is built, the compiler will detect types of the errors and thus appear the messages and line number where the faults placed. With this, the review of the codes and the tracing of the errors are more efficient.

7.2.1.2 Trace and test control objects

Besides tracing and fixing all the syntax and semantic errors, the control objects involved in the application are tested too. Initially, the most important are web forms elements, such as buttons, drop down lists, combo boxes, text boxes and so forth. All these elements are tested in order to ensure the events are well handled and thus the functionality of the elements is ensured. For examples, the *add* (submit) button in the forms are tested its actions and responds whenever it is clicked. The text boxes are tested by entering the different data types to perform the validation and format checking.

7.2.1.3 Examine and test different data type

In the dynamic web application, the data types are the important consideration. Different data types of input produce different output. The input's data types should be adapt to the suitable needed data types. These data types include integers, characters, strings, or date. In order to test the functionality and validity of the input, various data types are take into consideration in the testing process. This can ensure that the input are free from errors and thus produces faults-free output. Besides, NULL value is also taking into account and is used to detect the faults in the application.

7.2.1.4 Choose and implement test cases

Input data and condition need to be considered and chosen whenever a component is needed to be tested. Then, the component is allowed to manipulate that data, and the output is observed. The input is selected so that the output can demonstrate about the behaviour of the codes. Test case is a particular choice of input data to be utilized in the testing process. A test is a finite collection of test cases. In order to perform the testing towards certain components, the test objectives should be determined. Test cases are adopted then and define a test designed to meet the specific objectives. Certain data are needed to be given attention and are purposely chosen to be improper. This is important to check whether the codes can handle the incorrect data and situation.

Examples of ODRS Unit Testing

a. Test case for unit testing on adding new title into TableTitle table in the database and then modify the data as well as delete it from the database.

Table 7.1 Unit Test case on Adding New Title

Step	Test Procedure	Expected Output	Test Result
1	Add a new title into the TableTitle table.	The record is inserted into database according to the appropriate table chosen.	Record was inserted successfully.
2	Click on the [Edit] link in order to modify the information of title.	The information is shown in the related fields and the modified data is updated.	Modified data was updated successfully.
3	Check the checkbox on the item(s) that wish to be deleted, and then click on the [Delete] button to delete the added title.	The checked record(s) is deleted permanently from TableTitle table from database.	Chosen item(s) was deleted successfully.

b. Test case for unit testing on approving proposed title and then the status is saved in the database.

Table 7.2 Unit Test case on Approving Proposed Title

Step	Test Procedure	Expected Output	Test Result
1	Click on the [Approve Proposed Title] link from the menu shown	The proposed titles are listed in well-organised with the status shown.	Record was listed clearly as in the database.
2	Click on the [Approve] button in order to approve the title.	The status of the title is approved.	The status of the title was approved and saved into the database successfully.

7.2.2 Module Testing

When the individual components are working correctly and meet the objectives during the unit tests, these components are combined into a module. A module is a collection of dependent components. Module testing enables each module to be tested independently. This testing will ensure that the modules' calling sequences in the ODRS project is systematic and thus produce the desired output. Hence, in the ODRS, the module testing was carried out after all the relevant components for the specific module were developed and tested. This is crucial to ensure the functionality and the correct flows of the certain modules are carried out consistently and smoothly. Below shown are some test cases used in the module testing.

Examples of ODRS Module Testing

a. Module testing is carried out on the authentication module. Before the system is allowed to be accessed by the authorized users, the entering login identification and password need to be authenticated. This is to ensure the security and verification is performed properly.

Table 7.3 Module Test case on Authentication

Step	Test Procedure	Expected Output	Test Result
1	Type the ODRS URL.	The login page is displayed.	The page was loaded and displayed successfully.
2	Type in the invalid login ID and password to access the system. Then click on [Login] button.	The failed login message is displayed for notification to unauthorized user.	Verification was successfully performed and message was displayed for unauthorized access.
3.	Type in the valid login ID and password to access the system. Then click on [Login] button.	Access granted and redirect to the authentication page of specific section according to the user type.	Verification was successfully performed and authentication page was displayed according to the user type.

b. Module testing is carried on the user management module. Before the each user is given login identification and password, the administrator of the system need to add the new user's information and thus create a new account for the particular user.

Table 7.4 Module Test case on User Management

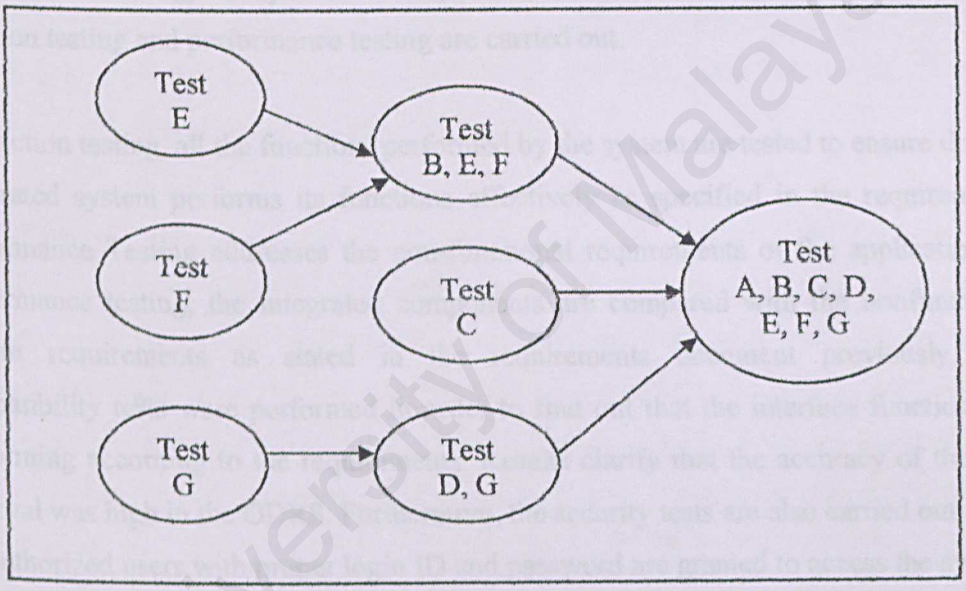
Step	Test Procedure	Expected Output	Test Result
1.	Click on the Add new user button on the menu shown.	The form for adding the new users is shown.	The form was successfully shown.
2.	Enter all the information needed in the fields. After all the fields had been completed, click on [Add] button.	Record will be added into the table according to the appropriate user type and then the form for creating account is displayed.	User was added successfully and the add new account form was displayed.
3.	Enter the login ID and password for the added user. The replicated Login ID is entered.	The failed entrance message is displayed as the LoginID cannot be replicated.	Expected output was accomplished with the failed entrance message shown.
4.	Enter the non-replicated login ID and password for the added user.	The account is created and added into the TableAccount in the database. The account listing is displayed.	The account was created in the database successfully and displayed in the listing.

7.2.3 Integration Testing

After each component and module has been tested, the interaction between these components and modules must be tested again to ensure that the components can be integrated. In other words, integration testing is the process of verifying that the system components work together as described in the system and program design specifications. Integration testing is used to uncover errors associated with interfacing. Interfaces are navigated repeatedly to detect any interface mismatch problem. Several important aspects are checked to ensure that the flow of data in ODRS is well organized and user friendly to all users.

There are several means of integration testing namely top-down approach, bottom-up approach and sandwich approach. In the ODRS, the bottom up approach integration testing is implemented. This approach facilitates that each component at the lowest level of the system hierarchy is tested individually first. Then, the next components to be tested are those that call the previously tested ones. This approach is followed repeatedly until all components are included in the testing. For example, the bottom up integration testing flows can be described as the figure shown.

Figure 7.1 Bottom-up Approach in Integration Testing



In the ODRS, the Bulletin Board Module, Project Title Proposal/Approval Module and Event Calendar Module are tested individually and thus tested for integration in the Postgraduate Coordinator Management Section. This is same with the Administrator Management Section, the Authentication module; Profile module and Bulletin module are tested individually before the integration. In the Lecturer Management Section, the Title Proposal Module, Title Assigning Module, Discussion Record Module are tested individually before the integration too. After the three sections has been undergone the integration testing, the student section is thus be tested.

7.2.4 System Testing

Final testing procedure done is system testing. However, testing the system entirely is very different from previous unit testing and integration testing. The sub-systems are integrated to make the entire system. A system testing is a series of different tests designed to fully exercise the system to uncover its limitation and to measure its capabilities. Its objective is to test an integrated system and verify that it meets the specified requirements. As this point, the behavior of the tests will define the quality of the systems. There are several steps in testing a system such as function testing, performance testing, acceptance testing and installation testing. In the ODRS, the function testing and performance testing are carried out.

In function testing, all the functions performed by the system are tested to ensure that the integrated system performs its functions effectively as specified in the requirements. Performance Testing addresses the non-functional requirements of the application. In performance testing, the integrated components are compared with the nonfunctional system requirements as stated in the requirements document previously. The compatibility tests were performed in order to find out that the interface functions are performing according to the requirements. Results clarify that the accuracy of the date retrieval was high in the ODRS. Furthermore, the security tests are also carried out. Only the authorized users with proper login ID and password are granted to access the system. These tests can ensure the ODRS fulfill the security requirements.

7.3 Chapter Summary

Testing is one of the important steps in developing a system. Precision and accuracy of output data is considered during this process. Unit testing, integration testing and system testing has been carried out for the ODRS. These testing approaches lead to delivering a quality system to users. The objective of a system will only achieve after all the thorough testing done by different user with different aspects.

Chapter 8 System Evaluation & Conclusion

8.1 Introduction

The last phase before the delivery of the system to the end users is the system evaluation. This is the ultimate phase of developing and also the significant phase. System evaluation is implemented by more than simply comparing the information obtained with the information which is expected. It normally related to user environment, attitudes, information priorities and several others concerns that are to be taken into consideration before effectiveness can be concluded.

In the stage of evaluation in the ODRS, problems related to the software and hardware were reviewed and observed. Furthermore, the suggestions and comments for the improvements and enhancements are also reviewed and taken into consideration. Changes are made, whereby the system's strengths and limitations were evaluated by various parties and users. Thus, the future enhancements can base on the evaluation results. Evaluation is a process that occurs continuously, drawing on a variety of sources and information.

8.2 Problems and Solutions

Throughout the development of the Online Dissertation Registration System (ODRS), some of the problems have been encountered.

8.2.1 Limited Information from Interviewee

Interview is one of the information gathering techniques that was used to gather the users' requirements. However, the formal interview is failed to be conducted; only the informal interview is conducted to gather some fundamental information. This is because the administrator who understands all the related matters of the dissertation registration is not available to allow me to conduct any formal interview. However, some other staffs and the simple official web site had provided me some needed information. Because of this, it is hard to view the entire system in fulfilling the administrator's requirements. In addition, in order to produce a feasible system, many similar systems had been studied and reviewed. Besides, observation and data collection is also one of the techniques used to gather the needed information. The data used in the system are obtained from the web sites and notice boards during the previous semester.

8.2.2 Inexperienced In the Chosen Software Tools

There are many types of web based system development tools available nowadays. Choosing the right tools and languages are important because the appropriate tools and languages would help to develop the system in a more efficient way. However, the solution taken up is the consideration and review on the benefits as well as the trends of the technology throughout the analysis phase. Furthermore, the discussions and advices from supervisors, lecturers and friends had really assisted me to choose the suitable development tools. Finally, the Microsoft Visual .NET with its great IDE has become the development of the Online Dissertation Registration System (ODRS).

8.2.3 Inexperienced In the Chosen Programming Languages

Since there was no prior knowledge of programming in ASP.NET, there was an uncertainty and doubt on how to develop a web based system by using ASP.NET with code behind page language C#. The new programming languages and concepts were never been taught. Hence, in order to implement such an application, it requires a fair grasp of the languages. Due to time constraint, the learning and developing processes were carried out in parallel. Therefore, the parallel learning and developing has slowed the progress of the development sometimes. Although, there are difficulties arose in the early stage of the development, ASP.NET is still a powerful technology to build the web-based application like ODRS. Furthermore, the object-oriented concept used in the ASP.NET had given a good understanding in the concept of object-oriented practically.

Problems are solved gradually by referring the reference books from UM Library, friends and in the market. In addition, resources from internet had also assisted me a lot in solving errors occured during the development and also building the structures of the codes. Discussion with friends who are using the same technology was hold occasionally in order to get more understanding on the technology.

8.2.4 Lack of Knowledge and Inexperienced in Setting up IIS and Database Server

Building the dynamic web-based application, the web server and database server are the important elements in ensuring the connectivity and functionality of the application. In ODRS, using the IIS as web server and MS SQL Server 2000 as the database server requires much understandings and experiences of its features as well as concepts. The setting and configuration of IIS is different from the non web-based application. Furthermore, the way of administering and manipulating database server are far different from the ordinary stand alone database systems. The wrong and improper configuration could cause the system malfunction. Therefore, in order to solve all these problems, much reading on documentation and discussion with some experienced friends had been done. Finally, the setting up and configuration can be carried out successfully in the PC at home and at faculty.

8.2.5 Lack of Resources

Due to the lack of resources such as hardware, time and people, the system could not be tested entirely on a wider scale. The developed ODRS system did not manage to be evaluated by the end users (the administrator from office) also due to insufficient time.

Therefore, valuable feedbacks were not received although the testing has been done thoroughly by people from IT and non-IT backgrounds. However, the feedbacks and suggestions from supervisors have been taking into consideration to improve the system so as to provide the similar functionalities as in the real life dissertation registration system. Besides, the lack of time is also one of the factors that affects the system could not be developed with more functionalities and thus less flexibility is brought to the users in managing the whole complete process of the dissertation registration.

8.3 System Strengths

8.3.1 Consistent and Easy-to-use Interface with Good Layout of the Website

ODRS is specifically designed based on the criteria of consistent, easy-to-use, user-friendly and interesting GUI design. The interface is designed in such a way that is suitable and easy to navigate by all the parties involved in the dissertation registration. The principles and techniques of good GUI design are applied when designing the interfaces. Some interesting images and icons are placed in order to make the web pages attractive yet simple and well-organized to the users. Besides that, the clear and standardization of the menu and layout in all the sections may help the novice users to learn and use the system effectively in the minimum amount of time.

8.3.2 Organized and Systematic With Clear Division of the Sections

ODRS is designed according to the main 4 parties who are involved in the dissertation registration. Each party has its own section and they do not have the authorization to access to other parties' section. The authorization to the specific section is decided by the super administrator of the system. Furthermore, each sub section with different functionalities in the section is divided in well-organized way in order to not confuse the users to navigate through the section.

8.3.3 Wide Accessibility

The ODRS is a web-based system. Thus, it provides wide accessibility to all the users whenever and wherever the users are, without any geographical boundary. All users only need the internet connection with client-side software such as web browser installation in the PC in order to access to the ODRS. Once the super administrator had registered the users into the account according to the user type, the authorized users can start accessing the system.

8.3.4 Support High Volume of Data and Database Backup

ODRS is deployed using the latest database management system that is Microsoft SQL Server 2000, which is one of the most powerful database management systems in the market. This will makes ODRS equipped to cope with large amount of data in the future. Besides that, with the capability of the Microsoft SQL Server 2000, it provides the functionality of database backup. All the records and information in the table and database can be restored from the backup files once the system is corrupted. This has increased the powerful and performance of system.

8.3.5 Confidentially and Integrity of Information with Security Protection

ODRS is a password-protected system, which prevent unauthorized users from accessing and changing the system data stored in the database. With this password-protected, the system protects also confidentially and integrity of information by providing only restricted access to certain functionalities in ODRS to authorized users. Each type of users has different level of access control to the system. For example, add, edit and delete new bulletin can only be accessed by administrator and the Postgraduate Coordinator but not students and lecturers.

8.3.6 Data Validation and Error Checking

In all the forms involved in the ODRS, the data validation is performed in each required and specific fields before the records are inserted into the database. The data validation is coded by using client script, JavaScript. The objective of this data validation is to ensure the invalid data are not inserted into the database which can cause the system to errors. All the fields in the forms are checked for null value, invalid data type, invalid format and also invalid information entered. With this feature, the occurrence of errors can be reduced and then the error messages can display for users without the sudden termination of the system.

8.3.7 Information Listing

The ODRS is mainly involving much data inserted by the users into forms. Thus, all the information listing is bound systematically into data grids has increased the maintainability and readability of the information in the system. Furthermore, hyperlinks to the detailed information are also provided for particular titles or fields. With this data binding and linking, all the information can be referred easily and clearly by users without any troublesome.

8.4 System Limitations

There are some limitations in the ODRS due to time constraints, facilities constraints, limitation of the programming skills and also the project boundaries.

8.4.1 No Encryption and Decryption

Due to the time constraints and inexperienced in the encryption and decryption, currently the ODRS provides no password encryption and decryption to enhance the security. Consequently, unauthorized access by certain parties such as hackers may be occurred. Hackers may access the data easily if they successfully hack into the system.

8.4.2 Browser Constraints

The ODRS is developed under the consideration of the Microsoft Windows platform. Furthermore, majority of the codes and scripts are written in C#, which is best viewed and supported only by using Microsoft Internet Explorer 5.5 or above. In addition, the JavaScript is written in the system which may cause some browsers cannot interpreter and read the JavaScript. Therefore ODRS is unable to be used with certain type of browsers which may cause the system can not be worked effectively and efficiently.

8.4.3 Lack of Utilities

All fundamental and critical functionalities that are needed by different parties involved in the registration can be obtained in ODRS. However, ODRS is still lacking of some utilities such as email server, chat room, forum, document or file submission and online help in order to make ODRS more interactive and comprehensive. These utilities can provide better coverage of all aspects of the system and thus increase the system usability and feasibility to the users.

8.5 Future Enhancements

Due to the limitations are appeared in the project, there are some suggestions arise for the enhancements and improvement to the current version of the ODRS. Below listed are the suggestions given.

8.5.1 Encryption and Decryption

Password encryption and decryption can be implemented to enhance the security of the ODRS. Thus, users' password is always protected and security is ensured. Once users had been registered as authorized users, their password will be encrypted and stored in the database. The password will only be decrypted when the users retrieve. By using this mean, hackers will not easily hack into the system and access or modify the actual data.

8.5.2 Browser Consideration

As mentioned before, the ODRS needs the Microsoft Internet Explorer for best viewing and accessing. For further enhancements and improvements on the ODRS, it can be coded in order to consider the different browsers installed by the users. Thus, the ability of browsers can be extended and the ODRS can be worked smoothly without the constraints of browsers.

8.5.3 Forum or Chat Room Utilities

Due to the involvement of various parties in the ODRS, the forum or chat room utility can be provided in ODRS for interactive messaging. Users can communicate with each other that are currently online through the chat room; whereas the messages or enquiries can be posted to the forum. With these two facilities provided, users can share their views, opinions, experiences and knowledge without any troublesome. Besides that, the experienced people like lecturers can respond to the messages or enquires in the shortest time or in advance related to the students' queries. Therefore, the forum and chat room can provide an alternative yet effective way for the interaction of the users.

8.5.4 E-mail Server and Appointment Making

Setting up an e-mail server and mailing option within the ODRS will open up lots of options for all the users. They will be able to contact each other easily through electronic mailing, especially for the students to contact their respective lecturers. This really provides a faster yet effective mailing option to the users. With this mailing facility, the appointment can be made easily with the lecturers by the students. Furthermore, the lecturers' schedules can be set up and sent to the respective students to ease the appointment making between students and lecturers by adding the appointment making section. This is really a time saving way.

8.5.5 Students' Registration and Report Submission

ODRS can further its functionalities in allowing the students to register the titles themselves after obtaining the approval and satisfaction from the lecturers on the proposal of the desired titles. Furthermore, ODRS can provide the capability for students to submit their proposal on the desired titles where lecturers can download and read the proposal. After the online approval is done by lecturers, the notification mail can be sent to the students. Besides, ODRS can provide the online report submission throughout the period of the dissertation preparation by the students to the lecturers. Therefore, the paperless environment can be built.

8.5.6 Report Generating

Due to the time constraint, there are only discussion record reports that are available in the printed version. Therefore, the others information that need to be generated into reports can be enhanced so that users can view and print various kinds of reports according to their needs in appropriate printed format.

8.6 Knowledge and Experience Gained

From the planning, analysis, design, implementation to the testing process of the Online Dissertation Registration System, much knowledge and valuable experience had been gained.

8.6.1 Exposure to the Real System Design and Development Environment

Developing the ODRS has really given me a great opportunity to exposure myself in designing and developing the real system from start till the end. Throughout the whole process of the developing, I learnt how to manage a project's in time and resources. Furthermore, I understand and implement the concept on how to integrate and develop the system with fully utilize various technologies. With this, I greatly improve myself.

8.6.2 The Importance of the Requirements Elicitation and System Analysis

Software Requirements elicitation is the most important phase in determining whether the system fulfill the users' needs. If this phase is wrong defined, it will later the progress of development and it may cause the failure of the system eventually. Thus, with the complete and thorough requirements elicitation and system analysis, the system can only fulfill the requirements and achieve the objectives of the system.

8.6.3 The Learning of the Development Tools and Programming Language

In developing this ODRS, Microsoft Visual Studio .NET, Microsoft SQL Server 2000 and Photoshop are used to code and design the web pages. Microsoft Visual Stidio.NET is a very suitable tool for web-based programming as it provides a graphical user interface platform, while Photoshop is a great yet efficient tool in producing various attractive images and icons. Microsoft SQL Server 2000 is also a powerful Database Management System which can support high volume of data. Together with the language used - ASP.NET and C#, all the codes and images are applied practically and thus the system can be built successfully. Therefore, it really improves my understanding and application of the languages together with the software tools in building a feasible system.

8.7 Conclusion

There were certain expectations on what would be achieved in this final stage. As stated in the objective of the Online Dissertation Registration System, it is expected that the organized and systematic online system with the good user interface can convert the current manual registration process into a computerized web-based information system and thus create the paperless environment by integrating the tasks involved. So, finally, the developed Online Dissertation Registration System had successfully created a system with all functionalities stated as well as all objectives achieved and thus assists all the related users in carrying the tasks of registration. Overall, the Online Dissertation Registration System has achieved its goals and fulfilled the functional requirements. The feasibility of the system depends on how much the parties involved will benefit from its implementation.

This project has greatly provided me a golden opportunity to build a full web application from scratch. Theories and knowledge gained throughout the studies of the Computer Science courses like System Analysis and Design, Software Engineering, Software Requirement Engineering, Database Management and so forth were literally put into practice and application. A lot of precious knowledge on web based programming was gained especially in the programming skills. These include the configuration and management of Windows XP and IIS, programming knowledge in ASP.NET, C#, JavaScript as well as the techniques and concepts in implementing Database Server. This project has been a great useful experience to me which includes the exposure of the research work and application of theories.

In a nutshell, there are still much more improvements in this developed system in turning it into a feasible and useable system in the real life. Enhancements and improvements can be made with more features and functions for newer version. Great efforts and research will still be done in improving myself in developing the real time system or application in the future.

REFERENCES

[Cheevers, 2002] Cheevers, S. (2002). *Oracle Database Summary.* Retrieved August 2, 2004, from http://www.oracle.com/ip/develop/ids/jdevdocs/oracle9i_Database_summary.pdf

[Chigrik, 2003] Chigrik, A. (2003). *The comparison of SQL Server 2000 with MySQL v4.1.* Retrieved August 3, 2004, from http://databasejournal.com/features/mssql/article.php/3087841

[DaySite Web Hosting, 2003] DaySite Web Hosting. (2003). *How JSP compares with ASP.* Retrieved 1 August 2004, from http://www.daysite.net/programming/jsp.htm

[Deitel et al, 2002] Deitel, H.M., Listfield, J., Nieto, T.R. & Yaeger, C. (2002). *C# How to Program.* New Jersey, Prentice Hall.

[Flynn et al, 2000] Flynn, I.M. & McHoes, A.M. (2000) *Understanding Operating System.* Third Edition. Pacific Grove, CA, Thomson Learning.

[Fowler et al, 1998] Fowler, H.W., Fowler, F.G. & Thompson, D. (Ed.). (1998). *The Concise Oxford Dictionary Ninth Edition.* New York, Clarendon Press.

[Hovath, 1998] Hovath, D.B. (1998). *UNIX for mainframer* (2nd ed.). Upper Saddle River, NJ: Prentice Hall PTR

[Kan, 1995] Kan, Stephen H. (1995) *Metrics and Models in Software Quality Engineering.* Massachusetts, Addison-Wesley.

[Karan, 2003] Karan Dass (2003). *JSP vs ASP.* Retrieved August 1, 2004, from http://www.indiawebdevelopers.com/technology/java/jspvsasp.asp

[MySQL Database Server, 2004] *MySQL Database Server.* Retrieved August 2, 2004, from http://www.mysql.com

[Pressman, 1997] Pressman, R.S. (1997). *Software Engineering: Ad Practitioner's Approach. (Fourth Edition).* New York, McGraw Hill, Inc.

[Prototyping Model, 2004] *Prototyping Model..* Retrieved August 20, 2004, from http://www.ctg.albany.edu/publications/reports/survey_of_sysdev?chapter=7

[Ramesh, 2003] Ramesh, B. (2003) *What is ASP.NET?* Retrieved August 1, 2004, from http://www.indiawebdevelopers.com/technology/java/India_Web Developers_com_aspnet1.htm

[Sadoski, 2004] Sadoski, D, GTE. (2004). *Client/Server Software Architectures - An Overview*. Retrieved July 20, 2004, from http://www.sei.cmu.edu/str/descriptions/clientserver.html

[Sadoski, 2004] Sadoski, D, GTE. (2004). *Two Tier Software Architectures*. Retrieved July 20, 2004, from http://www.sei.cmu.edu/str/descriptions/twotier.html

[Sadoski, 2004] Sadoski, D, GTE. (2004). *Three Tier Software Architectures*. Retrieved July 20, 2004, from http://www.sei.cmu.edu/str/descriptions/threetier.html

[Schach,2005] R. Schach, R. (2005). *Object-oriented & Classical Software Engineering.* Sixth Edition. New York, Mc-Graw Hill Companies.

[Stair et al, 2001] M. Stair, R. & W. Reynolds, G. (2001) *Principles of Information Systems – A Managerial Approach.* Fifth Edition. Australia, Course Technology Thomson Learning.

[Sommerville ,2001] Sommerville, I.(2001). *Software Engineering.(Sixth Edition)*. New York, Addison Wesley.

[System Requirements .NET Framework 1.1 SDK , 2004] *System Requirements .NET Framework 1.1 SDK.* (2004). Retrieved September 10, 2004, from http://msdn.microsoft.com/netframework/technologyinfo/sysreqs/default.aspx

[System Requirements for Microsoft SQL Server 2000] *System Requirements for Microsoft SQL Server 2000.* (2004). Retrieved September 10, 2004, from http://www.microsoft.com/sql/evaluation/sysreqs/2000/default.aspx

[System Requirements for Visual Studio .NET, 2004] *System Requirements for Visual Studio .NET 2003.* (2004). Retrieved September 10, 2004, from http://msdn.microsoft.com/vstudio/productinfo/sysreqs/default.aspx

[The Waterfall Model, 2004] *The Waterfall Model.* Retrieved August 20, 2004, from http://www.ctg.albany.edu/publications/reports/survey_of_sysdev?chapter=5

[Whitten et al, 2004] L. Whitten, J., D.Bentley, L. & C.dittman, K. (2004) *System Analysis and Design Methods*. Sixth Edition. New York, Mc-Graw-Hill Company.

Bibliography

Abrar Hussain. (2001). Developing a Record Navigation Form with Unbound Controls. Retrieved November 3, 2004, from http://www.c-sharpcorner.com/Database/NavigationInUnboundControlsAH002.asp

Andrei Pociu. (2004). Creating and using stored procedures. Retrieved January 3, 2004, from http://www.geekpedia.com/tutorial92_Creating-and-using-stored-procedures.html

Ansil. (2004). Session management options in ASP.NET. Retrieved December 23, 2004, from http://www.codeproject.com/aspnet/ASPNETSession.asp

ASPNet101.com (2004). Searching, a DataSet, Paging & Parameters (SQL Server). Retrieved January 15, 2004, from http://aspexpress.com/aspnet101/aspnetcode.aspx?recent=true

Atal Bihari Upadhyay. (2004). Paging in asp.net. Retrieved December 30, 2004, from http://www.dotnetspider.com/technology/kb/aspsamples.aspx?PageNumber=2

Bill Gauvey. (n.d.) Building an ASP .NET User Control. Retrieved January 3, 2004, from http://www.lrdnug.org/html/articles/article0001/default.htm

Budi Kurniawan. (2002). ASP.NET File Uploading. Retrieved by December 30, 2004, from http://www.ondotnet.com/pub/a/dotnet/2002/04/01/asp.html

Christoph Wille. (2000). Uploading Files with ASP.NET. Retrieved December 30, 2004, from http://www.aspheute.com/english/20000802.asp

Code Project. (1999-2005). http://www.codeproject.com/

Deitel & Deitel. (2004). C# How to Program. Pearson Education International, New Jersey.

Deitel & Deitel. (2004). Internet & World Wide Web (Third Edition). Pearson Education International, New Jersey.

Dimitrios Markatos. (2005). Custom ASP.NET Datagrid Paging With Exact Count - Accessing our Data. Retrieved November 3, 2004, from http://www.4guysfromrolla.com/webtech/082901-1.shtml

Dimitrios Markatos (2005). Multiple Checkbox Items In a DataGrid. Retrieved October 21, 2004, from http://www.developerfusion.com/show/4632/

Dipal Choksi. (n.d.) Advanced ASP.NET: Validation Controls in Web Forms. Retrieved December 30, 2004, from http://www.aspdotnetheaven.com/Tutorials/ValidationControls.asp

Dipal Choksi. (2001). User Control and Reusability. Retrieved February 3, 2004, from http://www.c-sharpcorner.com/asp/Code/UserControlReusabilityPDPL.asp

Dotnetindex.com. (2003). Executing Transact-SQL SELECT statement using SQLCommand. Retrieved January 3, 2004, from http://www.dotnetindex.com/articles/more6.asp

Jayram Tallamraju. (n.d.). Multi-Column Sort Using the ASP.NET DataGrid. Retrieved October 20, 2004, from http://www.asp101.com/articles/jayram/multicolumnsort/default.asp

Jeff Louie. (2004). Web Forms DataGrid and DataSet Programming - Updating a Row. Retrieved December 12, 2004, from http://www.developerfusion.co.uk/show/3801/6/

Konstantin Vasserman. (2002). File Upload with ASP.NET. Retrieved January 3, 2004, from http://www.codeproject.com/aspnet/fileupload.asp

Matt Rutledge. (n.d.). Using ASP.NET to Make a Secure Site and Move Login Processing Out of the Page. Retrieved December 6, 2004, from http://www.asp101.com/articles/matt/securesite/default.asp

Microsoft Corporation. (2002). Authentication and Authorization Concepts. Retrieved October 21, 2004, from http://msdn.microsoft.com/library/en-us/overview/htm/cms_ao_concepts_dbzp.asp

Microfsoft Corporation. (2004). How To Implement Forms-Based Authentication in Your ASP.NET Application by Using C# .NET. Retrieved December 30, 2004, from http://support.microsoft.com/default.aspxscid=kb;en-us;Q301240

Microsoft Corporation. (2004). MSDN Library: .NET Framework Class Library, SqlConnection Class. Retrieved January 3, 2004, from http://msdn.microsoft.com/library/en-us/cpref/html/frlrfSystemDataSqlClientSqlConnectionClassTopic.asp

Mohammad. (2003). Datagrid basic operations on ASP.NET/C# Retrieved by February 2, 2004, from http://www.codersource.net/asp_net_datagrid_part1_azam.html

Mohammad Azam. (2004). Dynamically Loading User Control on a Webform using PlaceHolder control. Retrieved December 30, 2004, from http://www.thecodeproject.com/aspnet/LoadingUSerControl.asp

Mysorian. (2004). ADO.NET 101: SQL Connection. Retrieved December 31, 2004, from http://www.codeproject.com/aspnet/SQLConnect.asp

Narendra. (2003). Creating a Custom ADO.NET DataAdapter. Retrieved October 2, 2004 from http://www.csharpfriends.com/Articles/getArticle.aspx?articleID=103

Oleg Gdalevich. (2002). Parameters and Return Values. Retrieved January 3, 2004, from http://www.vbip.com/books/1861003064/chapter_3064_03.asp

Ramakrishna jillellamudi. (2003). Redirecting User to Login Page After Session Timeouts. Retrieved December 12, 2004, from http://www.dnzone.com/go?565

Robet Vieira. (1999). Professional SQL Server 7.0. Programming. Wrox Press Ltd, Birmingham.

Server Intellect. (2004). ASP.NET Tutorial: Adding records to a database using ADO.NET, SQL Server 2000 and Visual Basic.NET. Retrieved December 3, 2004, from http://www.easerve.com/developer/tutorials/asp-net-tutorials-adding-records-database.aspx

Server Intellect. (2004). ASP.NET Tutorial: Working with Dates and Time in ASP.NET using Visual Basic.NET. Retrieved December 30, 2004, from http://easerve.com/developer/tutorials/asp-net-tutorials-dates.aspx

Sushila S. Patel. (2003). Change the Column Value Dynamically and Display it in a DataGrid. Retrieved October 3, 2004, from http://www.c-sharpcorner.com/Code/2003/June/ChangingColumnDyna.asp

Sushila D. Patel. (2003). How to Add a Counter Column to the DataGrid? Retrieved November 3, 2004, from http://www.c-sharpcorner.com/members/SushilaBowalekar.asp

Sushila D. Patel. (2003). How to use HyperLink Column of DataGrid? Retrieved October 20, 2004, from http://www.c-sharpcorner.com/Code/2003/June/DataGridHyperLinkColumn.asp

Sushila D. Patel. (2003). Using Calendar Control in ASP.NET. Retrieved February 15, 2004, from http://www.c-sharpcorner.com/members/SushilaBowalekar.asp

Thomas CAnnolly & Carolyn Begg. (2002). Database System - A Practical Approach to Design, Implementation, and Management. Addison-Wesley, New York.

Thomas Powell & Fritz Schneider. (2001). JavaScript: The Complete Reference. Osborne/McGraw-Hill, New York

Tushar Ameta. (2002). Playing with DataGrid Control. Retrieved November 3, 2004, from http://www.c-sharpcorner.com/Database/PlayingWithDataGridTA.asp

Appendix A – User Manual

TABLE OF CONTENTS

Page

Chapter 1 Introduction

1.1 About This Manual 1
1.2 Conventions 2

Chapter 2 Hardware and Software Requirements

2.1 Hardware Requirements 2
2.2 Software Requirements 3

Chapter 3 Installation Process

3.1 Setting Up ODRS Virtual Directory 3
3.2 Setting Up Database 6
3.3 Modifying the Connection Setting 7

Chapter 4 Getting Started

4.1 Access Authentication 9
4.1.1 Login / Logout.. 9

4.2 Administrator Management Area.. 11
4.2.1 User Management – Add New User.. 12
4.2.2 User Management – Edit Student/Lecturer Profile.. 13
4.2.3 User Management – Delete Student/Lecturer Profile.. 15
4.2.4 User Management – View Profile.. 15
4.2.5 Account Management – Edit Account.. 16
4.2.6 Account Management – Delete Account.. 17
4.2.7 Bulletin Management – Add New Bulletin.. 17
4.2.8 Bulletin Management – Edit Bulletin.. 18
4.2.9 Bulletin Management – Set bulletin(s) to Archives.. 19
4.2.10 Bulletin Management – Delete Bulletin.. 19
4.2.11 Bulletin Management – View bulletin.. 20

4.3 Lecturer Management Area.. 20
4.3.1 Title Proposal – Propose Title.. 21
4.3.2 Title Proposal – Edit Proposed Title.. 21
4.3.3 Title Proposal – Delete Proposed Title.. 22
4.3.4 Title Proposal – View Title.. 23
4.3.5 Student Registration – Assign Student.. 23
4.3.6 Student Registration – Delete Assigning.. 24
4.3.7 Student Registration – View Assigning.. 25
4.3.8 Document Uploading – Upload Document.. 25
4.3.9 Document Uploading – Edit Document's detail.. 26

4.3.10 Document Uploading – Delete Document's detail.................. 27
4.3.11 Document Uploading – Download and View Document's details.. 27
4.3.12 Discussion Record – Add New Record.................................. 28
4.3.13 Discussion Record – Edit Record.. 30
4.3.14 Discussion Record – Delete Record..................................... 31
4.3.15 Discussion Record – View and Print Record.......................... 31
4.3.16 View Student Info... 32
4.3.17 View Calendar Event.. 33

4.4 The Postgraduate Coordinator Management Area
4.4.1 Title Proposal – Approve Title.. 33
4.4.2 Bulletin Management.. 35
4.4.3 Calendar Management – Add New Event.............................. 35
4.4.4 Calendar Management – Edit Event..................................... 36
4.4.5 Calendar Management – Delete Event.................................. 36
4.4.6 Calendar Management – View Event.................................... 37

4.5 Student Main Site 37
4.5.1 Title.. 37
4.5.2 Bulletin.. 38
4.5.3 Download.. 38
4.5.4 Calendar.. 38
4.5.5 Register Info and My Profile.. 38
4.5.6 Contact.. 39

User Manual for Online Dissertation Registration System (ODRS)

Chapter 1 Introduction

Online Dissertation Registration System (ODRS) is an online information system that enables lecturers, students, postgraduate coordinator, and the system administrator to log in using the services and the functions provided in order to complete the registration process and to manage the related events. These events include user management, account management, title approval, title assigning, bulletin management, calendar event management, document management and discussion record management. This Online Dissertation Registration System (ODRS) is a necessary tool for every institute nowadays as the manual registration is no longer the efficient way to manage the registration process. With the online system, the registration dissertation process can be easier, systematic and efficient.

1.1 About this Manual

This user manual will guide you through all the functions available in the system. The step-by-steps for each function are described. It includes,

- System Overview – Hardware & Software Requirements
- Installation Process
- Administrator Management Area Overview
- Lecturers Management Area Overview
- The Postgraduate Coordinator Management Area Overview
- Students Main Site Overview

1.2 Conventions

In order to assist you to interpret and navigate the information easily, this user manual used the following consistent typographic and conventions to explain the commands and controls used in the system.

- [Button] indicates the button in the page
- ***Menu/Option*** indicates the menu shown in the page
- Hyperlink indicated the hyperlink in the system

Chapter 2 Hardware and Software Requirements

2.1 Hardware Requirements

Hardware requirements for the server computer are listed as following.

Type of Hardware	Requirements
Processor	450-megahertz (MHz) Pentium II-class processor 600-MHz Pentium III-class processor recommended
RAM	192MB or greater (256 MB recommended)
Hard disk	900MB of system drive; 3.3GB of installation drive Additional 1.9 GB for optional MSDN Library documentation
Drive	CD-ROM or DVD-ROM drive
Video	1024x768 or higher resolution display with 256 colour
Others	Keyboard and Microsoft Mouse or compatible pointing device, Modem 56K

2.2 Software Requirements

The software requirements for the server computer are listed as following.

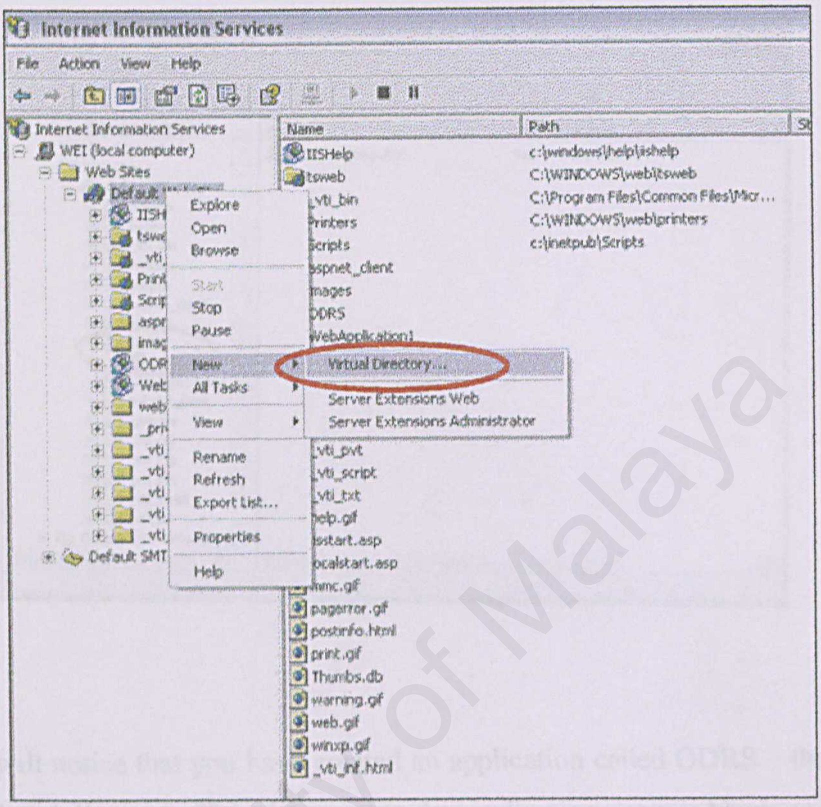
Type of Software	Requirements
Operating System	Windows XP Professional
Browser	Internet Explorer 5.01 or later
Web server	Microsoft Internet Information Server(IIS)
Database Server	Microsoft SQL Server 2000
Authoring Tools	.NET Framework 1.1 SDK

Chapter 3 Installation Process

3.1 Setting up ODRS Virtual Directory

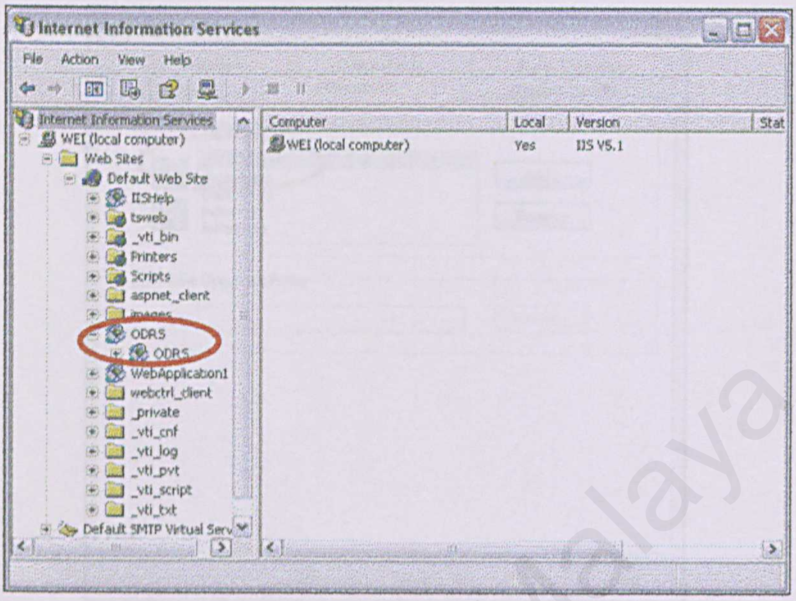
1. Firstly, start the ***Windows Explorer*** and a new physical directory named ODRS is created under the ***\Inetpub\wwwroot*** directory which is created by IIS on your hard drive.
2. Copy whole ODRS files to the directory above.
3. Then, go to ***Start > Control Panel > Administrative*** Tools, then click and enter the ***Internet Information Service***.
4. Expand ***<domain name>*** by clicking +. Right click on ***Default Web Site*** and choose *New* > ***Virtual Directory...*** as shown in the Figure 3.1. You will see the splash screen of the Virtual Directory Creation Wizards. Click on [Next].

Figure 3.1 The Creation of the Virtual Directory in IIS



1. Type ODRS in the Alias text box; then click on [Next].
2. On the next screen, click [Browse] button and select the directory ***Inetpub/wwwroot/ODRS/ODRS*** that you have created in step 1. Then, click [Next].
3. Make sure that the ***Read*** and ***Run*** scripts checkboxes are checked, and that the Execute checkbox is empty. Click on [Next] and in then, click on [Finish].
4. The ODRS virtual directory will appear on the tree in the IIS Administration Windows as shown in the Figure 3.2.
5. Right click on the ODRS Virtual Directory and select ***Properties***.

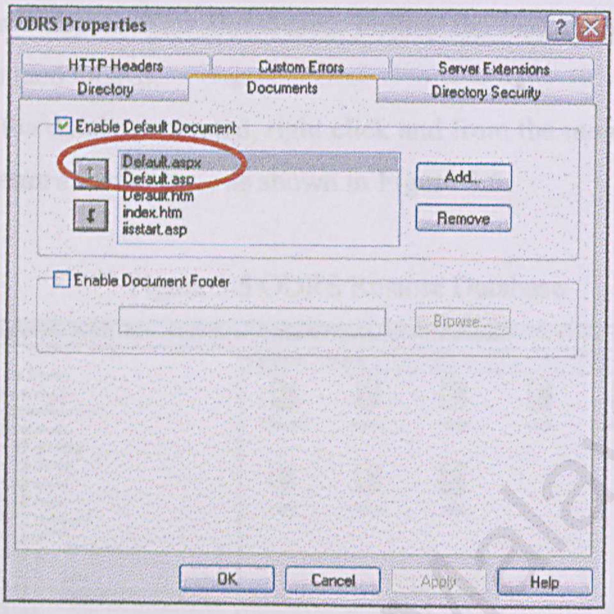
Figure 3.2 ODRS Virtual Directory created



Computer	Local	Version	Stat
WEI (local computer)	Yes	IIS V5.1	

1. You will notice that you have created an application called ODRS – the same as the virtual directory. If the box next to the application name is blank, you need to hit the [Create] which will turn into [Remove]. This is to enable us to use Global.asa in the scripts.
2. Check only the ***Read*** checkbox.
3. Select the ***Document*** tab, click on ***Default.aspx*** and move the file to the highest in the list box as shown in figure 3.3. Then click [OK].

Figure 3.3 ODRS > Properties > Document Tab

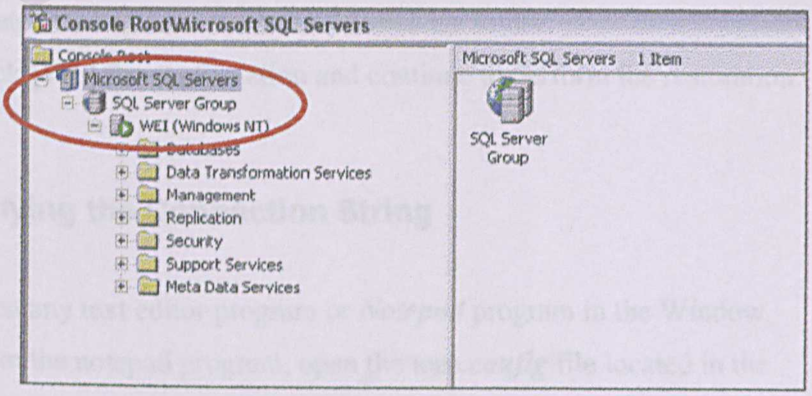


1. Exit IIS.

3.2 Setting up Database

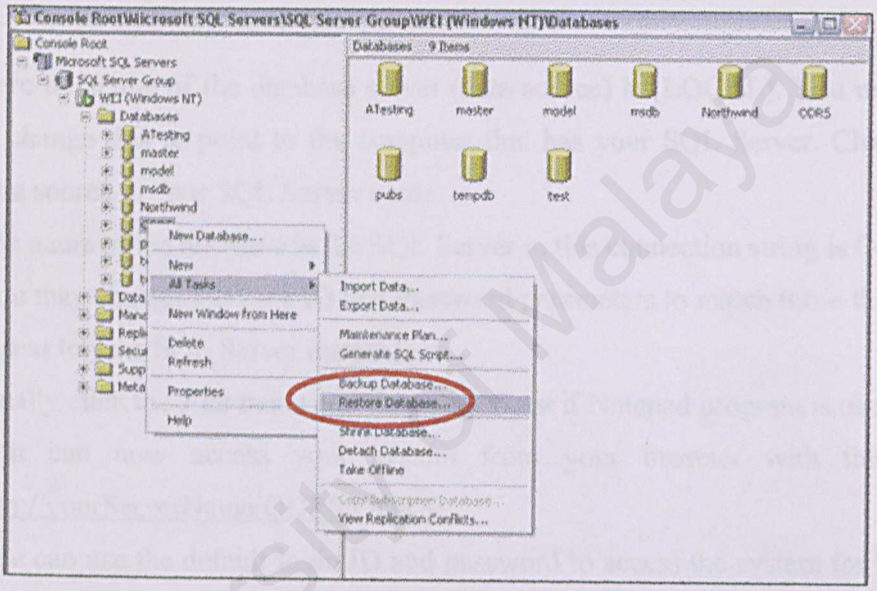
1. Firstly, go to ***Start > All Programs > Microsoft SQL Server > Enterprise Manager***. Expand ***SQL Server Group*** and thus ***<Server Name>*** by clicking +. You will see the screen as shown in Figure 3.4

Figure 3.4 Enterprise Manager



1. Create a new database in ***SQL Server*** named ODRS by right click ***Databases*** directory and choose ***New Database...*** Follow the instruction given and create a database named ODRS.
2. Then with the database created, right click and from the menu pop up, choose ***All Tasks*** > ***Restore Database...*** as shown in Figure 3.5.

Figure 3.5 ODRS Restore Database



3. You will then be in a new restore database window where you can select the restore option.
4. From the Restore, select ***From Device*** follow by ***Select Devices***.
5. In the restore devices window, select ***Add*** and browse for the database file that is located in the CD in the ***ODRS/Database*** folder.
6. Click [Yes] for confirmation and continue to perform the restoration.

3.3 Modifying the Connection String

1. Open any text editor program or ***Notepad*** program in the Window.
2. From the notepad program, open the ***web.config*** file located in the ***C:\Inetpub\wwwroot\ODRS\ODRS\***

3. When you open the file, you will see the codes as below.

```
<?xml version="1.0" encoding="utf-8"?>
<configuration>
<!-- APPLICATION SETTINGS-->
  <appSettings>
    <add key="connectionString" value="data source=(LOCAL); User ID=sa;
Password=; database=ODRS;"/>
    </appSettings>
  <system.web>
```

4. Here the name of the database server (data source) is (LOCAL). You may need to change this to point to the computer that has your SQL Server. Change the data source to your SQL Server name.
5. The name of the database in the SQL Server in this connection string is ODRS.
6. You may change the User ID and Password parameters to match those that allow access to your SQL Server installation.
7. Finally click the ***File*** menu and then click ***Save*** if Notepad program is used.
8. You can now access your system from your browser with this URL http://yourServerName/ODRS/ODRS/ .
9. You can use the default login ID and password to access the system for the first time.

Management Area	Login ID	Password
Administrator	superadmin	123456
Lecturer	lecturer	123456
Postgraduate Coordinator	PCO	123456
Student	student	123456

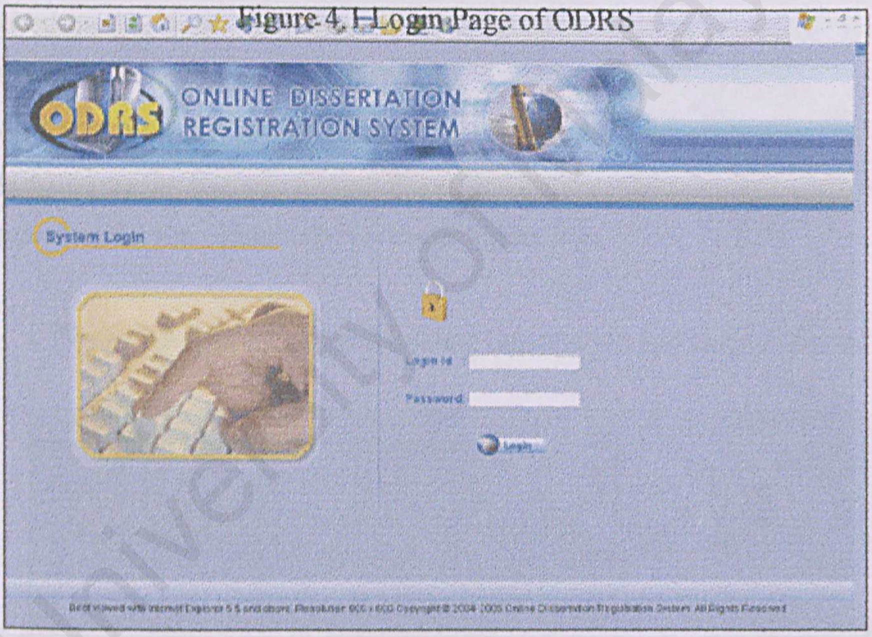
Chapter 4 Getting Started

4.1 Access Authentication

4.1.1 Login / Logout

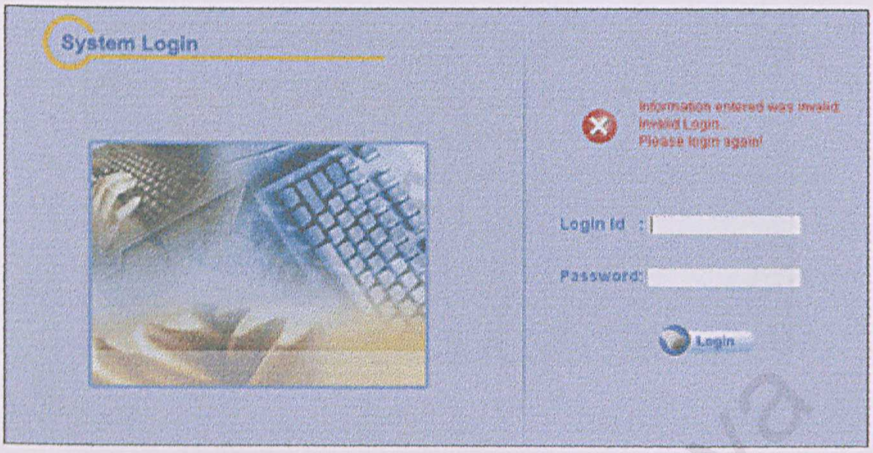
When the users type the URL of ODRS, the login page will be displayed. They have to login by entering their login ID and password in order to continue using the system. System will then check for validation to ensure that they are the authorized user.

Figure 4.1 Login Page of ODRS


ONLINE DISSERTATION REGISTRATION SYSTEM
ODRS
System Login
Login Id
Password
Login
Best viewed with Internet Explorer 5.5 and above. Resolution 800 x 600 Copyright © 2004-2005 Online Dissertation Registration System. All Rights Reserved

1. To start using ODRS, you have to login to the system.
2. Enter your login ID and password in the appropriate field, then click [Login] to access the system.
3. If you login successful, you will be redirected to the access authentication page based on your user type.
4. If a failed login attempt occurred, the message will be displayed and you are required to enter the correct login ID and password again.

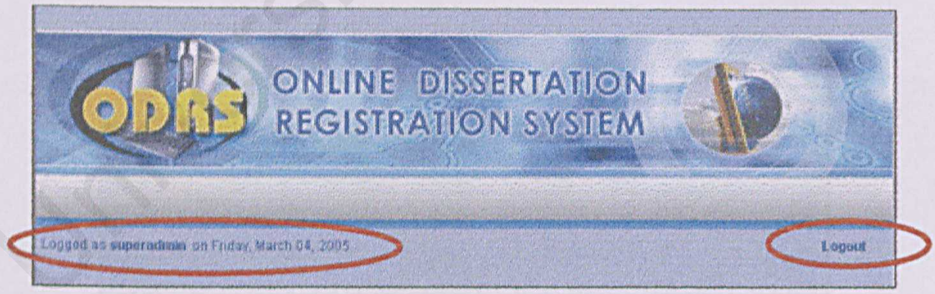
Figure 4.2 Fail Login into ODRS



Information entered was invalid.
Invalid Login..
Please login again!

5. Based on the login ID, different user type will be redirected to different section as different users have different authority in the events management.
6. After you have entered the system, your login ID and accessing date will be appeared on the top left corner below the banner; while the [Logout] button will appear on the top right as shown in the Figure 4.3.

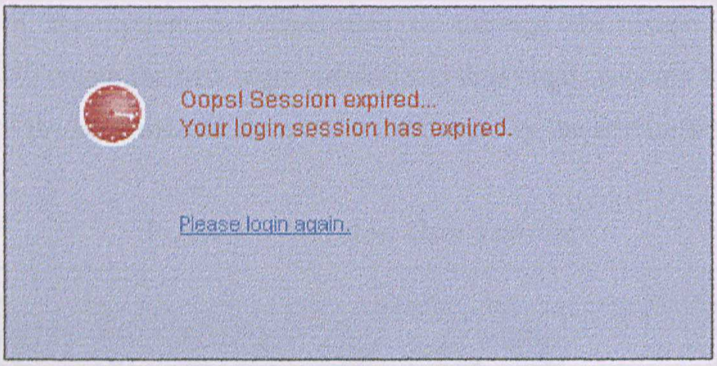
Figure 4.3 Current Logged In User Information and Logout Button



Logged as superadmin on Friday, March 04, 2005

7. Therefore, if you wish to logout, just click on the [Logout] to leave the system.
8. After you had logged out, you will be redirected back to the Login page and you are not allowed to access the system anymore until you login again.
9. During your login session, if you encounter the session expired, you are required to login again in order to continue your access to system.

Figure 4.4 Session Expired



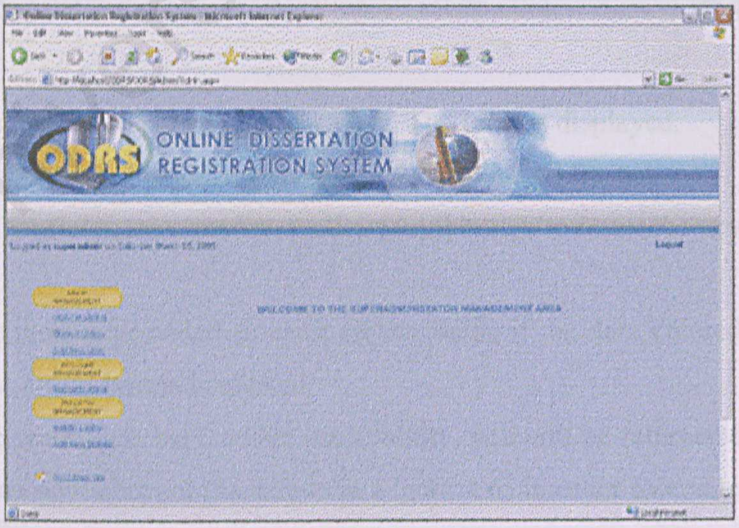
Oops! Session expired...
Your login session has expired.

Please login again.

4.2 Administrator Management Area

If you are the administrator of the system, you will be redirected to the administrator management area's home page, where the main menu is placed on the left of the page (as shown in the Figure 4.5). Go to Main Site is provided in order for administrator to access to the Student main site.

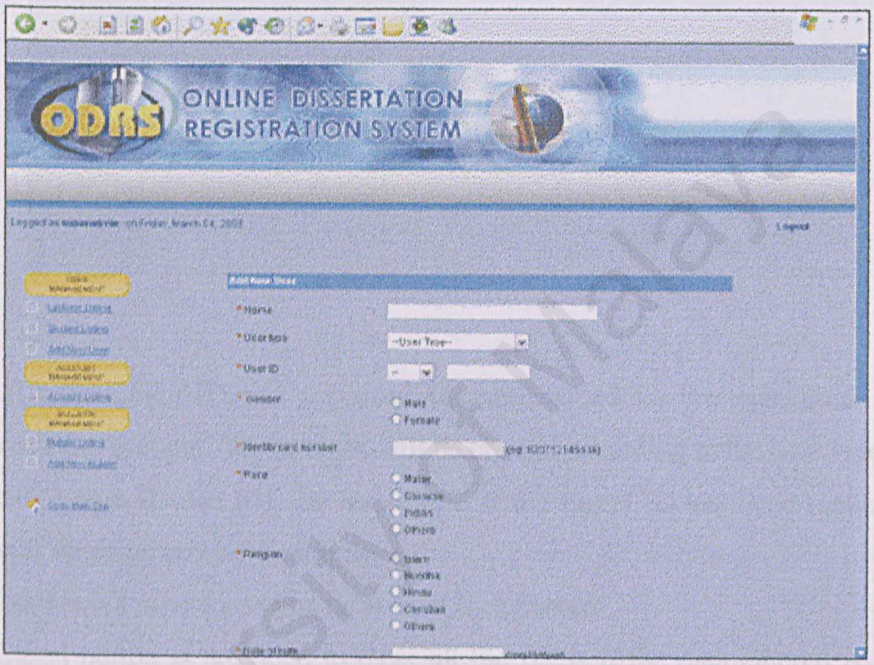
Figure 4.5 Administrator Management Area Home Page



4.2.1 User Management – Add New User

Administrator of the system is responsible to manage the users of the system. Administrator will create the new users' profile and thus create account for the users. All the particulars of the students and lecturers are managed by the administrator.

Figure 4.6 Add New User's profile



Add New User

*Name

*User type: --User Type--

*User ID

*Gender: ○ Male ○ Female

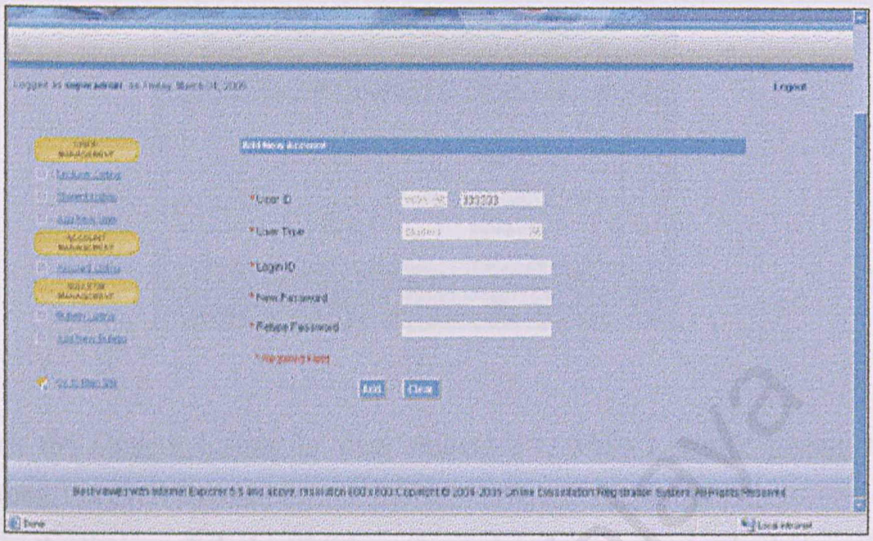
*Identity card number (eg: 820712145530)

*Race: ○ Malay ○ Chinese ○ Indian ○ Others

*Religion: ○ Islam ○ Buddha ○ Hindu ○ Christian ○ Others

- ➢ From the menu, click the Add New User. The form is displayed.
- ➢ Enter all the related fields with proper information and format.
- ➢ After all the fields are completed, click [Add] button to insert the new record into database.
- ➢ [Clear] button is provided to clear all the fields if the data entered is incorrect before the [Add] button is clicked.
- ➢ After the record has been added successfully, you will be redirected to the form of creating new account (as shown in Figure 4.6) in order to create the account for the inserted user.

Figure 4.7 Create New Account Form



Logout

*User ID 333333

*User Type Student

*Login ID

*New Password

*Retype Password

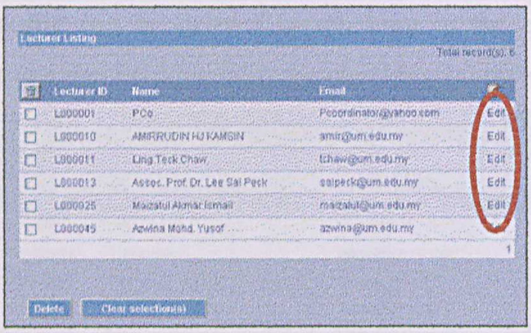
* Mandatory Field

- ➢ After the Login ID and password has been added for the user, click [Add] to create the account. The account will be created according to the user type.
- ➢ Each Login ID can only be used once. If the Login ID has been replicated, the error message will be displayed.
- ➢ The entered password must be at least 6 characters.
- ➢ [Clear] button is provided to clear all the fields if the data entered is incorrect before the [Add] button is clicked.
- ➢ After the [Add] button is clicked, the created account will be entered to the database.
- ➢ The account listing will be displayed upon successful insertion.

4.2.2 User Management – Edit Student/Lecturer Profile

After the new student/lecturer's profile is inserted into the database, the student's profile will be listed in the student listing while the lecturer's profile will be listed in the lecturer listing. The [Edit] button is provided to edit the chosen student/lecturer.

Figure 4.8 Student/Lecturer Listing



Lecturer Listing

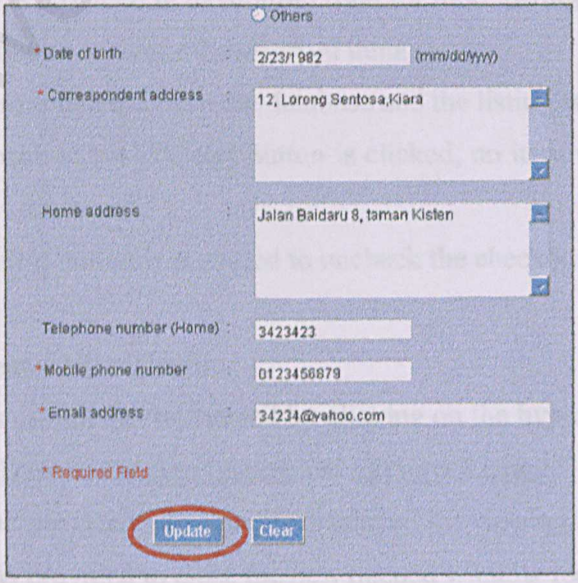
Total record(s): 6

☐	Lecturer ID	Name	Email	
☐	L000001	PCo	Pcoordinator@yahoo.com	Edit
☐	L000010	AMIRRUDIN HJ KAMSIN	amir@um.edu.my	Edit
☐	L000011	Ling Teck Chaw	tchaw@um.edu.my	Edit
☐	L000013	Assoc. Prof. Dr. Lee Sai Peck	saipeck@um.edu.my	Edit
☐	L000025	Maizatul Akmar Ismail	maizatul@um.edu.my	Edit
☐	L000045	Azwina Mohd. Yusof	azwina@um.edu.my	

1

- Click the Student Listing to view student list; while Lecturer Listing to view lecturer list from the menu.
- From the listing, click the [Edit] on the student/lecturer's profile that you wish to modify its data. Paging is provided in the bottom right for more listing.
- Then the chosen profile will be displayed in the form according to the specific fields.
- Modify the data that you wish to edit or update.
- After completed the modification, click [Update] button to update the data in the database.

Figure 4.9 Edit Page of Student



○ Others

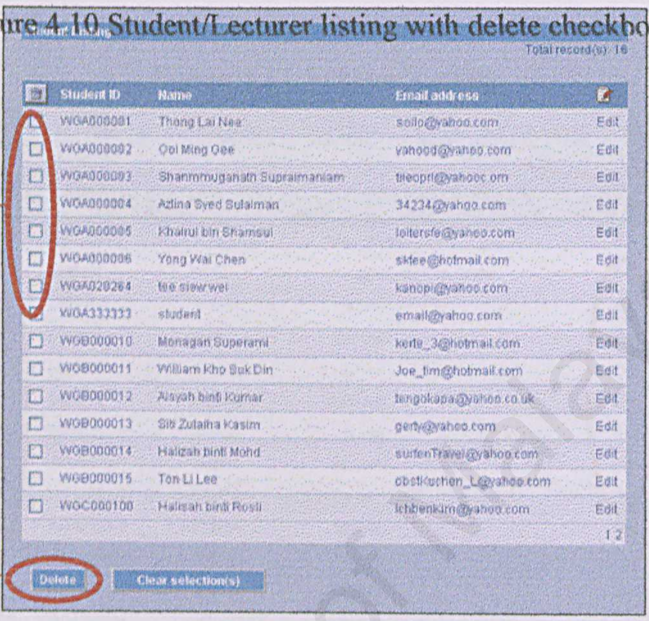
* Date of birth	:	2/23/1982 (mm/dd/yyyy)
* Correspondent address	:	12, Lorong Sentosa,Kiara
Home address	:	Jalan Baidaru 8, taman Kisten
Telephone number (Home)	:	3423423
* Mobile phone number	:	0123456879
* Email address	:	34234@yahoo.com

* Required Field

4.2.3 User Management – Delete Student/Lecturer Profile

The student/lecturer profile can be deleted by clicking in the checkbox(s) beside the item(s). Multiple selections can be done with the checkbox(s) provided.

Figure 4.10 Student/Lecturer listing with delete checkboxes



Total record(s): 16

□	Student ID	Name	Email address	
□	WGA000001	Thong Lai Nee	sollo@yahoo.com	Edit
□	WGA000002	Ooi Ming Gee	yahood@yahoo.com	Edit
□	WGA000003	Shanmmuganath Supraimaniam	tileopri@yahooc.om	Edit
□	WGA000004	Azlina Syed Sulaiman	34234@yahgo.com	Edit
□	WGA000005	Khairul bin Shamsul	loltersfe@yahoo.com	Edit
□	WGA000006	Yong Wai Chen	skfee@hotmail.com	Edit
□	WGA020264	lee siew wei	kanopi@yahoo.com	Edit
□	WGA333333	student	email@yahoo.com	Edit
□	WGB000010	Monagan Superami	kerle_3@hotmail.com	Edit
□	WGB000011	William Kho Suk Din	Joe_tim@hotmail.com	Edit
□	WGB000012	Aisyah binti Kumar	tengokapa@yahoo.co.uk	Edit
□	WGB000013	Siti Zulaiha Kasim	gerty@yahoo.com	Edit
□	WGB000014	Halizah binti Mohd	surfenTravel@yahoo.com	Edit
□	WGB000015	Ton Li Lee	cbstKuchen_L@yahoo.com	Edit
□	WGC000100	Halisah binti Rosli	lchbenkim@yahoo.com	Edit

Checkboxes for deleting

[Delete]

- ➢ From the listing, click on the checkbox(s) to check the item(s) that wish to be deleted. Paging is provided in the bottom right for more listing.
- ➢ Click [Delete] button to delete the checked item(s).
- ➢ Then, the item(s) is deleted from the database and the listing is refreshed.
- ➢ If no item is checked but [Delete] button is clicked, no item will be deleted and error message is displayed.
- ➢ [Clear selection(s)] button is provided to uncheck the checkbox(s).

4.2.4 User Management – View Profile

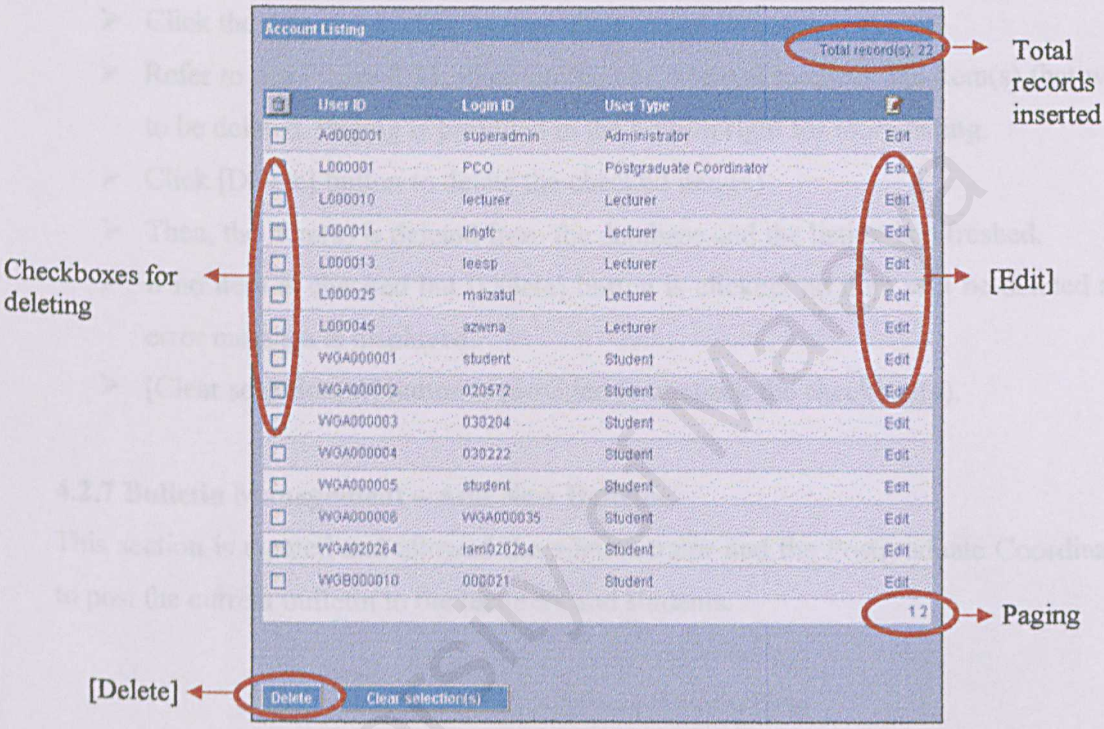
Each student/lecturer's profile can be viewed by clicking on the hyperlink provided with the student/lecturer's ID in the Student Listing and Lecturer Listing.

- ➢ Click on the link provided, the detail is displayed for viewing in new window.
- ➢ [Close] button is provided to close the new opened window for viewing.

4.2.5 Account Management – Edit Account

The account created can be modified its Login ID and password by the administrator only. All the created account will be listed in the listing with paging. The [Edit] is provided to modify the data.

Figure 4.11 Account Listing



Account Listing

Total record(s): 22

🗑	User ID	Login ID	User Type	📝
□	Ad000001	superadmin	Administrator	Edit
□	L000001	PCO	Postgraduate Coordinator	Edit
□	L000010	lecturer	Lecturer	Edit
□	L000011	lingtc	Lecturer	Edit
□	L000013	leesp	Lecturer	Edit
□	L000025	maizatul	Lecturer	Edit
□	L000045	azwina	Lecturer	Edit
□	WGA000001	student	Student	Edit
□	WGA000002	020572	Student	Edit
□	WGA000003	030204	Student	Edit
□	WGA000004	030222	Student	Edit
□	WGA000005	student	Student	Edit
□	WGA000006	WGA000035	Student	Edit
□	WGA020264	lam020264	Student	Edit
□	WGB000010	000021	Student	Edit

1 2

- Click on the Account Listing from the menu.
- From the listing, click the [Edit] on the account that you wish to modify its data. Paging is provided in the bottom right for more listing.
- Then the chosen account will be displayed in the form according to the specific fields.
- Modify the data that you wish to edit or update.
- After completed the modification, click [Update] button to update the data in the database.

4.2.6 Account Management – Delete Account

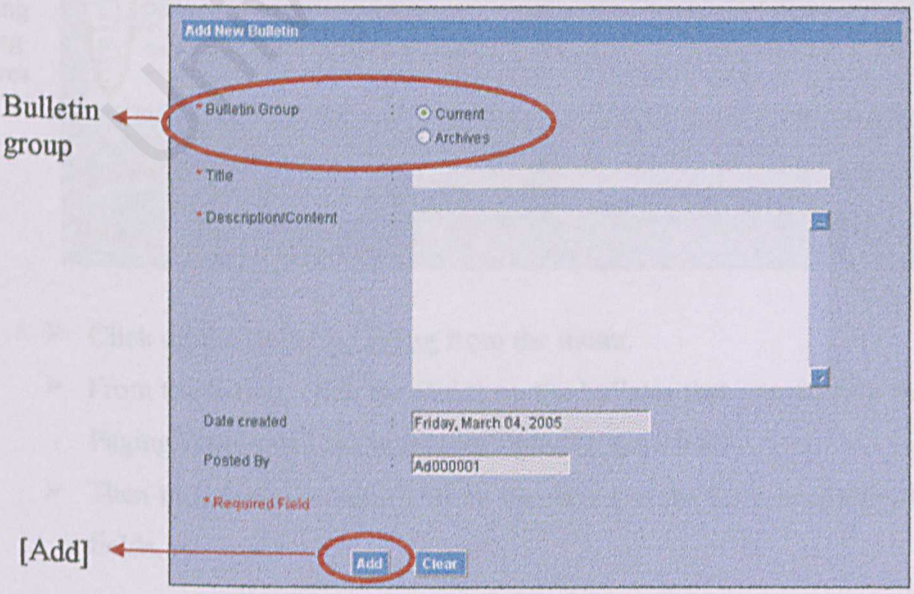
The account can only be deleted by the administrator only. Multiple selections can be done by clicking and checking the checkbox(s) beside the item(s) in order to delete the account.

- Click the Account Listing to view the account list.
- Refer to the Figure 4.11, click on the checkbox(s) to check the item(s) that wish to be deleted. Paging is provided in the bottom right for more listing.
- Click [Delete] button to delete the checked item(s).
- Then, the item(s) is deleted from the database and the listing is refreshed.
- If no item is checked but [Delete] button is clicked, no item will be deleted and error message is displayed.
- [Clear selection(s)] button is provided to uncheck the checkbox(s).

4.2.7 Bulletin Management – Add New Bulletin

This section is created and allowed the administrator and the Postgraduate Coordinator to post the current bulletin to the lecturers and students.

Figure 4.12 Add new bulletin form



Add New Bulletin

* Bulletin Group	:	⊙ Current ○ Archives
* Title	:	
* Description/Content	:	
Date created	:	Friday, March 04, 2005
Posted By	:	Ad000001

* Required Field

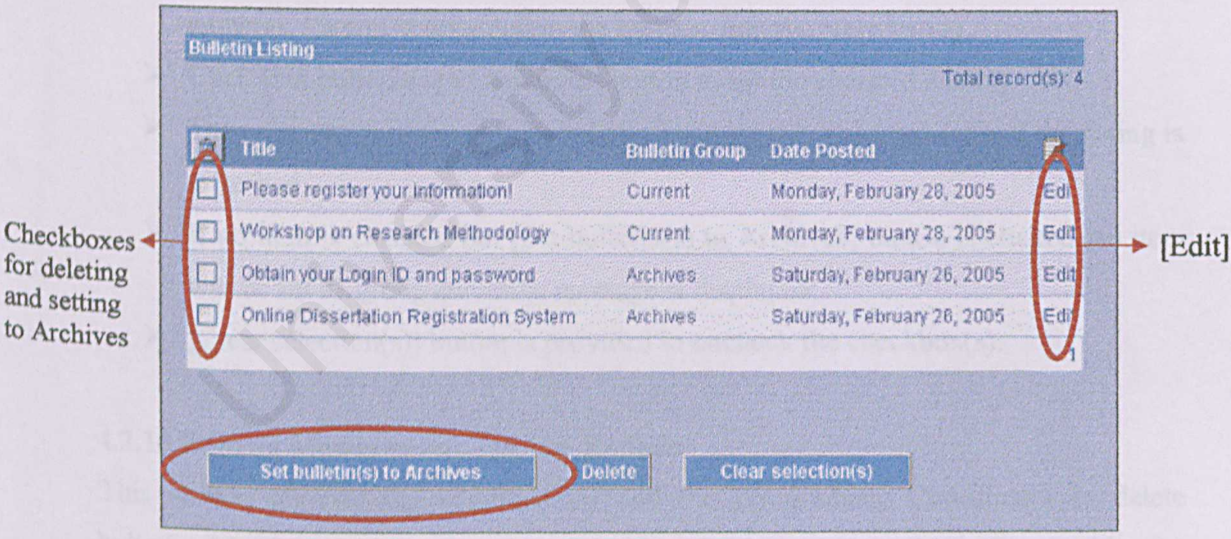
Add Clear

- Click on the Add New Bulletin from the menu.
- The form will be displayed as shown in the Figure 4.12.
- Enter all the related information about the bulletin.
- The bulletin group is important in determining whether the bulletin is up-to-date or archives.
- After completed all the fields, click [Add] to enter the bulletin's detail to the database and thus display it in the main site of the students.
- [Clear] button is provided to clear all the fields if the data entered is incorrect before the [Add] button is clicked.

4.2.8 Bulletin Management – Edit Bulletin

After the new bulletin is inserted into the database, the bulletin will be listed in the Bulletin Listing. The [Edit] button is provided to edit the chosen bulletin.

Figure 4.13 Bulletin Listing



Bulletin Listing

Total record(s): 4

☐	Title	Bulletin Group	Date Posted	
☐	Please register your information!	Current	Monday, February 28, 2005	Edit
☐	Workshop on Research Methodology	Current	Monday, February 28, 2005	Edit
☐	Obtain your Login ID and password	Archives	Saturday, February 26, 2005	Edit
☐	Online Dissertation Registration System	Archives	Saturday, February 26, 2005	Edit

1

Set bulletin(s) to Archives
Delete
Clear selection(s)
Checkboxes for deleting and setting to Archives
[Edit]

- Click on the Bulletin Listing from the menu.
- From the listing, click the [Edit] on the bulletin that you wish to modify its data. Paging is provided in the bottom right for more listing.
- Then the chosen bulletin will be displayed in the form according to the specific fields.

- Modify the data that you wish to edit or update.
- After completed the modification, click [Update] button to update the data in the database.

4.2.9 Bulletin Management – Set bulletin(s) to Archives

Due to the status of the bulletin is important during the display to the lecturers and students, administrator and the Postgraduate Coordinator who responsible in managing the bulletin need to always make sure the right status of the bulletin. The current bulletin will be displayed in the home page of the student section in order to get their attention on latest bulletin. Thus, the [Set bulletin(s) to Archives] is provided to ease the Administrator and Postgraduate Coordinator to manage the status of the bulletin.

- Click the Bulletin Listing from the menu and listing will be displayed as shown in the Figure 4.13.
- Click and check on the checkbox(s) to check the item(s) that wish to be set to archives. Paging is provided in the bottom right for more listing.
- Click [Set bulletin(s) to Archives] button to set the checked item(s).
- Then, the group of the chosen bulletin is updated in the database and the listing is refreshed.
- If no item is checked but [Set bulletin(s) to Archives] button is clicked, no item will be set its group and error message is displayed.
- [Clear selection(s)] button is provided to uncheck the checkbox(s).

4.2.10 Bulletin Management – Delete Bulletin

This section allowed the administrator and the Postgraduate Coordinator to delete bulletin. Multiple selections can be done by clicking and checking the checkbox(s) beside the item(s) in order to delete the bulletin(s).

- Click the Bulletin Listing from the menu and listing will be displayed as shown in the Figure 4.13.

- Click and check on the checkbox(s) to check the item(s) that wish to be deleted. Paging is provided in the bottom right for more listing.
- Click [Delete] button to delete the checked item(s).
- Then, the item(s) is deleted from the database and the listing is refreshed.
- If no item is checked but [Delete] button is clicked, no item will be deleted and error message is displayed.
- [Clear selection(s)] button is provided to uncheck the checkbox(s).

4.2.11 Bulletin Management – View bulletin

Each bulletin's detail can be viewed by clicking on the hyperlink provided with the bulletin title in the Bulletin Listing.

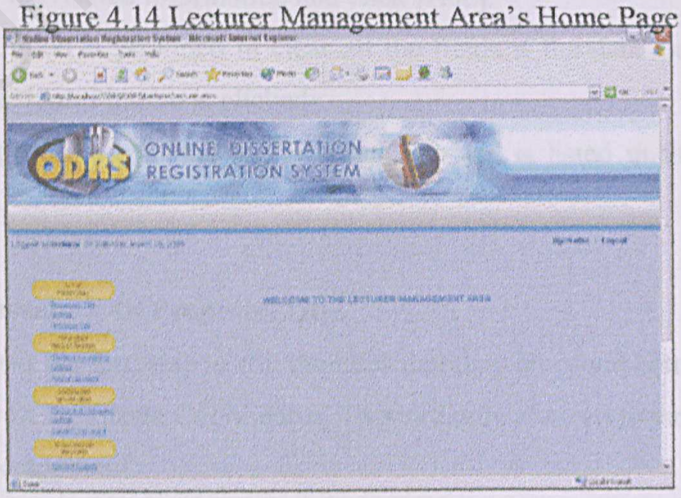
- Click on the title of the bulletin, the detail is displayed for viewing.
- [Close] button is provided to close the new opened window for viewing.

4.3 Lecturer Management Area

If you are the lecturer, you will be redirected to the lecturer management area's home page, where the main menu is placed on the left of the page (as shown in the Figure 4.14).

- Go to Main Site is provided for lecturers to access to the Student main site.
- My Profile is provided for the lecturers to view their profile.

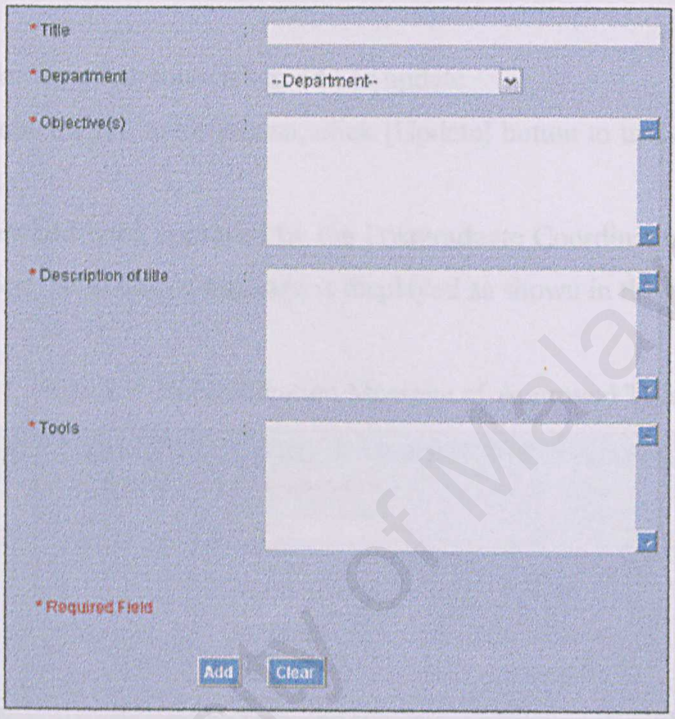
Figure 4.14 Lecturer Management Area's Home Page


ODRS
ONLINE DISSERTATION REGISTRATION SYSTEM
WELCOME TO THE LECTURER MANAGEMENT AREA

4.3.1 Title Proposal – Propose Title

This section gives a simple way to lecturers to propose the titles to the Postgraduate Coordinator.

Figure 4.15 Propose Title Form



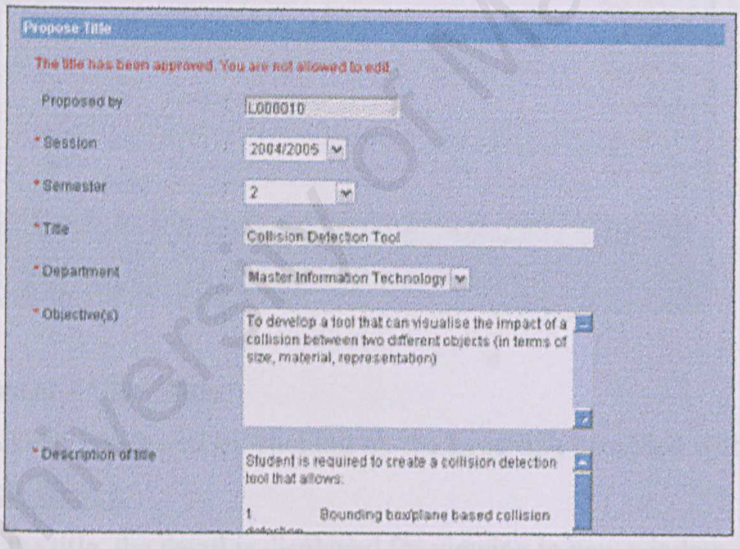
- ➢ Click Propose Title from menu.
- ➢ The propose title form as shown in the Figure 4.15 will be displayed.
- ➢ Enter all the related information and click [Add].
- ➢ [Clear] button is provided to clear all the fields if the data entered is incorrect before the [Add] button is clicked.
- ➢ The proposed title is inserted into database and is listed in the Proposed Title Listing.

4.3.2 Title Proposal – Edit Proposed Title

This section allows the lecturers to edit the titles that they proposed before the titles are approved by the Postgraduate Coordinator. They are only allowed to view the details of the titles after the approval.

- Click on the Proposed Title Listing on the menu.
- From the listing, click the [Edit] on the bulletin that you wish to modify its data. Paging is provided in the bottom right for more listing.
- Then the chosen bulletin will be displayed in the form according to the specific fields.
- Modify the data that you wish to edit or update.
- After completed the modification, click [Update] button to update the data in the database.
- If the titles had been approved by the Postgraduate Coordinator, the data cannot be modified. Notification message is displayed as shown in the Figure 4.16.

Figure 4.16 Notification Message of Approved Title



The title has been approved. You are not allowed to edit

Proposed by	L000010
* Session	2004/2005
* Semester	2
* Title	Collision Detection Tool
* Department	Master Information Technology
* Objective(s)	To develop a tool that can visualise the impact of a collision between two different objects (in terms of size, material, representation)
* Description of title	Student is required to create a collision detection tool that allows: 1 Bounding box/plane based collision detection

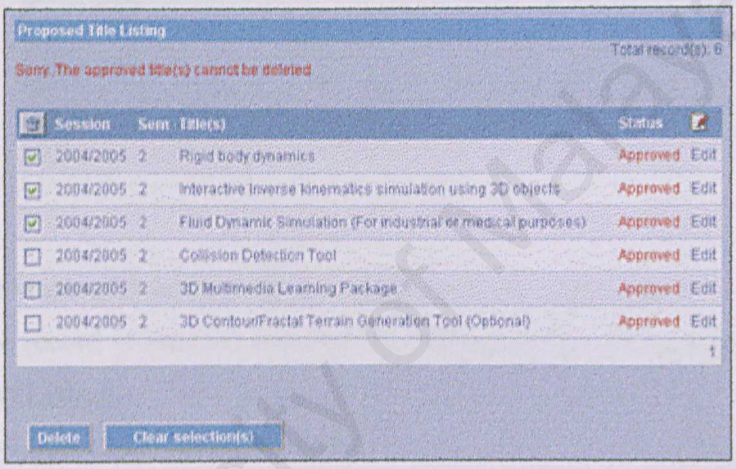
4.3.3 Title Proposal – Delete Proposed Title

This section allowed the lecturer to delete the proposed titles before the titles are approved. Multiple selections can be done by clicking and checking the checkbox(s) beside the item(s) in order to delete the title(s).

- Click the Proposed Title Listing from the menu and listing will be displayed.
- Click and check on the checkbox(s) to check the item(s) that wish to be deleted. Paging is provided in the bottom right for more listing.

- Click [Delete] button to delete the checked item(s).
- Then, the item(s) is deleted from the database and the listing is refreshed.
- If no item is checked but [Delete] button is clicked, no item will be deleted and error message is displayed.
- If the selected title(s) are with the Approved status, deletion is denied and the notification message is displayed as shown in the Figure 4.17.
- [Clear selection(s)] button is provided to uncheck the checkbox(s).

Figure 4.17 Notification Message of Denial of Deletion



Proposed Title Listing

Total record(s): 6

Sorry. The approved title(s) cannot be deleted

☑	Session	Sem	Title(s)	Status	
☑	2004/2005	2	Rigid body dynamics	Approved	Edit
☑	2004/2005	2	Interactive Inverse kinematics simulation using 3D objects	Approved	Edit
☑	2004/2005	2	Fluid Dynamic Simulation (For industrial or medical purposes)	Approved	Edit
□	2004/2005	2	Collision Detection Tool	Approved	Edit
□	2004/2005	2	3D Multimedia Learning Package	Approved	Edit
□	2004/2005	2	3D Contour/Fractal Terrain Generation Tool (Optional)	Approved	Edit

1

4.3.4 Title Proposal – View Title

Each title's detail can be viewed by clicking on the hyperlink provided with the Title in the Proposed Title Listing.

- Click on the title, the detail is displayed for viewing in new window.
- [Close] button is provided to close the new opened window for viewing.

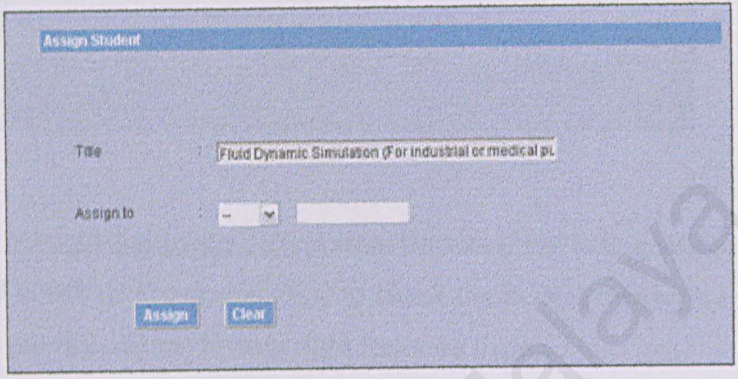
4.3.5 Student Registration – Assign Student

This section is specifically for the lecturers to assign the title to the students after the consensus had been achieved among them. Lecturer can only assign a title to a student who is a registered user of the system and at the same time too, a student can only be assigned a title.

- Click Assign student from the menu.

- The approved titles are listed. You can choose the title you wish to assign and click the [Assign] to start the assigning process.
- Then, you will be redirected to an assigning page as shown in Figure 4.18.

Figure 4.18 Assigning Student Page



Assign Student

Title : Fluid Dynamic Simulation (For industrial or medical pi.

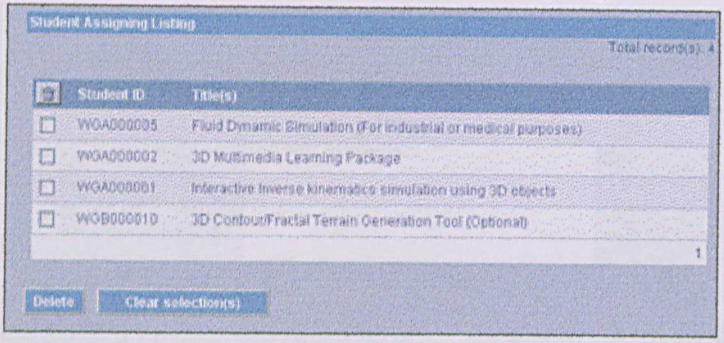
Assign to : --

- Enter the student ID in the particular field. Then click [Assign].
- Then, the record is inserted successfully; the assigning will be listed in <u>Student Assigning Listing</u>.
- Otherwise, record will not be inserted,
  - if the title is already registered by others.
  - if the student is not a registered student.
  - if the student has already been assign another title.
- Error message will be displayed to notify the lecturer.

4.3.6 Student Registration – Delete Assigning

This section allows the lecturer to do either single or multiple deletions on assigning of the students for the titles.

Figure 4.19 Delete Students Assigning



☐	Student ID	Title(s)
☐	WGA000005	Fluid Dynamic Simulation (For industrial or medical purposes)
☐	WGA000002	3D Multimedia Learning Package
☐	WGA000001	Interactive Inverse kinematics simulation using 3D objects
☐	WGB000010	3D Contour/Fractal Terrain Generation Tool (Optional)

- ➢ Click the Student Assigning Listing from the menu and listing will be displayed.
- ➢ Click and check on the checkbox(s) to check the item(s) that wish to be deleted. Paging is provided in the bottom right for more listing.
- ➢ Click [Delete] button to delete the checked item(s).
- ➢ Then, the item(s) is deleted from the database and the listing is refreshed.
- ➢ If no item is checked but [Delete] button is clicked, no item will be deleted and error message is displayed.
- ➢ [Clear selection(s)] button is provided to uncheck the checkbox(s).

4.3.7 Student Registration – View Assigning

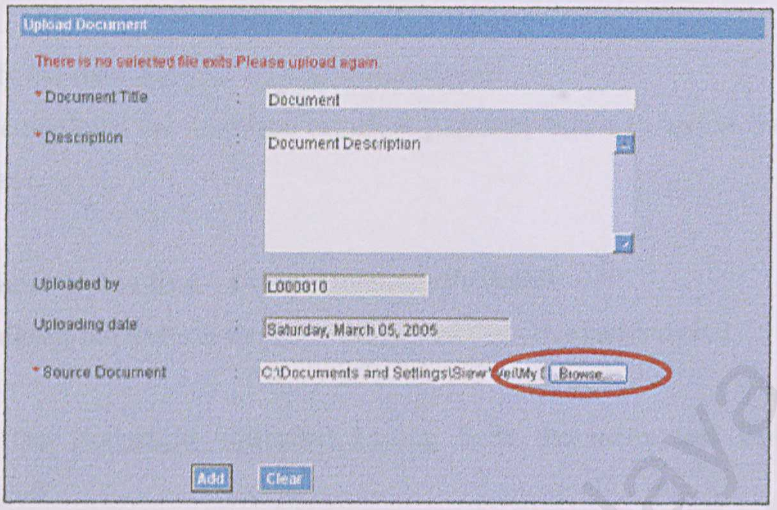
In the Student Assigning Listing, the student ID and Title provided hyperlinks for lecturers to view the students' and titles' details. Thus, lecturer can easy browser through without any troublesome.

- ➢ Click on the student ID/title, the detail is displayed for viewing in new window.
- ➢ [Close] button is provided to close the new opened window for viewing.

4.3.8 Document Uploading – Upload Document

This section allows the lecturers to upload the articles, pdfs, forms or other documents together with the documents' description to let the students to download.

Figure 4.20 Upload Document


Upload Document

There is no selected file exits.Please upload again.

*Document Title : Document

*Description : Document Description

Uploaded by : L000010

Uploading date : Saturday, March 05, 2005

*Source Document : C:\Documents and Settings\Siew\...\My [

- Click Upload Document from menu.
- The upload document form as shown in the Figure 4.20 will be displayed.
- Enter all the related information.
- Click [Browse] in order to choose a document to upload.
- Then, click [Add] to upload the document.
- [Clear] button is provided to clear all the fields if the data entered is incorrect before the [Add] button is clicked.
- If invalid file names or no file is uploaded, error message will be displayed to notify users.
- The uploaded document is listed upon successful insertion.

4.3.9 Document Uploading – Edit Document's detail

This section will give way to lecturer who uploaded document to edit or update the details of the document.

- Click on the Document Uploaded Listing on the menu.
- From the listing, click the [Edit] on the document's detail that you wish to modify its data. Paging is provided in the bottom right for more listing.

- Then the chosen item will be displayed in the form according to the specific fields.
- Modify the data that you wish to edit or update.
- After completed the modification, click [Update] button to update the data in the database.

4.3.10 Document Uploading – Delete Document's detail

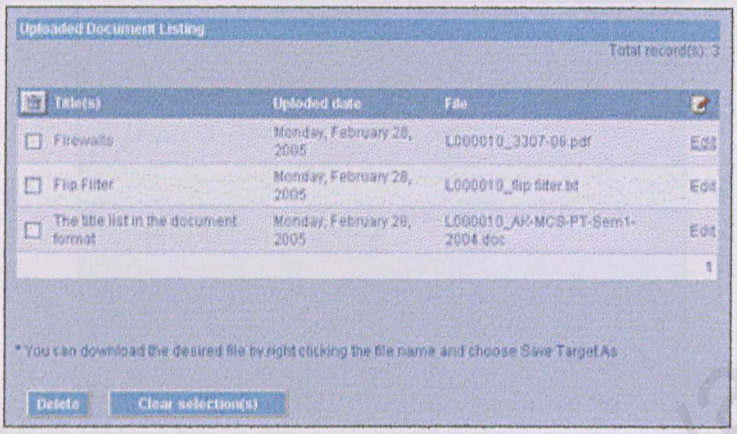
This section allows the lecturer to delete the record that they had inserted.

- Click the <u>Document Uploaded Listing</u> from the menu and listing will be displayed.
- Click and check on the checkbox(s) to check the item(s) that wish to be deleted. Paging is provided in the bottom right for more listing.
- Click [Delete] button to delete the checked item(s).
- Then, the item(s) is deleted from the database and the listing is refreshed.
- If no item is checked but [Delete] button is clicked, no item will be deleted and error message is displayed.
- [Clear selection(s)] button is provided to uncheck the checkbox(s).

4.3.11 Document Uploading – Download and View Document's details

From the provided listing, the detail of the documents can be viewed. In addition, lecturer can also download the documents that have been uploaded.

Figure 4.21 Document Uploaded Listing



Uploaded Document Listing

Total record(s): 3

☐ Title(s)	Uploded date	File	
☐ Firewalls	Monday, February 28, 2005	L000010_3307-06.pdf	Edit
☐ Flip Filter	Monday, February 28, 2005	L000010_flip filter.txt	Edit
☐ The title list in the document format	Monday, February 28, 2005	L000010_AK-MCS-PT-Sem1-2004.doc	Edit
			1

* You can download the desired file by right clicking the file name and choose Save Target As

The uploaded documents' detail can be viewed by clicking on the hyperlink provided with the Title in the Document Uploaded Listing.

- Click on the title, the detail is displayed for viewing in new window.
- [Close] button is provided to close the new opened window for viewing.

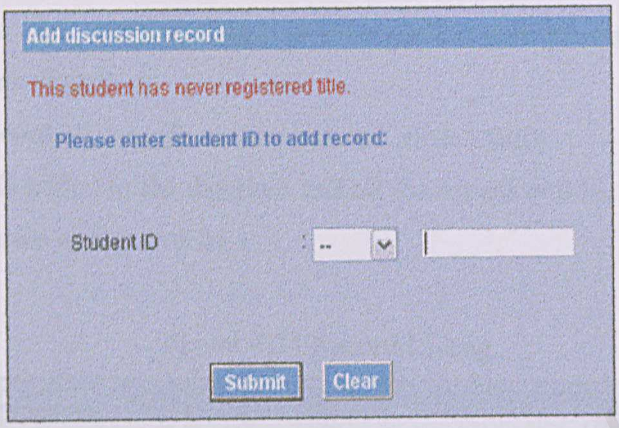
The uploaded documents' filename, types can also be viewed and downloaded from the listing.

- Right click on the chosen file and choose Save Target As.
- The splash window screen will pop up to locate your file.

4.3.12 Discussion Record – Add New Record

This discussion record module is designed for lecturer to record the face-to-face discussion's detail together with their respective students. Lecturers can add any discussion records online without taking down any notes on the paper.

Figure 4.22 Add New Record



Add discussion record

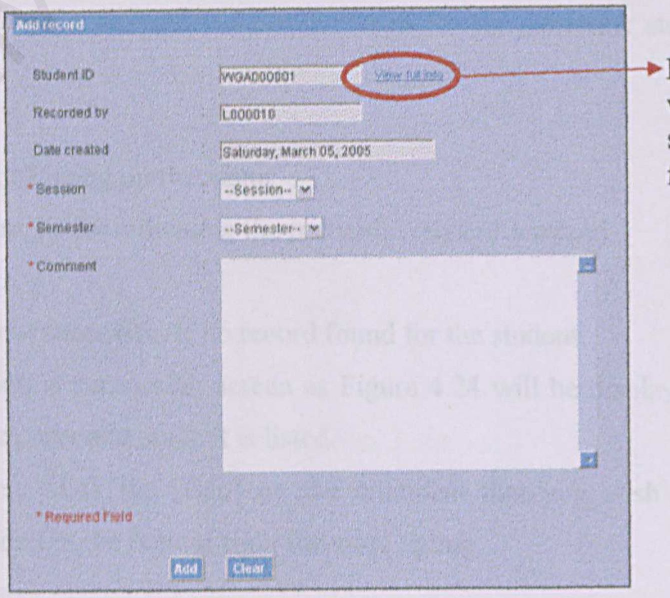
This student has never registered title.

Please enter student ID to add record:

Student ID : --

- Click the Add Record from the menu.
- The screen as shown in Figure 4.22 will be displayed.
- Enter the student ID that you wish to add the record the respective student.
- The student ID must be entered properly.
- Then, click [Submit]. You will be redirected to the ***add new record*** form.
- Error message will be shown if the entered student ID has never registered any title. Thus, no record can be added.
- If the submission is successful, the ***add new record*** form is displayed as shown in the Figure 4.23.

Figure 4.23 Add New Record Form



Add record

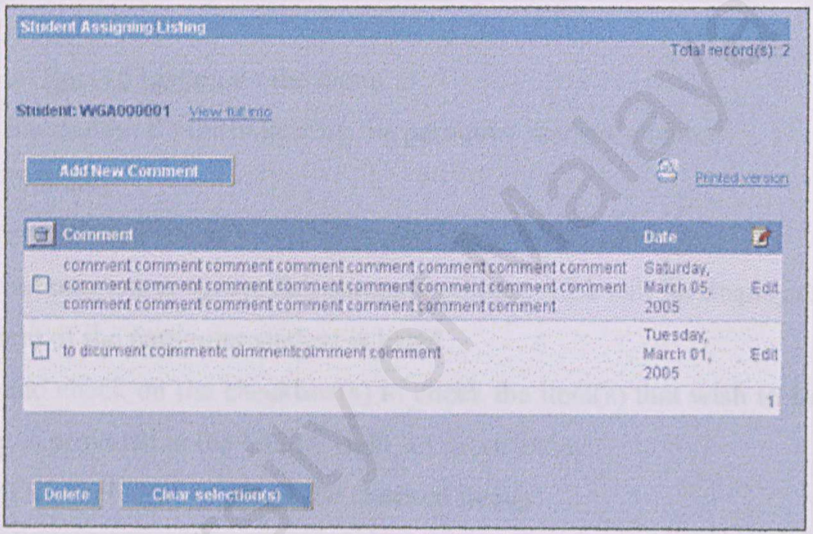
Student ID	WGA000001 View full info
Recorded by	L000010
Date created	Saturday, March 05, 2005
* Session	--Session--
* Semester	--Semester--
* Comment	

* Required Field

Link to view student's full info

- Enter all the related information and comment.
- Lecturer can click View Full Info link beside the particular student to view the full particular of student.
- After completed all the information entered, click [Add].
- The record is added to the database and all the record will be listed in the record listing as shown in Figure 4.24.

Figure 4.24 Record Listing



Student Assigning Listing

Total record(s): 2

Student: WGA000001 View full info

	Comment	Date	
□	comment comment	Saturday, March 05, 2005	Edit
□	to dicument coimmentc oimmentcoimment coimment	Tuesday, March 01, 2005	Edit

1

4.3.13 Discussion Record – Edit Record

Lecturer can edit/update the added record or comment for the particular student as the steps shown below.

- Click on Record Listing on the menu.
- Enter the student ID for indicating the particular student's record.
- Click [Submit].
- Submission is not successful if no record found for the student.
- If the submission is successful, screen as Figure 4.24 will be displayed. All the comment of the particular student is listed.
- From the listing, click the [Edit] on the comment that you wish to modify. Paging is provided in the bottom right for more listing.

- Then the chosen item will be displayed in the form according to the specific fields.
- Modify the data that you wish to edit or update.
- After completed the modification, click [Update] button to update the data in the database.

4.3.14 Discussion Record – Delete Record

This section allows the comment added to be deleted.

- Click on Record Listing on the menu.
- Enter the student ID for indicating the particular student's record.
- Click [Submit].
- Submission is not successful if no record found for the student.
- If the submission is successful, screen as Figure 4.24 will be displayed. All the comment of the particular student is listed.
- Click and check on the checkbox(s) to check the item(s) that wish to be deleted. Paging is provided in the bottom right for more listing.
- Click [Delete] button to delete the checked item(s).
- Then, the item(s) is deleted from the database and the listing is refreshed.
- If no item is checked but [Delete] button is clicked, no item will be deleted and error message is displayed.
- [Clear selection(s)] button is provided to uncheck the checkbox(s).

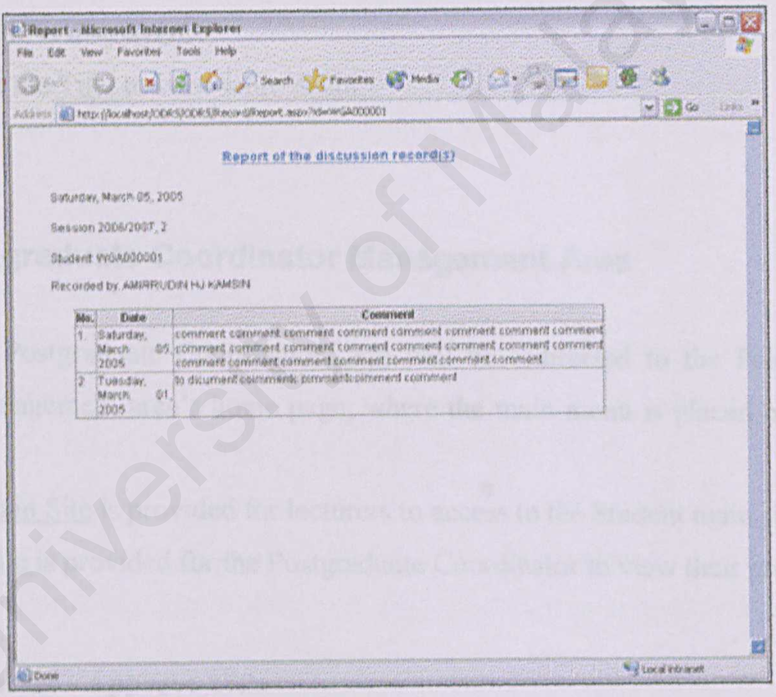
4.3.15 Discussion Record – View and Print Record

Each added record can be viewed from the listing and the printed version of the report is provided in this section.

- Click on Record Listing on the menu.
- Enter the student ID for indicating the particular student's record.
- Click [Submit].
- Submission is not successful if no record found for the student.

- If the submission is successful, screen as Figure 4.24 will be displayed.
- All the comment of the particular student can be viewed.
- The [Add New Record] is provided if the lecturer wish to add the record after viewing the previous records.
- The Printed version is provided to let the lecturer to print the summary report of comments of the particular student.
- Click on the Printed version, the new window with the printed version of report is displayed as Figure 4.25.

Figure 4.25 Printed version of Student's record


Report - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://localhost/ODRS/ODRS/Record/Report.aspx?id=WGA000001
Go Links
Done
Local intranet

Report of the discussion record(s)

Saturday, March 05, 2005

Session 2006/2007, 2

Student WGA000001

Recorded by AMIRRUDIN HJ KAMSIN

No.	Date	Comment
1.	Saturday, March 05, 2005	comment comment
2	Tuesday, March 01, 2005	to dicument comment comment comment comment

- In the new window, click ***File*** > ***Print***, a splash screen is displayed.
- Choose your printer, and then click [Print].

4.3.16 View Student Info

This section gives a way to lecturer to check and view students' full particulars that are registered by the administrator and students themselves for the discussion record.

- Click on View Student Full Info on the menu.

- Enter the student ID for indicating the particular student's record.
- Click [Submit].
- Submission is not successful if no record found for the student.
- If the submission is successful, the student's full information will be displayed in a new window for viewing.

4.3.17 View Calendar Event

This section will display the listing of all the calendar events that are entered by the Postgraduate Coordinator about the dissertation.

- Click [View Calendar Event].
- The listing is displayed.
- Click on the Title to view more detail.

4.4 The Postgraduate Coordinator Management Area

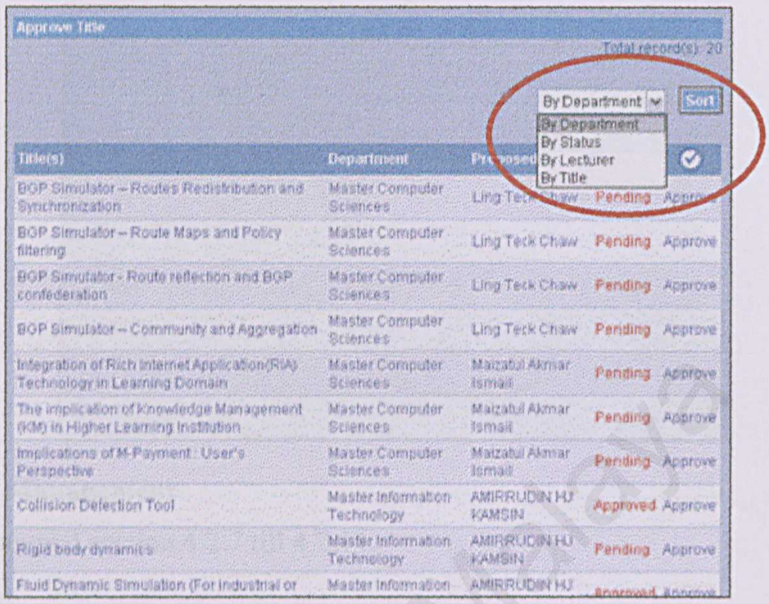
If you are the Postgraduate Coordinator, you will be redirected to the Postgraduate Coordinator management area's home page, where the main menu is placed on the left of the page.

- Go to Main Site is provided for lecturers to access to the Student main site.
- My Profile is provided for the Postgraduate Coordinator to view their profile.

4.4.1 Title Proposal – Approve Title

This section is specifically designed to the Postgraduate Coordinator to approve the lecturers' proposed titles.

Figure 4.26 Approve Titles



Approve Title

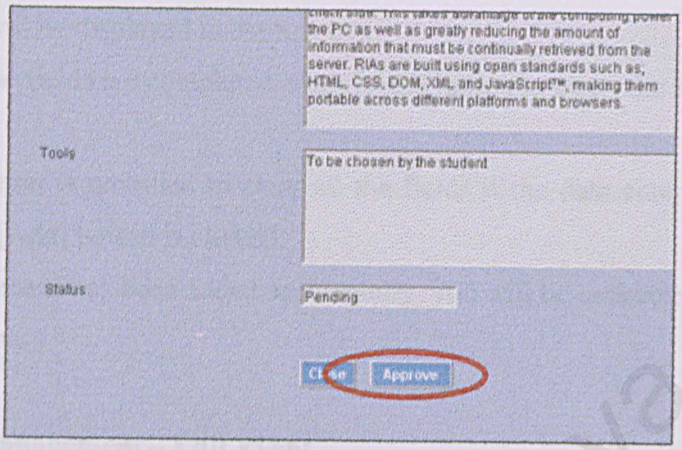
Total record(s): 20

By Department ▾ Sort

Title(s)	Department	Proposed		✓
BGP Simulator – Routes Redistribution and Synchronization	Master Computer Sciences	Ling Teck Chaw	Pending	Approve
BGP Simulator – Route Maps and Policy filtering	Master Computer Sciences	Ling Teck Chaw	Pending	Approve
BGP Simulator - Route reflection and BGP confederation	Master Computer Sciences	Ling Teck Chaw	Pending	Approve
BGP Simulator – Community and Aggregation	Master Computer Sciences	Ling Teck Chaw	Pending	Approve
Integration of Rich Internet Application(RIA) Technology in Learning Domain	Master Computer Sciences	Maizatul Akmar Ismail	Pending	Approve
The implication of Knowledge Management (KM) in Higher Learning Institution	Master Computer Sciences	Maizatul Akmar Ismail	Pending	Approve
Implications of M-Payment : User's Perspective	Master Computer Sciences	Maizatul Akmar Ismail	Pending	Approve
Collision Detection Tool	Master Information Technology	AMIRRUDIN HJ KAMSIN	Approved	Approve
Rigid body dynamits	Master Information Technology	AMIRRUDIN HJ KAMSIN	Pending	Approve
Fluid Dynamic Simulation (For industrial or	Master Information	AMIRRUDIN HJ	Approved	Approve

- Click Approve Proposed Title on the menu.
- The listing will be displayed.
- On the top right corner, sorting by types is provided. Paging is provided in the bottom right for more listing.
- Choose the type that you wish to sort, and then click [Sort].
- The listing will be sorted as you wish.
- Click [Approve] to the particular title that you wish to approve.
- The notification message is displayed upon the successfully approval.
- Title link is provided to view the detail description of the proposed titles.
- The [Approve] button is provided too at the bottom of the details to ease you to approve the title once after viewing.

Figure 4.27 Detail Description of Proposed Title



client side. This takes advantage of the computing power the PC as well as greatly reducing the amount of information that must be continually retrieved from the server. RIAs are built using open standards such as, HTML, CSS, DOM, XML, and JavaScript™, making them portable across different platforms and browsers.

Tools: To be chosen by the student

Status: Pending

4.4.2 Bulletin Management

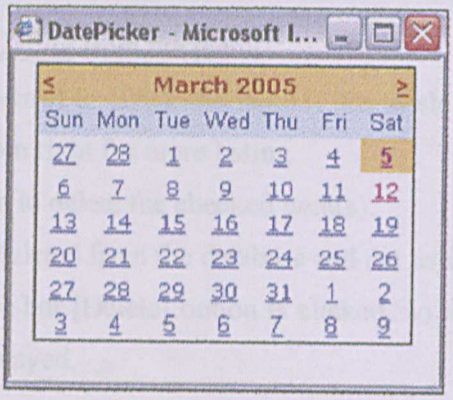
Refer to user manual section 4.2.7 till 4.2.11.

4.4.3 Calendar Management – Add New Event

This section provides the facility to the Postgraduate Coordinator to add the new event of Calendar (Dissertation).

- ➢ From the menu, click the <u>Add New Event</u>. The form is displayed.
- ➢ Enter all the related fields with proper information and format.
- ➢ Calendar is provided to pick a suitable date for the event.
- ➢ Click <u>Pick Date from Calendar</u>, and then the calendar will be pop up as shown in Figure 4.28.

Figure 4.28 Calendar Picker



DatePicker - Microsoft I...

≤ March 2005 ≥

Sun	Mon	Tue	Wed	Thu	Fri	Sat
27	28	1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31	1	2
3	4	5	6	7	8	9

- Choose your preferred date by clicking it.
- The date will be displayed in the text box.
- After all the fields are completed, click [Add] button to insert the new record into database.
- [Clear] button is provided to clear all the fields if the data entered is incorrect before the [Add] button is clicked.
- After the record has been added successfully, you will be redirected to the listing of the events.

4.4.4 Calendar Management – Edit Event

The Postgraduate Coordinator can edit the events which are inserted previously.

- Click on the Calendar Listing from the menu.
- From the listing, click the [Edit] on the event that you wish to modify its data. Paging is provided in the bottom right for more listing.
- Then the chosen event will be displayed in the form according to the specific fields.
- Modify the data that you wish to edit or update.
- After completed the modification, click [Update] button to update the data in the database.

4.4.5 Calendar Management – Delete Event

This section allows the inserted calendar events to be deleted.

- Click the Calendar Listing from the menu.
- Click on the checkbox(s) to check the item(s) that wish to be deleted. Paging is provided in the bottom right for more listing.
- Click [Delete] button to delete the checked item(s).
- Then, the item(s) is deleted from the database and the listing is refreshed.
- If no item is checked but [Delete] button is clicked, no item will be deleted and error message is displayed.

- ➢ [Clear selection(s)] button is provided to uncheck the checkbox(s).

4.4.6 Calendar Management – View Event

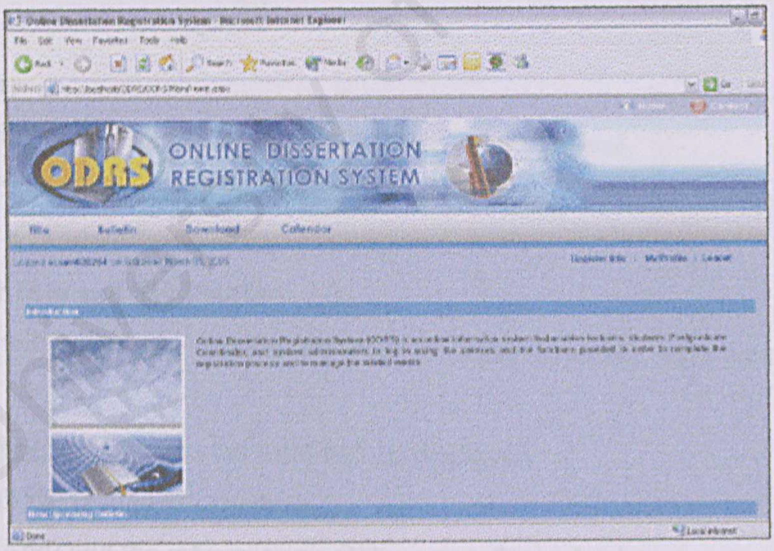
All the events and detail can be viewed.

- ➢ Click on the events' title, the detail is displayed for viewing in new window.
- ➢ [Close] button is provided to close the new opened window for viewing.

4.5 Student Main Site

There are 7 sections in the student main site. These include Title, Bulletin, Download, Calendar, Contact, Register Info and My Profile. Students just need to click on the Menu according to your preferred section. Figure 4.29 show the menu of the main site is located at the top of the whole page.

Figure 4.29 Menu Bar of Main Site



4.5.1 Title

This section will list all the approved titles according to the department. The titles together with further details will be displayed.

- ➢ Click on each Title to view the title's description.
- ➢ Student Assigning Listing is also provided to view the students who have been assigned titles.

4.5.2 Bulletin

This section will be listed the current and archives bulletins that are posted by the administrator and the Postgraduate Coordinator. The current bulletin will be displayed in the New Upcoming Bulletin section in the home page in order to obtain the attention from the students and lecturers about the current bulletins.

- Click on Read More link provided to read more about the bulletins' detail.

4.5.3 Download

This section allows the students to download the documents that are uploaded by the lecturers. The Top 5 new upcoming download will also be listed in the home page of the main site in order to get the students' attention.

- Click on the Document Title to view the detail.
- *Right click on the* Download *then choose for* Save Target As *in order to* download the desired file.

4.5.4 Calendar

This section allows the events about the dissertation to be viewed.

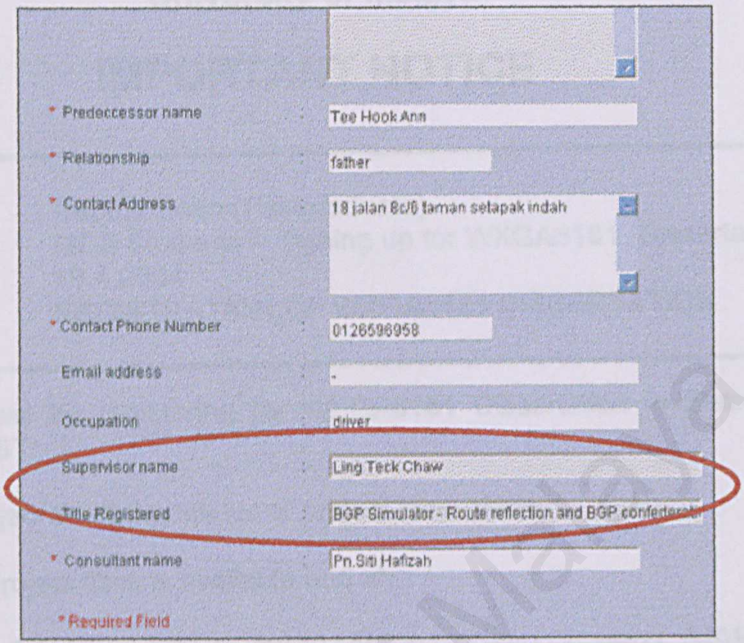
- Click on the Event Title in order to read more about the event.

4.5.5 Register Info and My Profile

This section will allow the student to register some personal information to their supervisors.

- Click Register Info and the form will be displayed.
- Enter all the related information.
- Click [Add] and the information will be inserted into database.
- If the student has been assigned the title by his/her respective lecturer, the title and the supervisor name will be appeared in the field for references. (Figure 4.30)

Figure 4.30 Register Info Form



* Predeccessor name	:	Tee Hook Ann
* Relationship	:	father
* Contact Address	:	18 jalan 8c/6 taman setapak indah
* Contact Phone Number	:	0126596958
Email address	:	-
Occupation	:	driver
Supervisor name	:	Ling Teck Chaw
Title Registered	:	BGP Simulator - Route reflection and BGP confederati
* Consultant name	:	Pn.Siti Hafizah

* Required Field

4.5.6 Contact

This section is listed down all the lecturers' contact number and email address to ease the student to contact their respective lecturers.

- ➢ Click on the email address of the chosen lecturer, you will be redirected to send your mail by your default mailing machine.

Appendix B

MCS/2004

Faculty of Computer Science and Information Technology
University of Malaya

IMPORTANT NOTICE

From : Deputy Dean (Postgraduate)
To : MCS Students – Signing up for WXGA6181 (Dissertation)
Date : 19.4.2004
Subject : **REGISTRATION OF WXGA6181 DISSERTATION**

Students who will be registering for WXGA6181 Dissertation in Semester 1, 2004/2005 **MUST**:

a. select a project title from the list of project titles given.

The list of project titles is available and at:

http://www.fsktm.um.edu.my/notispelajar/WXGA6181sem1_0405.htm

b. get a copy of the **"WXGA6181 Dissertation – Registration Form"** and **Borang Permohonan Pelantikan Penyelia** (UM-PT01-PK02-BR02-S00) or from the general office at the Main Building or download from the above URL.

c. fill in the forms after getting the approval from the project supervisor.

d. get the project supervisor to sign the form.

e. submit the registration form to Cik Habibah by **7 June 2004**.

f. register for WXGA6181 at Institut Pengajian Siswazah (IPS).

g. write a research proposal which should be submitted to your supervisor at the latest 2 months after your registration.

NOTE:

1. **Students are advised to find out the details about the project title from the lecturer concerned before signing up for WXGC6181.**

2. **Selection of project title is based on the approval from the project supervisor concerned and the Head of Department or Deputy Dean (Postgraduate).**

BORANG PENYELIAAN PELAJAR IJAZAH TINGGI

No. Pendaftaran/Kad Pengenalan: ______________________________

Tempoh Pencalonan: ______________________________

NAMA: ______________________________
(*mengikut kad pengenalan*)

Tarikh Lahir: ______________________________

Alamat Sesi: ______________________________

____

____

____

(*Pertukaran alamat perlu dimaklumkan kepada Pejabat Timbalan Dekan (Ijazah Tinggi)*

No. Telefon: ______________________________

Nama Waris (*nyatakan hubungan*)______________________________

____

Alamat: ______________________________

____

____

No. Telefon: ______________________________

Pekerjaan: ______________________________

Nama/Alamat Majikan:______________________________

____

Nama Penyelia Pertama: ______________________________

Nama Penyelia Kedua: ______________________________

Nama Perunding: ______________________________

SEMESTER _____

PERTEMUAN	ULASAN PENYELIA	TANDATANGAN PENYELIA
TARIKH:		
TARIKH:		
TARIKH:		
TARIKH:		