

3

↓  
compactive.

**ONLINE POSTGRADUATE COURSE REGISTRATION AND  
PAYMENT SYSTEM AT FSKTM, UM.**

**A Dissertation Submitted To The  
Faculty Of Computer Science And Information Technology  
University Of Malaya**

**By**

**SEE CHUN KIAT**

**In Partial Fulfillment Of The Requirement For The  
Degree Of Master Of Information Technology**

**October 2002**

Perpustakaan Universiti Malaya



A511140124

## ABSTRACT

An Online Postgraduate Course Registration and Payment System is developed for postgraduate students in the Faculty of Computer Science and Information Technology (FSKTM), University of Malaya (UM). It is a Web application development. The system is developed to facilitate student online course registration and payment of fees. The system overcomes the queuing problem during the course registration and fees payment. UM administrators can view, delete and update the course registration, student, and administrative record through the system. ASP (Active Server Pages) is used to develop the system. Microsoft Access 2000 is used to store the student database.



# ACKNOWLEDGEMENT

I would like to thank my supervisor, Associate Professor Dr.P.Sellappan, who has given me helpful advice throughout my project. It is only with his encouragement and support that I am able to complete this project.

Further thanks to all the general staff at the Faculty of Computer Science and Information Technology, especially Miss Agnes, for their generous help wherever possible.

See Chun Kiat

Faculty of Computer Science and Information Technology

University of Malaya

October 2002

# TABLE OF CONTENTS

<b>ABSTRACT</b>	<b>I</b>
<b>ACKNOWLEDGEMENT</b>	<b>II</b>
<b>TABLE OF CONTENTS</b>	<b>III</b>
<b>LIST OF FIGURES</b>	<b>VIII</b>
<b>LIST OF TABLES</b>	<b>X</b>
<b>1.0 INTRODUCTION</b>	<b>1</b>
1.1 Introduction of the Project	2
1.2 Problem Statements	4
1.3 Mission	5
1.4 Objectives	5
1.5 Project Scope	6
1.6 Expected Outcomes	7
1.7 Report	7
<b>2.0 LITERATURE REVIEW</b>	<b>9</b>
2.1 Interactive Features Through Internet	9
2.1.1 HTML (Hypertext Markup Language)	9
2.1.2 Cascading Style Sheets (CSS)	9
2.1.3 Dynamic HTML	11
2.1.3.1 Data Binding	11
2.1.3.2 Fonts	12
2.1.4 Differences Between Netscape and Microsoft Version Of CSS and Dynamic HTML	12
2.1.5 Common Gateway Interface (CGI)	14
2.1.6 Hypertext Transfer Protocol	14
2.2 Web Application Development	15
2.3 Database Server	16
2.4 Database Management System (DBMS)	19
2.5 Data Access Interface Paradigm	20
2.5.1 ODBC Data Source Administrator	20
2.6 Client/Server Development	21
2.7 Case Studies	22
2.7.1 Case Study 1	22
2.7.2 Case Study 2	24
2.7.3 Case Study 3	25
2.7.4 Case Study 4	26
2.7.5 Case Study 5	27
2.7.6 Case Study 6	28



2.8 Database Consideration	29
2.8.1 Microsoft Access	29
2.8.2 Conclusion of The Database Consideration	29
2.9 System Application Development Tools Consideration	29
2.9.1 Microsoft Visual Basic 6.0	30
2.9.2 Visual C++	30
2.9.3 Active Server Pages (ASP)	31
2.9.4 Microsoft Personal Web Server	31
2.9.5 Microsoft Visual InterDev 6.0	32
2.9.6 Conclusion of Development Tools Consideration	32
2.10 Scripting Languages Consideration	34
2.11 Browser Consideration	35
<b>3.0 SYSTEM ANALYSIS</b>	<b>36</b>
3.1 Methodology	36
3.2 Project Schedule	39
3.3 Requirements Analysis	41
3.3.1 Functional Requirements	41
3.3.1.1 Students & Administrators Login Function	41
3.3.1.2 Change Password Function	41
3.3.1.3 Payment Calculating Function	42
3.3.1.4 Data Transaction	42
3.3.1.5 Data Queries	42
3.3.2 Non-Functional Requirements	43
3.3.2.1 User Interface	43
3.3.2.2 Storage Capacity	43
3.3.2.3 Easy to Use and Understand	44
3.3.2.4 Reliable	44
3.3.2.5 Robust	44
3.4 Run-Time Requirements	44
3.4.1 Window 95/98/2000/ME	44
3.5 Data Flow Diagram	46
<b>4.0 SYSTEM DESIGN</b>	<b>51</b>
4.1 Detail Design	51
4.1.1 Architectural Design	51
4.1.2 System Architecture	52
4.1.2.1 Data Architecture	52
4.1.2.2 Application Architecture	53
a) Main page module	53
b) Student login module	53
c) Student change password module	54
d) Student's personal information list module	54
e) Course selections module	54
f) Course selected module	54



g) Fees payment module	55
h) Timetable module	55
i) Administrator login module	55
j) Administrator change password module	55
k) View students' information module	56
l) Edit menu module	56
m) Edit students' information module	56
n) Edit administrators' information module	56
o) Edit registration fee and exam fee module	57
p) Edit MIT course information module	57
q) Edit MSE course information module	57
r) Edit MCS course information module	57
s) Edit MLIS course information module	57
t) Edit PHD course information module	57
u) Students access list module	58
v) Administrators access list module	58
w) Course registration reports module	58
 4.1.3 Database Design	 58
4.1.3.1 E-R Model	58
4.1.3.2 Data Dictionary	60
 4.1.4 User Interface Design	 64
4.1.4.1 Principles of Interface Design	65
4.1.4.2 Main Page Design	67
4.1.4.3 Course Registration Pages Design	68
4.1.4.4 Time Table Design	76
4.1.4.5 Administrators Pages Design	77
 <b>5.0 SYSTEM TESTING AND IMPLEMENTATION</b>	 <b>97</b>
5.1 Testing Strategies	97
5.1.1 Unit Testing	98
5.1.2 Module Testing	99
5.1.3 System Testing	100
 5.2 System Testing	 100
5.2.1 Main page module testing	101
5.2.2 Student login module testing	101
5.2.3 Student change password module testing	102
5.2.4 Student's personal information list module testing	102
5.2.5 Course selections module testing	102
5.2.6 Course selected module testing	102
5.2.7 Fees payment module testing	103
5.2.8 Timetable module testing	103
5.2.9 Administrator login module testing	103

5.2.10 Administrator change password module testing	103
5.2.11 View students' information module testing	104
5.2.12 Edit menu module testing	104
5.2.13 Edit students' information module testing	104
5.2.14 Edit administrators' information module testing	104
5.2.15 Edit registration fee and exam fee module testing	105
5.2.16 Edit MIT course information module testing	105
5.2.17 Edit MSE course information module testing	105
5.2.18 Edit MCS course information module testing	105
5.2.19 Edit MLIS course information module testing	105
5.2.20 Edit PHD course information module testing	106
5.2.21 Students access list module	106
5.2.22 Administrators access list module	106
5.2.23 Course registration reports module	106
5.3 System Implementation	107
5.3.1 Online Postgraduate Course Registration and Payment System Main Page	108
5.3.2 Course Registration	109
5.3.2.1 Change Password	110
5.3.2.2 Student Login	112
5.3.2.3 Student Personal Information	115
5.3.2.4 Course Selections	117
5.3.2.5 Selected Course(s)	118
5.3.2.6 Payment	119
5.3.3 Time Table	119
5.3.4 Administrator Login	121
5.3.4.1 Change Password	122
5.3.4.2 Administrator Login	123
5.3.4.3 View Student's Record	124
5.3.4.4 Edit Menu	126
5.3.4.4.1 Edit Students' Information	127
5.3.4.4.2 Edit Administrators' Information	130
5.3.4.4.3 Edit Registration Fee and Examination Fee	132
5.3.4.4.4 Edit MIT Course Info/ Edit MSE Course Info/ Edit MCS Course Info/ Edit MLIS Course Info/ Edit PHD Course Info	133
5.3.4.4.5 Students Access List	137
5.3.4.4.6 Administrators Access List	137
5.3.4.4.7 Course Registration Report	138



<b>6.0 SYSTEM EVALUATIONS AND CONCLUSIONS</b>	<b>142</b>
6.1 Strengths	142
6.1.1 Simple and Easy to Maintain	142
6.1.2 Low Cost of Ownership	142
6.1.3 High Level of Upgrade	142
6.1.4 Navigation of the System	143
6.2 Limitations	143
6.2.1 Stability	143
6.2.2 Low Recoverability of Stored Data	143
6.2.3 Manual Backup of Database	143
6.2.4 Storage Capability	144
6.2.5 Interface Design	144
6.2.6 Security	144
6.3 Future Enhancements to Upgrade the Quality of the System	144
6.3.1 Security Features	145
6.3.2 Designing and Deploying Web-Enabled Applications: A Changing Architecture for Changing Needs	146
6.3.3 Good Quality of Performance Database Server	148
6.3.4 Stable Web Server	148
6.3.5 Implement Firewall	148
6.3.6 Cisco	149
6.4 Conclusion	149-150

## GLOSSARY

## REFERENCES



## LIST OF FIGURES

Figure 1 When accessing database information from a Web browser, the Web Server acts as the client to the database server	18
Figure 2 Prototyping Method	38
Figure 3 Context Level Diagram For An Online Postgraduate Course Registration and Payment System	46
Figure 4 Diagram 0 For An Online Postgraduate Course Registration And Payment System	48
Figure 5 Diagram 0 (Continue)	49
Figure 6 E-R Diagram for Online Postgraduate Course Registration And Payment System	59
Figure 7 Main Page Design	67
Figure 8 Student Login Page Design	68
Figure 9 Change Password Page Design	69
Figure 10 Successful Login Page Design	70
Figure 11 Student Personal Information List Page Design	71
Figure 12 Course Selections Page Design	72
Figure 13 (Continue from Figure 12)	73
Figure 14 Selected Course(s) Page Design	74
Figure 15 Payment Page Design	75
Figure 16 Timetable Selections Page Design	76
Figure 17 MCS Time Table Page Design	77
Figure 18 Administrator Login Page Design	78
Figure 19 Successful Login Page Design	79
Figure 20 View Student' Record Page Design	80
Figure 21 Student's Record Page Design	81
Figure 22 Edit Menu Page Design	82
Figure 23 Edit Student's Record Page Design	83
Figure 24 Student's Record Page Design	84
Figure 25 Add New Student Page Design	85
Figure 26 Edit Administrators' Records Page Design	86
Figure 27 Edit/Update Record Page Design	87
Figure 28 Edit Registration Fee and Examination Fee Page Design	88
Figure 29 Edit MIT Course Page Design	89
Figure 30 (Continue from Figure 29)	90
Figure 31 Update Course Information Page Design	91
Figure 32 Students' Login Time	92
Figure 33 Administrators' Login Time	93
Figure 34 Course Registration Report for FSKTM	94
Figure 35 Report for MCS	95
Figure 36 Students that Register Course WRGA6302	96

Figure 37: Online Postgraduate Course Registration and Payment System Main Page	108
Figure 38: Student Login	110
Figure 39: Change Password	111
Figure 40: Password Successfully Changed	112
Figure 41: Student Login Page	113
Figure 42: Successful Login	114
Figure 43: Student Personal Information List	115
Figure 44: Course Selections	116
Figure 45: (Continue from Figure 44)	117
Figure 46: Selected Course(s)	118
Figure 47: Payment Information	119
Figure 48: Time Table	120
Figure 49: Time Table for Master of Computer Science	121
Figure 50: Administrator Login	122
Figure 51: Administrator Login	123
Figure 52: Successful Login	124
Figure 53: View Student's Record	125
Figure 54: Student's Record	126
Figure 55: Edit Menu	127
Figure 56: Edit Student's Record	128
Figure 57: Student's Record	129
Figure 58: Add New Student	130
Figure 59: Administrators' Records	131
Figure 60: Edit/Update Record	132
Figure 61: Edit Registration Fee and Examination Fee	133
Figure 62: Edit MIT Course	134
Figure 63: (Continue from Figure 62)	135
Figure 64: Edit Course Information	136
Figure 65: Students Login Time	137
Figure 66: Administrators Login Time	138
Figure 67: Course Registration Report for FSKTM	139
Figure 68: Report for MCS	140
Figure 69: Students that Register Course WRGA6302	141
Figure 70 The evolution of Web architecture to support the basic needs Of a Web-enabled application	147



## 1.0 INTRODUCTION

### LIST OF TABLES

Table 1 Project Schedule	40
Table 2 UM Administrators Information	60
Table 3 FSKTM Postgraduate Students Information	61
Table 4 Courses Particulars	61
Table 5 Course(s) Selected Information and Payment Information	62
Table 6 Registration Fee & Examination Fee Per Credit Hour	63
Table 7 Students' Login Time	63
Table 8 Administrators' Login Time	64



## 1.0 INTRODUCTION

From E-Connection (Internet) to E-Contents (Web Portals) to E-Commerce, the world is witnessing successive milestones of ongoing digital revolution - the former being pre-requisite of the latter. The digital revolution leading to information society is silently but surely triggering transformation of social infrastructure within the nation for its socio-economic development as well as acting as a vehicle to integrate itself as a part of global networked/digital economy. (Arvind Panagariya, 1999)

While there is no single globally accepted definition of E-Commerce, it is gradually leaning towards "goods and services transacted over Internet". E-Commerce is thus a new way of conducting, managing and executing business transactions using modern Information Technology. The Internet provides access 24 hours a day, seven days a week – any time anywhere. Thus, time and place are no longer the binding factors. Electronic Commerce builds on the structures of traditional commerce by adding the flexibility offered by electronic networks. This facilitates improvement in operations leading to substantial cost savings as well as increasing competitiveness and efficiency through the redesign of traditional business. (Kalakota, Ravi & Whinston A, 1996)

Payment gateways are credit card processing services that allow organizations to authorize and process credit card orders online in real time. Payment gateways act as a

bridge between the merchant's website and the financial institutions that process the transaction. Payment gateways are an integral part of e-commerce. Existing brick and mortar companies may find that they can target a new market by selling online. Whether or not a business becomes more competitive by providing online sales will depend on the marketing mix, product, place, promotion and price. The marketing mix will help to determine whether the particular business will be competitive. (Kathy Brown, 2000)

Online course registration meaning students can register for classes online via a custom, Web-based registration application. Students can view and choose the list of all the online courses offered each semester.

## **1.1 Introduction of the Project**

Online course registration and payment system is yet to apply in University of Malaya (UM), and other local universities in Malaysia as well. But, it is popular in European countries.

Online payment means using credit card (visa or master card) to pay fees for the course they have registered online. But, students can also directly pay cash to UM Bursar's Office after they have registered their course online. All fees are payable after online course registration and registration is not complete until these fees have been paid. Fees are subject to change prior to each semester. All the payment record



such as mode of payment, credit card type, credit card number, expire date of the credit card, name of the card holder and address of the card holder will be stored in the database.

UM Bursar's Office will process students' credit card information immediately for an authorization after the students have made the online course registration. This means that UM Bursar's Office will contact the bank electronically and verify that credit card information students have provided matches the credit card information where their credit cards are mailed. If the information they have provided is different from information that bank has provided to UM Bursar's Office, the course registration will be delayed or canceled. UM Bursar's Office will resolve problems that may arise from the address verification process by contact the bank or the student. UM Bursar's Office will contact the local police department to report all fraudulent fees payment.

Online Postgraduate Course Registration and Payment System is implemented for the postgraduate students in Faculty of Computer Science and Information Technology (FSKTM) in University of Malaya. This project is done for postgraduate students in Faculty Computer Science & Information Technology, as the scope is too broad if it is expanded for all students in UM. The system will cater for students in Master of Information Technology, Master of Computer Science, Master of Library and Information Science, Master of Software Engineering, and PhD's.



Besides registering online for their courses and make payment with this system, students can also view their latest timetable for the semester. For administrators, they can view, delete and update students' record, course information, administrators' information and registration report. User ID and password are needed for both students and administrators to access into the online course registration and administrator site.

## 1.2 Problem Statements

The current system of course registration and fees payment in UM shows inefficiency in the method of course registration and fees payment. UM allocates specific windows of time each semester for different populations of students to register for classes. The largest window is for the courses that are popular, and it accommodates nearly 400 to 1000 students trying to register simultaneously for that particular course. Therefore, every time during the registration period, course registration application will become overloaded, and some students experienced delays in registering for courses.

In addition, students have to wait in a long queue at the bank to pay their fees. After that, they have to go to UM Bursar' Office to submit the bank statement to prove that they have paid the fees. The students have to queue up again.

❖ Online course registration and fees payment will lessen queuing problem in

Then, students have to endure another queue to submit their course registration form in the faculty's office. On top of that, if the students want to change courses, UM staff will need to do a lot of work like checking all the information in the forms is correctly written by the students, check the redundancy of courses that register by students. After that, the staff will submit all the forms to the person in charge in order for him to record the new information into the database.

rather than paying in bank or UM Registrar's Office.

With the Web-based registration application, these problems are resolved. The students can access to the Web site to pay fees online and register their courses online.

The scopes of the project are as follows:

➤ Develop a web application system for FSKTM postgraduate students to

### **1.3 Mission**

The purpose of Online Postgraduate Course Registration and Payment System is to overcome the queuing problem and paperwork during course registration. It will facilitate faster course registration and payment process.

➤ Provide an administrative facility where UM administrators can view,

delete and update students' records, course information as well as

students' records on the web.

### **1.4 Objectives**

This project has the following objectives:

- ❖ Online course registration and payment system can save students' and UM administrators' time in course registration and fees payment.



- ❖ Online course registration and fees payment will lessen queuing problem in Administrators Office, Bank and UM Bursar's Office.
- ❖ Online course registration can avoid unnecessary hassle during course registration and increase the effectiveness of course registration process.
- ❖ Online course registration will decrease the workload of the UM administrators and staffs during and after the registration.
- ❖ Online payment using credit card will make the fees payment process faster rather than paying in bank or UM Bursar's Office.

## 1.5 Project Scope

The scopes of the project are as follows:

- Develop a web application system for FSKTM postgraduate students to register their course and pay fees online.
- Implement course fees calculation facility for students during the process of course registration in the system.
- Provide the latest online timetable for students.
- Provide UM an administrative facility, where UM administrators can view, delete and update students' records, course information as well as administrators' records on the web.
- UM administrators can see the registration report after students register online.

## 1.6 Expected Outcomes

The final outcome of using this system is expected to have the following features.

- A web application system that can register students' course and pay fees online.
- A web application system that enables UM administrators to view, delete and update course information, student records, and administrators information.
- A database system that will store and organize all records pertaining to the registration, such as the students personal information, administrator records, course registered by students, fees, and payment information.
- Automatic calculating facility for students to find out how much they need to pay.
- UM administrators can check the registration report after students register their course online.

## 1.7 Report

The purpose of this report is to document all the essential information gathered and used to develop this system. It covers several instructional design phases that include analysis, design, development, testing, implementation, and evaluation phase of the system. This report is divided into 6 chapters. A brief synopsis of each chapter is listed below.



**Chapter 1: Introduction** gives overview of developing the system, problem statements, project mission, objectives, project scope, and expected outcomes of the system.

**Chapter 2: Literature Review** serves to build up the basic knowledge of the setting up of the whole system, with the involvement of various softwares. For a better knowledge of developing the system, research of case studies also have been done. Software preference is also discussed, and comparison between softwares is done to choose the most suitable one.

**Chapter 3: System Analysis** focuses on the methodology of developing the system, project schedule, requirement analysis, where functional and non-functional requirements are described, run time requirement and also data flow diagram (DFD).

**Chapter 4: System Design** focuses on architectural design, system architecture, database design, and user interface design.

**Chapter 5: System Testing and Implementation** stresses on the implementation part and system testing.

**Chapter 6: System Evaluations and Conclusions** begin with various evaluation techniques. Technical problems, recommended solutions and future enhancement are also discussed here.

## **2.0 LITERATURE REVIEW**

### **2.1 Interactive Features Through Internet**

#### **2.1.1 HTML (Hypertext Markup Language)**

HTML is by far the most common document format on the World Wide Web. HTML is a form of SGML (Standard Generalized Markup Language), which is primarily concerned with structure, rather than layout or presentation styles. (Nadav Savio, 1997). Thus, though widely used, HTML does not support most of the page layout features found in desktop publishing applications. It does include tags that provide for some forms of presentation, such as tables, and tags to specify fonts and colors. However, these tags are bulky and require extensive coding. A major complaint against HTML is that it does not allow the precise positioning of text and images, as required in desktop publishing. Authors and publishers want greater control over layout so that they can create sophisticated publications that retain the same layout and look regardless of the configuration of a user's browser. Two developments, Cascading Style Sheets and dynamic HTML, have been designed to address these limitations of HTML. (Laura Lemay, 1996)

#### **2.1.2 Cascading Style Sheets (CSS)**

Cascading Style Sheets (CSS) address the problem of layout control, putting the power of desktop publishing into the hands of Web publishers. Authors can create pages that display precise margins, various font styles, measured white space around text and



graphical elements, and other desktop publishing-like page formatting controls. In addition to helping authors create the exact layout they want for a page, Cascading Style Sheets allow them to apply a universal look and feel over an entire Web site without the work of inserting new tags and templates into each Web page. (Jeff Kozoris, 1997)

#### Features and advantages of Cascading Style Sheets:

- Provides precise control over margins, line spacing, element placement, colors, font faces, and font sizes
- Removes the need to re-type HTML style tags each time a new style is needed
- Ensures every user sees the same view regardless of the ways in which the browser's size and colors are configured
- Provides the ability to change the overall look of a Web page or even an entire site by changing a single style sheet.

Alternatively, style information can be contained in a separate, single file to which all relevant HTML files refer. Thus, when a change to a Web site's overall look and feel is desired, changes need only be made to the style sheet file. This aspect is a significant development for those who maintain extensive Web sites. Presently, a change in look and feel across a site requires that each HTML file be opened and the code modified. With CSS information in a separate file, the code changes take place once in a single file. (Que Corporation, 1997)

### 2.1.3 Dynamic HTML

Dynamic HTML is a combination of Cascading Style Sheets and a scripting language (such as Visual Basic script or JavaScript) that merges the actual HTML document with the style sheet. Using Dynamic HTML, developers will be able to create a Web page that can respond dynamically to user-generated events, such as mouse clicks, by expanding an HTML table, changing the attributes of a font, or moving a graphic across a page. All of this can be done without going to the server or relying on plug-ins or Java applets. (Levitt, J, 2000)

#### 2.1.3.1 Data Binding

One of the key features of dynamic HTML is that the browser does the processing of effects locally, putting less strain on the server and speeding up load time. A very powerful example of this feature is "data binding." Currently, when a user interacts with a database, data are pulled from a system; tagged on the fly with HTML, then send to the user. If a user wants a different view of the data -- for example, sorted by author or by date -- he or she must make another call to the server which sorts the data, reformats it in HTML, and resends the data to the user. With the data-binding feature of dynamic HTML, all relevant data are sent to the user to be sorted, filtered, and modified repeatedly, without having to contact the server again. (Levitt, J, 2000)



### **2.1.3.2 Fonts**

In both desktop and Web publishing, the availability of fonts is a major issue. Currently, Web authors are restricted to using fonts that are commonly present on users' systems. A feature of dynamic HTML is the ability to use virtually any font on a Web page without the need for end users to have it present locally. This feature allows a Web page to retain its intended look when a certain font is specified. The text will not revert to New Times Roman if the font is not present, as is currently the case with standard HTML.

### **2.1.4 Differences Between Netscape and Microsoft Version of CSS and Dynamic HTML**

Netscape Communicator 4.0 supports Level 1 version of CSS (CSS1) under the name "JavaScript Accessible Style Sheets" (JASS), while limited CSS support is now available in Microsoft's Internet Explorer 3.0, to be known as "Dynamic HTML." Both the Netscape and Microsoft versions are recognized under the generic name of "dynamic HTML," with the lower case 'd.' In addition, each company has its own definition of dynamic HTML.

Microsoft defines DHTML as the HTML document object model a way to control the positioning of elements on a page, a set of multimedia controls for animation, alpha-channel filtering, and other effects and the ability to bind sets of data to an HTML page. (Zelnick, N, 1997)

### 7.1.3 Common Gateway Interface (CGI)

Netscape, on the other hand, defines DHTML as HTML plus Netscape extensions, JavaScript, Java, and an object model for HTML documents. (Zelnick, N, 1997)

It is very important to note that CSS1 has not yet been coalesced into a common standard. As with earlier versions of standard HTML, Netscape and Microsoft have created their own, idiosyncratic features of dynamic HTML. Both have put forward these ideas to the W3C to be considered for the standard. This divergence causes problems: if these two major companies are developing different forms of dynamic HTML, elements of CSS Web will be browser dependent. This erodes the cross-platform compatibility that has allowed the Web to grow as wildly as it has.

Finally, dynamic HTML is supported only by Netscape Navigator and Microsoft Internet Explorer. There is currently no support for dynamic HTML in older Netscape and Microsoft browsers or browsers of a different type. When non-compatible browsers encounter dynamic HTML, all the information will be displayed, but without the new features. Without a common standard, authors face the possibility of creating three versions of their pages, one with Microsoft Dynamic HTML, one with Netscape JASS, and one without dynamic HTML at all.



basis for information exchange on the Internet), HTTP is an application protocol. Essential concepts that are part of HTTP include (as its name implies) the idea that files can contain references to other files whose selection will elicit additional transfer requests. Any Web server machine contains, in addition to the HTML and other files it can serve, an HTTP daemon, a program that is designed to wait for HTTP requests and handle them when they arrive. Web browser is an HTTP client, sending requests to server machines. When the browser user enters file requests by either "opening" a Web file (typing in a Uniform Resource Locator) or clicking on a hypertext link, the browser builds an HTTP request and sends it to the Internet Protocol address indicated by the URL. The HTTP daemon in the destination server machine receives the request and, after any necessary processing, the requested file is returned.

## **2.2 Web Application Development**

Developing Web-based applications can be very different from other types of programming. Unlike normal client/server applications, web pages are stateless, which means that when the page is called, it grabs the data it needs, and then sends the document to us. A system that developed using Web-based applications will involve many interactive features such as insert, delete, and add features, dynamic database, shopping cart, visa or master payment gateway and others. (Eric A. Smith, 2000)

In traditional client/server development, a client application maintains a connection with its server component. Besides requesting and sending data, the client queries the

server periodically to make sure the connection is still alive. If the server goes down, it will sense this and take the appropriate measures such as sending an error message to the user.

The application will have three components: the requesting browser, the Web server, and the database server (see Figure 1). The browser is responsible for submitting query requests and displaying the results from the database. The Web

As a Web application developer, I have to deal with the short-lived memory of web servers, with the Internet being a connectionless network. When a browser requests a page, it is serve up and serve out. The server is not going to check and see whether the page and all of its components (graphic, sound and so on) are on the browser.

But ASP (Active Server Pages) and ADO (ActiveX Data Objects) are significant steps in dealing with these issues. It has a memory to “remember” user.

## 2.3 Database Server

The database server plays a vital role in Internet application development. The database server can be used to store, search, and retrieve information that is stored in a database. This same database that distributes information to Web users can also be accessed and maintained from or within the corporate walls. (Eric A. Smith, 2000)

When using the HTTP server as a connection utility to database servers, there is a slight shift in architecture that differs from traditional application development. For



example, when creating an online classified section for the local newspaper or mechanism that search for vehicles of a particular type. From an architecture perspective, this application will have three components: the requesting browser, the Web server, and the database server (see Figure 1). The browser is responsible for submitting query requests and displaying the results from the database. The Web server is responsible for the database, formatting the results into HTML, and delivering the HTML to the requesting browser. The database server is responsible for accepting requests from the Web server and delivering results back to the Web server.

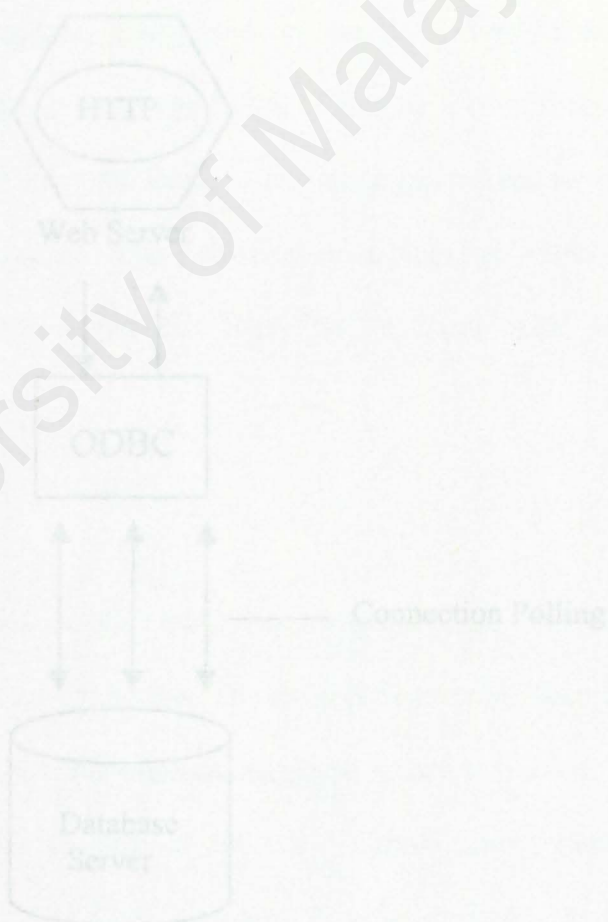
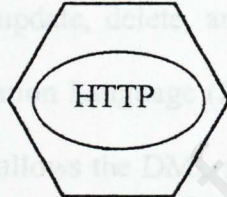
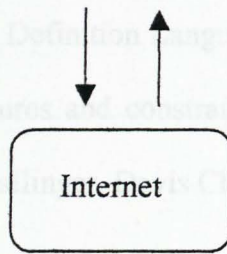


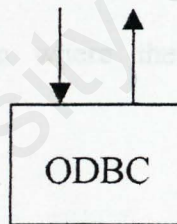
Figure 1: When accessing database information from a Web browser, the Web server acts as the client to the database server



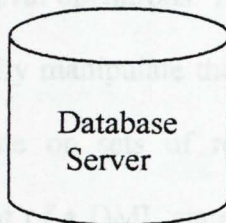
Web browser



Web Server



ODBC



Database  
Server

Connection Polling

**Figure 1: When accessing database information from a Web browser, the Web server acts as the client to the database server**



## 2.4 Database Management System (DBMS)

DBMS is the software that interacts with the user's application program and the database. Typically, a DBMS provides several facilities. It allows users to define the database, usually through a Data Definition Language (DDL). The DDL allows users to specify the data types, structures and constraints on the data to be stored in the database. (Scot Johnson, Keith Ballinger, Davis Chapman, 2000)

DBMS allows users to insert, update, delete, and retrieve data from the database, usually through a Data Manipulation Language (DML). Having a central repository for all data and data description allows the DML to provide a general inquiry facility to this data, called a query language. The provision of a language alleviates the problems with file-based system where the user has to work with software management problems.

There are two types of DML, such as procedural and non-procedural, which we can distinguish according to the retrieval operations. The main difference between them is that procedural languages typically manipulate the database record by record. While non-procedural languages operate on sets of records. Consequently, procedural languages specify how the output of a DML statement is to be obtained, while non-procedural DMLs describe only what data is to be obtained. The most common type

of non-procedural language is the Structure Query Language (SQL). SQL is now both the standard and the *de facto* language for DBMS.

DBMS also provides controlled access to the database. For example, it may provide:

- A security system, which prevents unauthorized users from accessing the database.
- An integrity system, which maintains the consistency of stored data.
- A concurrency control system, which allows shared access of the database.
- A recovery control system, which restores the database to previous consistent state following a hardware or software failure.
- A user-accessible catalog, which contains description of the data in the database.

## **2.5 Data Access Interface Paradigm**

### **2.5.1 ODBC (Open Database Connectivity) Data Source Administrator**

ODBC is a programming interface that enables application to access data in database management system that uses Structure Query Language (SQL) as a data access standard. There are three major data sources in ODBC Data Source Administrator: System Data Sources, User Data Sources and File Data Sources. (Scot Johnson & Keith Ballinger, 1997)



In this project, I am using an ODBC File data source. An ODBC File data source allows us to connect to a data provider. The reason that I use File DSN is because it can be shared by users who have the same drivers installed.

On the other hand, Microsoft Access Driver has been used in the ODBC Data Source Administrator. Microsoft Access Driver is an ODBC driver that allows ODBC-enabled programs to get information from ODBC data sources.

## **2.6 Client/Server Development**

When developing a database-driven application, it will be a very rare case where many users do not access the database at the same time over a network or the Internet. This type of application is called client/server, and it refers to a system in which there is a client program that handles some tasks, and a server program that handles others. The Web is an example of a client/server application. The Web server handles certain tasks, such as finding the appropriate page, while the browser displays the page contents. With client/server database programming, there is, at the very least, a server where the database resides, and several client machines on which a client program is used to access the database server. (Scot Johnson & Keith Ballinger, 1997)

## **2.7 Case Studies**

### **2.7.1 Case Study 1**

From the article written by Andrew Aird (2000), "E-commerce in Higher Education: Can We afford to do nothing?" argues that the impact of e-commerce is beginning to

When developing client/server systems, one must consider the tasks and processes that are performed by the server and the client. The very best application is a marriage of tasks between these two components.

Some of the tasks best suited for the client are as follow:

- ❑ Display of result sets from the database
- ❑ Simple data validation
- ❑ Gathering data before querying the database
- ❑ Formatting and filtering result sets form the database

Meanwhile, the server is often best used for these tasks:

- ❑ Complex data validation
- ❑ Indexing data
- ❑ Finding and returning result sets to the client
- ❑ Maintaining database integrity
- ❑ Managing transactions

## 2.7 Case Studies

### 2.7.1 Case Study 1

From the article written by Andrew Aird (2000), “*E-commerce in Higher Education: can we afford to do nothing?*” argues that the impact of e-commerce is beginning to



be felt in the non-commercial sector, and reveals the results of a survey of opinion taken after a recent web management workshop. With e-commerce emerging as the driving force behind commercial websites, its impact is beginning to be felt in the non-commercial sector. Those working in Higher Education, in particular, need to prepare carefully if we are to reap the rewards and avoid the pitfalls by establishing appropriate 'business' structures as well as refining technical and managerial approaches. Survey of the 76 Higher Education organizations who responded over 80% claim to have some kind of e-commerce system in place now or plan to by the middle of 2001. (E-commerce here is defined as having an online system for ordering and paying for goods or services, although the method of payment may not necessarily be by credit/debit card.) In fact, 70% of the institutions surveyed will continue to use paper-based systems (such as invoice generation, follow-up cheque) rather than online transfer of funds. (Andrew Aird, 2001). Selling publications, payment of fees and alumni goods make up the largest categories of transactions (74%). Other uses included supplying advice and information (5%), sales of concert/theatre tickets and printing/photocopying charges. In many cases it was web staff (60%), although academic departments themselves show significant involvement (13%). Perhaps surprisingly, Finance Departments appeared to be involved in very few cases (6%). Only one institution surveyed had a project officer dedicated solely to e-commerce development. The survey also looked briefly at the online shopping habits of the individual respondents, with the aim of seeing how much our consumer experience might affect our approach as providers. Almost all (98%) of those who had bought goods or services online were satisfied, and most (92%) regarded this as a safe method of making payments. Incidentally, the most popular sites used were Amazon (62%),

Easyjet (11%) and Go-fly, Expedia and Tesco equal in third place (4%). The survey contrasts sharply with figures given at the Parallel Session by John Williams, CEO of Trustmarque: Taking a world view, 25% of e-commerce transactions are never fulfilled, and 53% of online customers are concerned about security and privacy. Furthermore 'conversion rates' of visitors to buyers currently run at about 1.8%. Evidently the experience of e-commerce in Higher Education is more positive, and this may be an important starting point when planning or running such a system. The 'brand' of Higher Education Institutions (HEIs) already has much global credibility – no one suspects that HEIs is going to steal your identity or defraud your credit card, and they already have a reputation for providing 'quality'. This may help to reassure users in the future who may be paying substantial course fees online, or signing up to an expensive distance learning.

### **2.7.2 Case Study 2**

A study to an article on web site <http://www.oar.uiuc.edu/current/Default.htm/> shows that University of Illinois at Urbana-Champaign was implemented the UI Direct system for their students to register courses online. The UI Direct system will control access based upon registration time, class size, academic program, maximum credit limit, advising hold (if applicable), and encumbrances. This system has some drawback although the university's students can register their course online. For instance, UI Direct Assistance is available only from 8:30 a.m. until 5:00 p.m. Monday-Friday at the Records Service Center. If an advising hold has been placed on student's record, it must be removed before he/she can register. The students should



resolve all encumbrances as soon as possible. If they have an encumbrance on their record, access to UI Direct to drop or add classes will be blocked approximately three weeks prior to the first day of instruction. If the student has an encumbrance on their record as of 5:00 p.m. on the first day of instruction, their course registration for the term will be cancelled. The student can avoid having their course registration cancelled by resolving encumbrances. Besides that, using this system will facing the problem of fees payment by the students. It is because this system does not have the online payment gateway using visa or master card. UI Direct is a system to register for courses by computer. When the students do so, they agree to pay tuition and fees to the University according to the payment policies and schedules adopted by the Board of Trustees. If they wish to cancel their registration, and thus avoid payment of fees charges, they must do so by 5:00 p.m. of the first day of instruction. Furthermore, the lack of securities of the system is found. Unauthorized use of University of Illinois computer systems, data, or resources; unauthorized use of another individual's identification, account, or password; or an attempt to gain unauthorized access is prohibited by University policy and may constitute a violation of Illinois state law. Access to U of I Direct will be terminated if students are found to be making excessive unsuccessful registration attempts.

### **2.7.3 Case Study 3**

Another study that has been done by Suffolk County Community College (SCCC) at <http://www.sunysuffolk.edu/Web/VirtualCampus/> shows that this college is implementing two kinds of course registration and payment system, such as online

registration and payment system or by using the SCCC telephone registration and payment system. If the student wishes to use the telephone registration system, they need to gather the following information: the four-digit section number of the course they wish to take, the name of the campus offering the course, the semester that the course is being offered (use 1 for intersession, 2 for spring, 6 for summer, 9 for fall) and the nine-digit student ID. If the student registers their course online, during the first week of classes they have the opportunity to make changes to their schedule, which will not appear on their transcript. Students can accomplish this by going to the online registration and system or by calling the SCCC telephone registration system.

#### 2.7.3 Case Study 3

#### 2.7.4 Case Study 4

A study that have been done by New Brunswick Distance Education Network Inc in Canada at <https://www.sis-sia.gnb.ca/test/sis/> shows that before the organization/institution begins using the TeleCampus online registration and payment system (SIS - Student Information System), the organization/institution must be capable of accepting credit card payments. The organization/institution has to make sure that their business accounts are compatible with their PAYWAY Service. Once their business accounts are ready to accept credit card transactions, TeleEducation NB's TeleCampus SIS admin via email and request access to the system. At that point, the organization/institution will be configured in the Test SIS System. If they are not a NB Community College, an institution number will be automatically assigned (by the system) and the courses given course codes. These are used by the system to track the courses and students. Once the organization/institution supplied TeleCampus with the



information needed to track the courses and accept online payment, then they must configure their website to forward registration requests to the SIS. To do this, they simply must embed the application button or link in their page. Each application link must include the respective course code to TeleEducation. After a short testing period and approval of the registration process, TeleEducation will move the course information into the production SIS system. The testing, approval and moving phase may be as short as two days, or as long as they require. Organization/institution is now ready to accept online course registrations and payments.

#### **2.7.5 Case Study 5**

Students at Duke University's undergraduate, graduate, and professional schools register for classes online via a custom, Web-based registration application. Duke. (Cisco System, 2002). During registration for the spring semester 2001, Duke's online course registration application became overloaded, and some students experienced delays in registering for courses. At that time, the Web servers handling the registration application received requests through a round-robin Domain Name System (DNS) rotary. This system provided no mechanism to intelligently balance the workload and could not compensate for a "sick" or "dead" server. As a result, groups of students could be routed to a server that was performing slowly or was completely unresponsive. Duke had to stabilize the online registration system to increase performance and guarantee that every student would be able to register in a timely manner while also ensuring that each student's private data was protected. Duke University implemented the Cisco 11000 Series Content Services Switch (CSS) and

the Cisco 11000 Series Secure Content Accelerator (SCA) to enhance the performance of its online course registration application, guaranteeing 100 percent server availability and ensuring secure and successful online registration.

### **2.7.6 Case Study 6**

In the article that wrote by Steve Patient, “ Reducing Online Credit Card Fraud ”, credit card company figures show that 90 per cent of consumers are reimbursed when their cards are used fraudulently, while 75 per cent of online retailers have to eat the cost when they're the victims of credit card fraud. There is currently no way to avoid this ludicrously high risk, though it can be reduced. (Steve Patient, 2002). Unfortunately, online payment remains a major area of Internet immaturity. Payment and data transfer security are allied problems. When buyer and seller meet physically to exchange money for goods, trust is less of an issue than when two entities deal blind online. Though buyers - rightly - distrust online credit card payments, merchants suffer more from credit fraud. This is because most online payment is by credit or debit cards, and the payment protocols for these were originally intended for face-to-face sales where the cardholder and card are both physically present. Physical presence offers security based on a customer signature and card imprint. But the merchant is almost always responsible for losses when sales are made on a 'Cardholder Not Present' basis even when the vendor has obtained authorization from the card issuer.



## **2.8 Database Consideration**

### **2.8.1 Microsoft Access**

Microsoft Access is a relational database management system and can be used in a client/server or an n-tier architecture system. It provides intuitive and user-friendly interface to create a database easily.

### **2.8.2 Conclusion of The Database Consideration**

For database development in this project, Microsoft Access 2000 will be chosen, as it has an easy-to-use visual interface. It enables users to group and sort data. It also can be upsized to SQL database when necessary in the future. Microsoft Access has a simple and complete development environment. Furthermore the cost is much lower compared to SQL Server 7.0. Therefore, using Microsoft Access will significantly reduces the development cost.

## **2.9 System Application Development Tools Consideration**

After choosing the database (Microsoft Access 2000) that will be used in the system, the development tools for developing this system are considered.

### 2.9.1 Microsoft Visual Basic 6.0

Visual Basic 6.0 is a high-level visual programming version of BASIC. It is able to create a wide variety of component types, including Microsoft ActiveX dynamic link libraries (DLLs), ActiveX Controls and ActiveX Components for deployments in Intranets or the Internet. (Eric A.Smith, 2000)

### 2.9.2 Active Server Pages (ASP)

ASP means is Web-based application development. ASP is a new technology from

Visual Basic has a native code compiler and offers the ability to develop ActiveX controls and DLLs; it is a good choice for just about any type of Windows application development. Visual Basic is particularly strong when it comes to file processing and the development of a database front end.

ASP provides a powerful interface to develop programming logic that can be used to implement and distribute the information through applications across the Web. (Eric A.Smith, 2000)

Visual Basic is probably the easiest development tool to use. The environment makes it very easy to build a user interface, and it is easy for just about any programmer to pick up. It is very strong for building database application, especially those that use Microsoft Access database.

### 2.9.2 Visual C++

Visual C++ is a good choice for building scientific and engineering applications. This is because it still has a large speed advantage over Visual basic when performing these types of tasks. One drawback is that designing a user interface with Visual C++ is not as easy, and developer has to know the C/C++ language in order to use it. Even with



native code compilation, for some application, Visual C++ will still have a hefty speed advantage compare with Visual Basic. Visual C++ has more multithreading ability than Visual Basic, which still remains rather limited in this area. (Eric A.Smith, 2000)

### 2.9.3 Active Server Pages (ASP)

ASP means to Web-based application development. ASP is a new technology from Microsoft that provides the capability for the Web server to process application logic and then delivers standard HTML to the client browser. The results can then be delivered to a variety of client-side Web technologies, such as standard HTML, ActiveX, Java, browser plug-ins and DHTML. In addition, ASP provides a powerful interface to develop programming logic that can be used to implement and distribute the information through applications across the Web. (Eric A.Smith, 2000)

### 2.9.4 Microsoft Personal Web Server

Microsoft Personal Web Server (PWS) is the answer to the personal information sharing and Web development needs. PWS is a desktop Web server that makes Web site setup quick and simple, from creating a personalize home page automatically, to drag-and-drop publishing of documents. (Eric A.Smith, 2000)

On the corporate intranet, Personal Web Server can be used to quickly share documents in their native format, or convert documents to HTML, and then use PWS to share them across different operating systems.

Because Personal Web Server supports Active Server Pages, it can be used as a development and testing platform for Web sites. Create the web site in the office or at home, and test it by using Personal Web Server, before hosting it on the corporate server or an Internet service provider.

### **2.9.5 Microsoft Visual InterDev 6.0**

Microsoft's Visual InterDev is a development environment in which can create, edit, deploy and manage Active Server Pages. Visual InterDev combines a rich set of database connectivity tools, wizards, and design-time controls to increase the functionality and decrease the development time to build Active Server Applications.

The functionality of Visual InterDev can be categorize into four sections:

- ❖ File and source code management
- ❖ Database connectivity and live-design-time access
- ❖ Active Server Page functionality
- ❖ Client-side functionality

### **2.9.6 Conclusion of Development Tools Consideration**

After surveying the above development tools, I decided that Microsoft Visual Basic 6.0 and Visual C++ is not suitable to use in this project. It is because Microsoft Visual



Basic 6.0 and Visual C++ is not suitable for web application development although it provides a flexible, extensible and friendly development environment to develop database application. Microsoft Visual Basic 6.0 and Visual C++ are only suitable for window application development. But, online postgraduate course registration and payment system is a web application system development.

The reasons I choose to use ASP because it is suitable for existing application developers who have some experience developing client/server applications. Therefore, the application developers can translate their existing experience developing applications into building robust, scalable, Web-based applications using ASP. Besides, ASP gives the ability to deliver more than just HTML. ASP can deliver HTML, client-side scripting, Web controls, and server-side processing and connectivity features. ASP scripts can deliver client-side scripts, such as **VBScript** and **JavaScript**, to be executed on the client's browser. In addition, ASP can also delivers wide range of Web functionality by acting as a transfer vehicle for ActiveX controls, Java applets and other Web components. ASP not only produce a dynamic HTML depending on the client request, but also provides the capability to tap into existing systems, such as databases, document retrieval services, mail servers, and other COM-based information servers.

ASP now act as a HTML interpreter that was once only accessible through native interfaces, such a Microsoft Exchange client, Lotus Notes client, or a customize Visual

Basic application. With the wide range of functionality, ASP act as a medium for porting existing applications to --- and building new applications for --- the Web.

Other than that, Microsoft Personal Web Server and Microsoft Visual InterDev 6.0 are also chosen to support my task.

## **2.10 Scripting Languages Consideration**

Since I decided to use ASP as my development tool, therefore VBScript and JavaScript will be the scripting languages in my project. However, VBScript will be the scripting language that I frequently use in this project.

### **2.11 Browser Consideration**

VBScript was initially created by Microsoft to be a lightweight scripting language to interpret users events triggered within the Internet Explorer browser. VBScript is actually created from Visual Basic for Applications (VBA), a pure subset of Visual Basic. VBScript can be used to create references to control HTML intrinsic object, ActiveX automation objects, ActiveX controls, and Java applets.

Besides, JavaScript is also a lightweight, interpreted scripting language that provides the same functionality as its VBScript counterpart. The syntax of the JavaScript languages is similar to C. JavaScript is a product of Netscape Communications Corporation and Sun Microsystems Inc.



From the security point of view, VBScript was initially designed to be a safe, interpreted language to be executed on the Internet Explorer browser. Designed to minimize possible security violations to the client's file and subsystems, VBScript prevents any file input/output and prevents access to the client's operating system. To further prevent destructive intentions, VBScript uses only one data type, *Variant*. (Johnson, S. & Ballinger, K. & Chapman, 1997)

Finally, I will use HTML (Hypertext Markup Language) and Java Script in web or system design while the server scripting is developed using VB Script.

## **2.11 Browser Consideration**

ASP only supports Microsoft Internet Explorer's server. It does not support Netscape Communicator's server. The Netscape Communicator's server could not understand the ASP syntax. Therefore, the Microsoft Internet Explorer browser is being used in my project.

### **3.0 SYSTEM ANALYSIS**

This chapter intends to ascertain the functional and non-functional requirements of this project. System development methodology and project planning will be discussed here. Besides that, the determination of the programming language, database and hardware is needed. A mixture of various kinds of tools will be determined and used to build this project.

#### **3.1 Methodology**

There are many types of software process models in software engineering. The software model process selected for this project is software prototyping. This model will clearly show what goes on during the development process, and it suggests developing the sequence of events the developers should encounter. (Sommerville, 2002)

Prototyping is a worthwhile technique for gathering specific information before a “real” system is being developed. It enables the developer to create a model of the system and refine it into a newer version, or discard and start creating another model that will fulfill the developer’s and user’s needs.



Prototyping allows more flexibility to meet with new requirements and the addition of more innovative features. The prototyping model is shown in Figure 2.

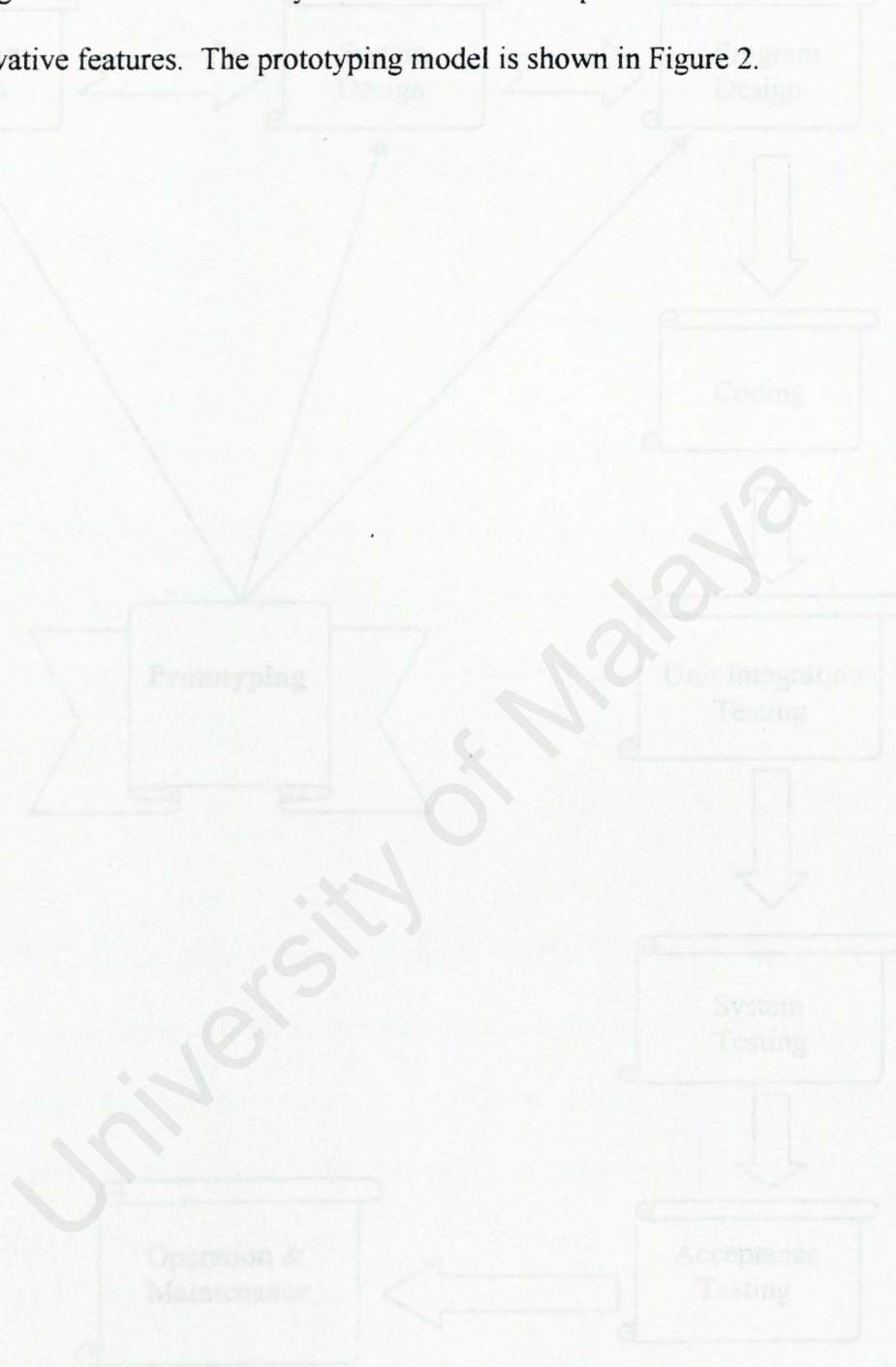
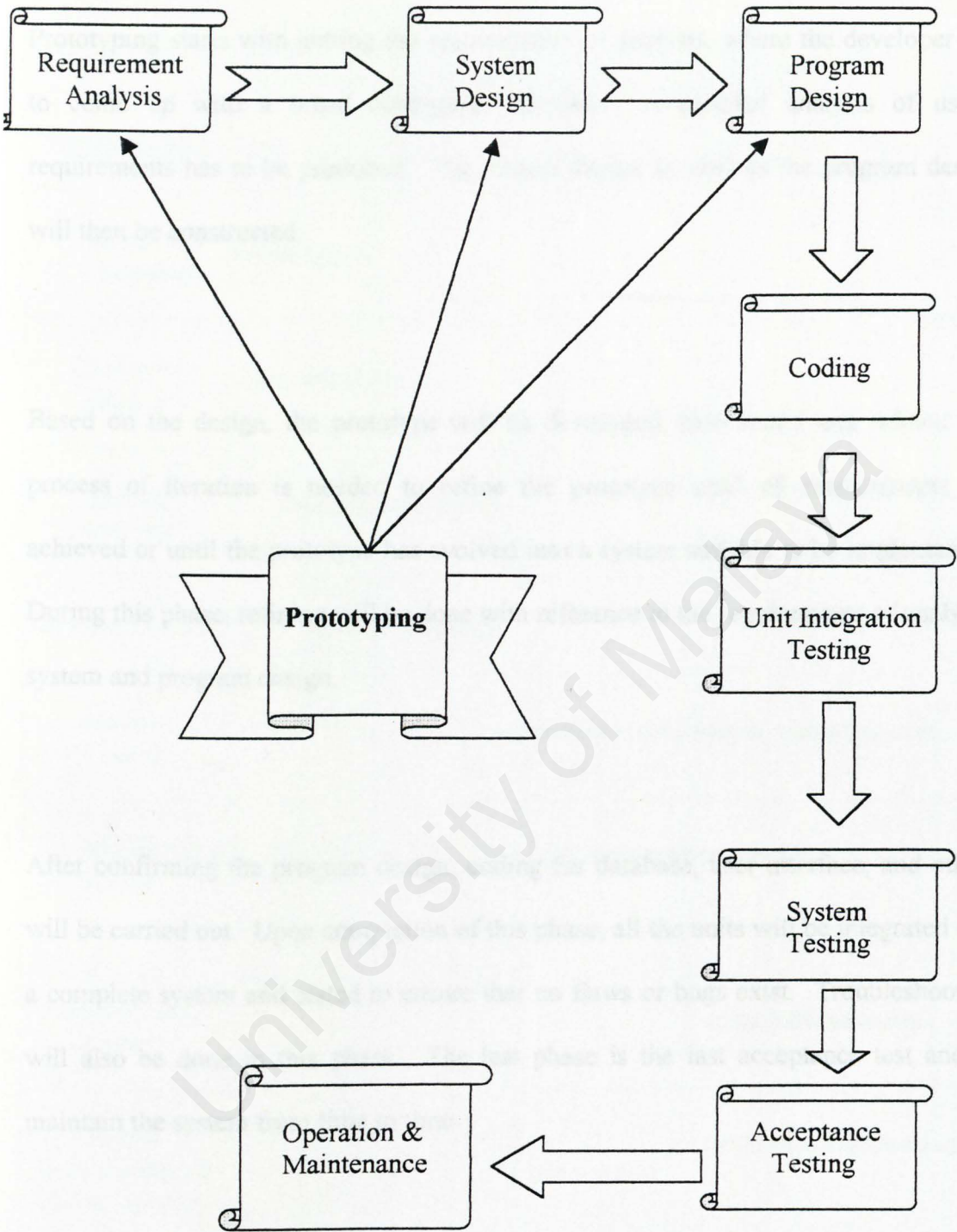


Figure 2: Prototyping Method



**Figure 2: Prototyping Method**



Prototyping starts with getting the requirements of analysis, where the developer has to come up with a broad conceptual solution. A detailed analysis of user's requirements has to be produced. The system design as well as the program design will then be constructed.

Based on the design, the prototype will be developed, then tested and refined. A process of iteration is needed to refine the prototype until all requirements are achieved or until the prototype has evolved into a system suitable to be implemented. During this phase, refining will be done with reference to the requirements of analysis, system and program design.

After confirming the program design, coding for database, user interface, and others will be carried out. Upon completion of this phase, all the units will be integrated into a complete system and tested to ensure that no flaws or bugs exist. Troubleshooting will also be done in this phase. The last phase is the last acceptance test and to maintain the system from time to time.

### **3.2 Project Schedule**

To achieve the objectives of the project in time, a timeline is planned to manage tasks. The schedule will ensure the project is accomplished before the submission date. The project schedule is shown in Table 1.

### 3.3 Requirements Analysis

Project Schedule (July 2001 – April 2002)										
Key Activities	July	Aug	Sept	Oct	Nov	Dec	Jan	Feb	Mac	Apr
<b>Research &amp; Reading</b>	■	■								
<b>Literature Review</b>	■	■								
<b>System Analysis</b>		■	■	■						
<b>System Design</b>		■	■	■						
<b>Module Coding</b>				■	■	■	■	■	■	■
<b>Module Testing</b>					■	■	■	■		
<b>System Integration &amp; Testing</b>							■	■	■	
<b>Report Documentation &amp; Submission</b>							■	■	■	■

Table 1: Project Schedule



### **3.3 Requirements Analysis**

Implementation of requirement analysis is to discuss more about abstract information and restrictions of the online course registration and payment system. Requirements analysis can be divided into two – functional requirements and non-functional requirements.

#### **3.3.1 Functional Requirements**

Functional requirements are the main features that will complete the operation and produce required results of the online course registration and payment system. The lists of functional requirements are as below.

##### **3.3.1.1 Students & Administrators Login Function**

This function will allow or restrict students or administrators to access into the following page. Students and administrators need an ID and password to login.

##### **3.3.1.2 Change Password Function**

This function allows students and administrators to change their password if they feel the old password is not safe.

### 3.3.1.3 Payment Calculating Function

This function will assist students to calculate the course registration payment based on the information from the database. The registration fees and examination fees will be automatically summed up. Therefore, the total amount that shown in the calculation would be the total fee that includes registration fee and examination fee.

### 3.3.1.4 Data Transaction

The modules involved here allow the end users (*UM administrators*) to create, retrieve, delete and update students' information, administrative records and course registration information. This component will ensure the database remains current with the 'real world' and support the information need of the UM administrators.

### 3.3.1.5 Data Queries

These modules allow UM administrators to retrieve the details of records from the database based on:

- i. Keywords such as student's matrix number, administrator's ID, and course number.
- ii. Non-keywords such as students and administrators login time.



### 3.3.2 Non-Functional Requirements

Non-functional modules are the modules that will not affect the system when they are operating and producing the required output. In this system, alteration in administrative records will not affect the required results. For example, a change in administrators' name, ID, and password will not affect the course registration and payment result. Besides that, changing information in the timetable will not affect the required result of the system as well.

#### 3.3.2.1 User Interface

The user interface in the system should present the required information in a user-friendly way. When designing the forms, some useful guidelines need to be followed, including meaningful title, logical grouping and sequencing of fields, visually appealing layout of the forms, familiar field labels, use of distinguished colors, visible space and boundaries for data-entry fields and convenient cursor movement.

#### 3.3.2.2 Storage Capacity

The storage capacity of the database must be enough for the postgraduate students in FSKTM. Besides that, the RAM (Random Access Memory) of the computer must be high enough to support the workload.

### **3.3.2.3 Easy to Use and Understand**

A standard navigation buttons and interface design of the system must be used.

Furthermore, the information in the system has to be easy to understand.

### **3.3.2.4 Reliable**

The system has to be reliable with less error occurred. The security of the data must be control by User ID and password.

### **3.3.2.5 Robust**

The system has to be robust. Only a little of time need to spend to recover the system when system failure. Data error has to be minimized after system failure with the validation of data if the data are safe to transfer to database.

## **3.4 Run-Time Requirements**

Online Postgraduate Course Registration and Payment System is developed to run on a standalone PC. This standalone PC can act as a server. The operating system for development work is Windows 95/98/2000/ME.

### **3.4.1 Window 95/98/2000/ME**

- At least a Pentium II 300 MHz machine



- 64 MB RAM (recommended for performance wise)
- 3.2 GB Hard Disk (recommended for performance wise)
- 52 X CD\_ROM Drive
- Modem with at least 56 Kbps
- Other standard computer peripherals

### 3.5 Data Flow Diagram

Data Flow Diagram (DFD) is a graphical characterization of the data processes and flows in a system. DFD depicts the broadest possible overview of system inputs, processes, and output, which correspond to the data movement through the system. Data flow diagram in this system is shown as below (Figure 3).

#### i. Context Diagram

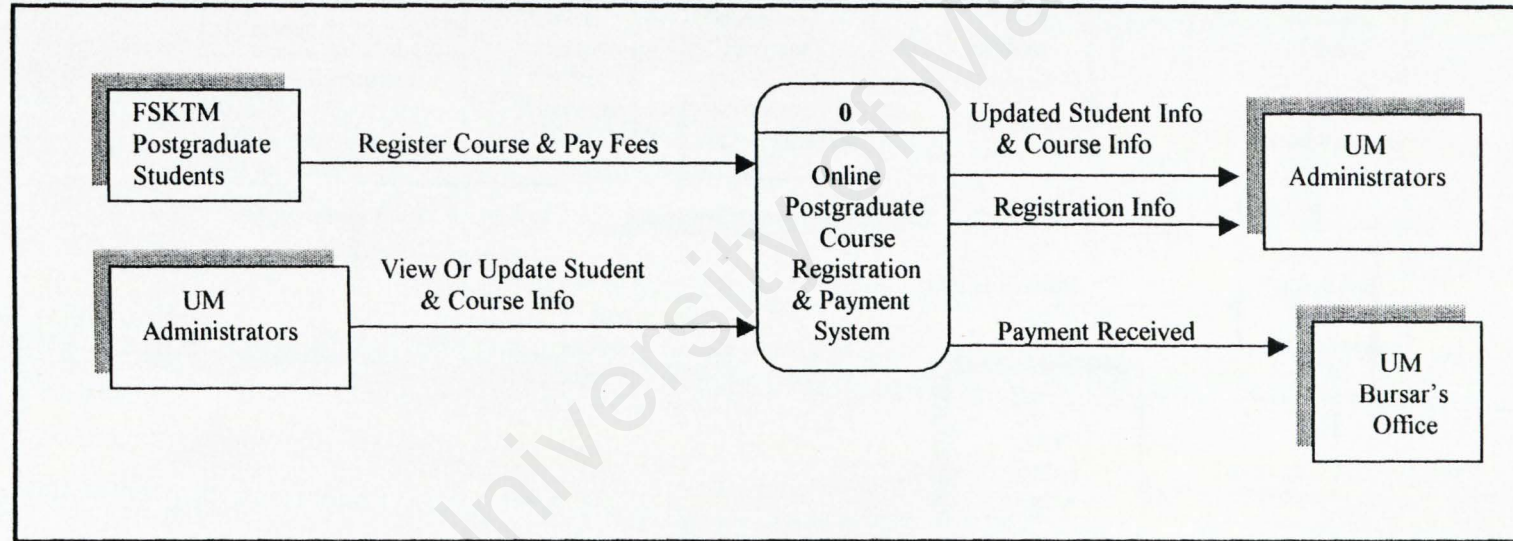


Figure 3: Context Level Diagram For An Online Postgraduate Course Registration and Payment System





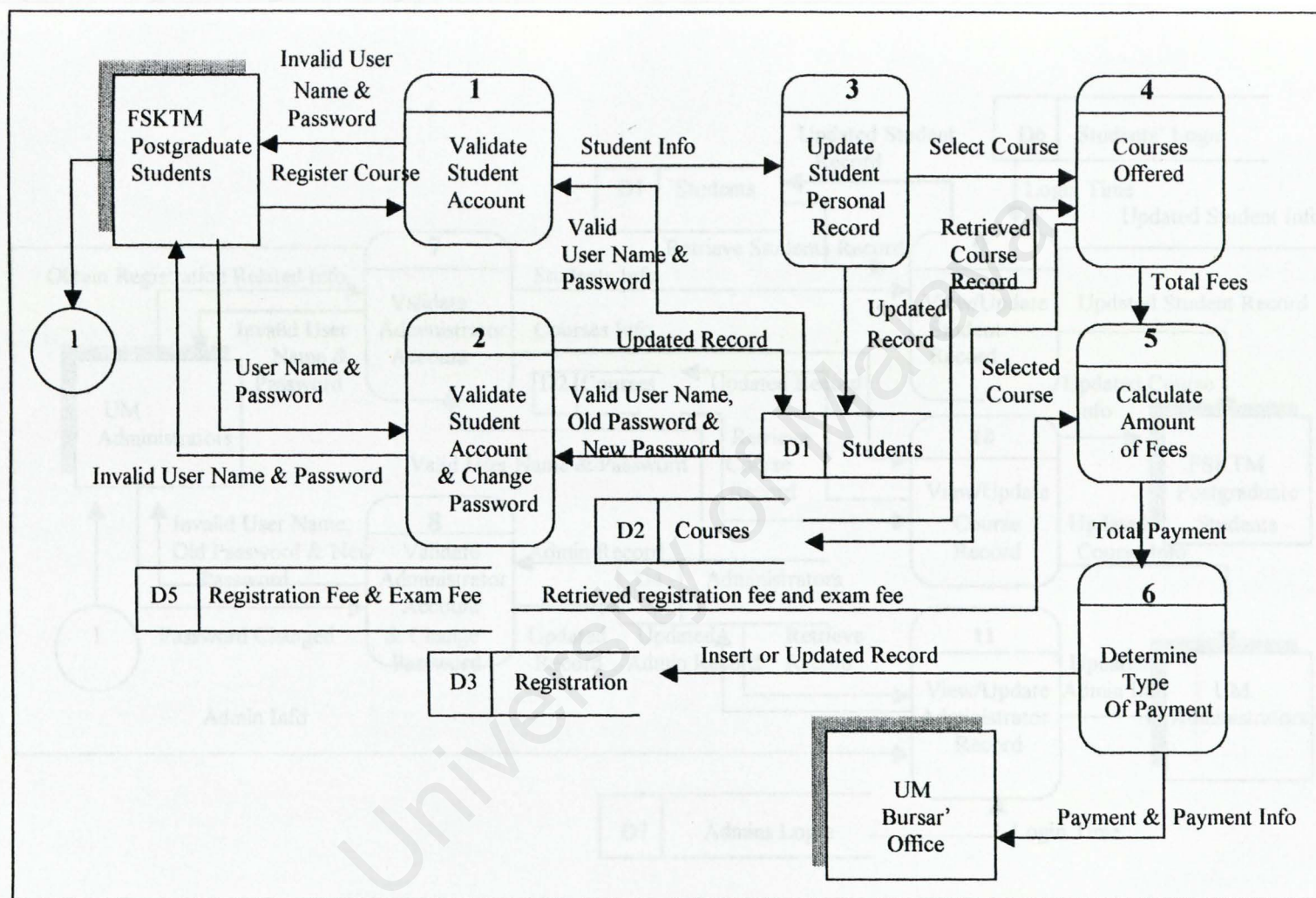


Figure 4: Diagram 0 For An Online Postgraduate Course Registration and Payment System



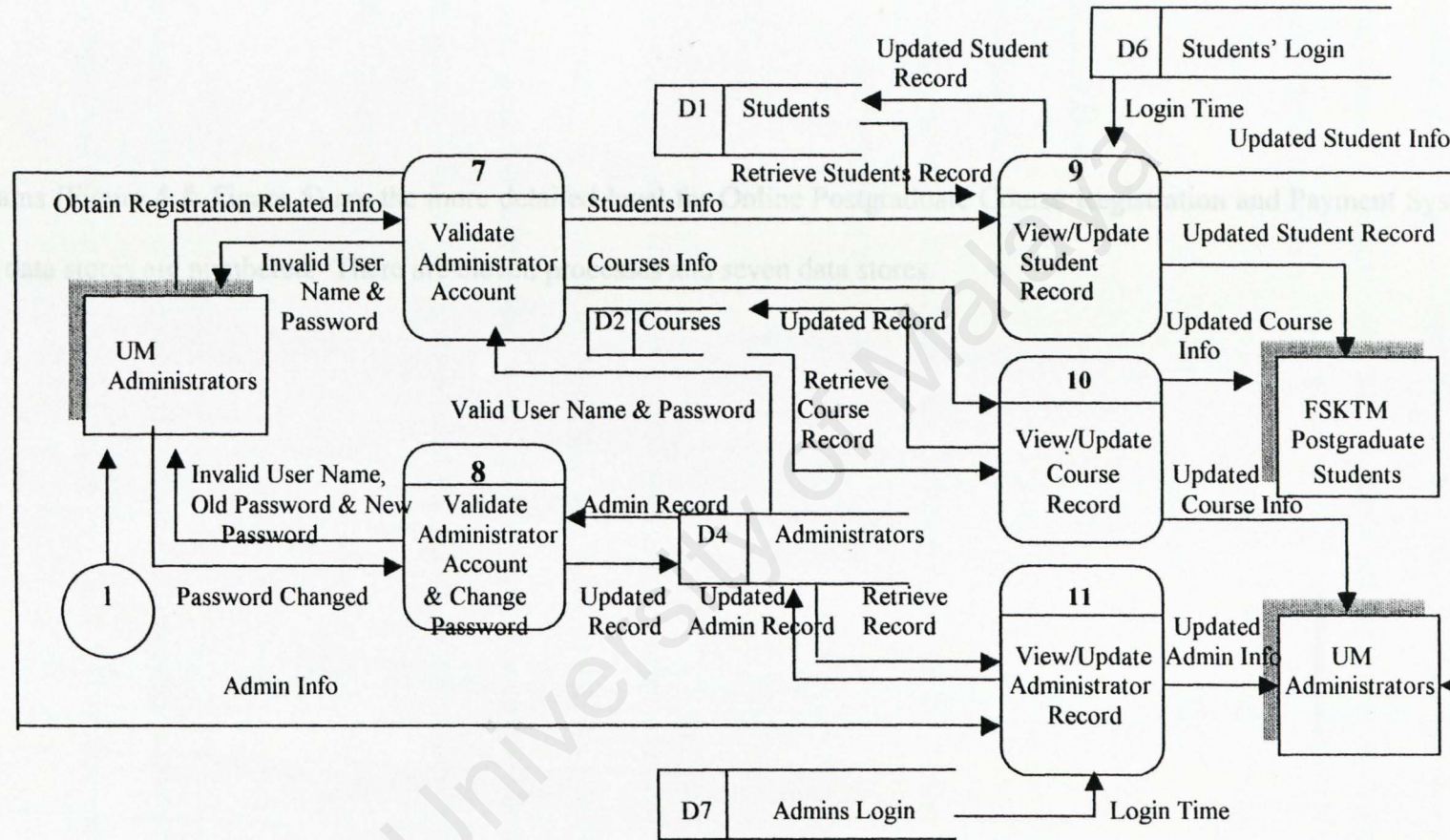


Figure 5: Diagram 0 (Continue)

These diagrams (Figure 4 & Figure 5) are the more detailed level for Online Postgraduate Course Registration and Payment System.

Process and data stores are numbered. There are eleven processes and seven data stores.



## 4.0 SYSTEM DESIGN

System design is a process through which the requirements are translated into a model or representations of the software that can be assessed for quality before coding begin. For this stage, the requirements that are identified earlier are translated into system features and characteristics.

### 4.1 Detail Design

#### 4.1.1 Architectural Design

The process model for this architectural design is based on the system structuring and modular decomposition approach.

In system structuring, the system is structured into a number of principal subsystems. A subsystem is a system in its own right, and its operation does not depend on the services provided by other subsystems. Each subsystem is composed of modules, and has defined interfaces, which are needed for communication with other subsystems. (Kendall & Kendall, 1999)

For module decomposition, it is based on assigning functions to components. I begin with a high-level description of the function that are to be implemented and build the

lower-level explanations of how each component will be organized and related to other components. (Kendall & Kendall, 1999)

Online Postgraduate Course Registration and Payment System is developed using a modular approach where each module is developed separately in components and are tested thoroughly. By using modular approach, future modifications and enhancements are made easy.

#### 4.1.2 System Architecture

The system architecture of this project consists of two parts, namely data architecture and application architecture. Therefore, the discussion will more concentrate on data structures and system modules in online course registration and payment system.

##### 4.1.2.1 Data Architecture

In this system, there are a few main modules in data architecture that need to be supported.

- ❑ The functions of view, update, and delete postgraduate students' records.
- ❑ The functions of view, update, and delete UM administrators' records.
- ❑ Course registration and fee payment function.
- ❑ The functions of view, update, and delete course record for MIT, MSE, MCS, MLIS, and PHD.



- ❑ The function to edit registration fee and examination fee records.
- ❑ The function of changing students' and administrators' passwords.
- ❑ Timetables for MIT, MSE, MCS, MLIS, and PHD.
- ❑ The login functions for students and administrators.
- ❑ Generate registration report functions.

#### 4.1.2.2 Application Architecture

Modular decomposition approach was used in designing this system. System functional separates the system into few modules. Modules that involved are:

##### a) Main page module

The main menu of the application is responsible for allowing the students to select the modules they require. The main menu consist of 4 major hyperlinks; *Home*, *Course Registration*, *Time Table* and *Administrator Login*. Besides that, there are three minor links such as FSKTM Site, Postgraduate Enquires Site and Email. This links will bring users to the web page that they would like to visit.

##### b) Student login module

If a student chooses to enter *Course Registration* module, they have to login with their programme name, ID and password.

### c) Student change password module

Students can change their password in this module. The information such as ID number, existing password, and new password are needed for this module to change password.

### d) Student's personal information list module

After login, student will go to this module. They can go to their own personal information list like Matrix No, Programme, Full Name, IC No, Sex, Marital Status, Race, Address, Telephone No, and Email. They can update their information such as phone number, home address or email address.

### e) Course selections module

Student will then proceed to course selections module to choose the course they want to register for the current semester. In this module, courses name, course code, credit hours, fees per credit hours and fees per course were shown.

### f) Course selected module

This module will automatically calculate the course fee including registration fee and examination fee after the student has chosen their course.



### **g) Fees payment module**

This module will allow the student to choose either to pay fees via visa or master card or pay cash to UM Bursar' Office.

### **h) Timetable module**

These are the updated timetables for MIT, MSC, MSE, MLIS and PHD students. All the timetables contain the information such as time for the class start, course name, lecturers' name, and the classroom location.

### **i) Administrator login module**

This module is for the administrators, not students. Administrators like department head, treasurer, lecturer and staff also have to login with ID and password if they want to access this module.

### **j) Administrator change password module**

Administrators can change their password if they feel that the existing password is not safe. The information such as ID number, existing password, and new password are needed for this module to change password.

#### **k) View students' information module**

Lecturer and staff only can view the students' records. They cannot edit or delete the students' records. But, the Department Head and UM Treasurer can view, edit, update, and delete the course information, students' records, administrators' records, course registration report, registration fee and examination fee.

Administrators can edit, delete and insert the MIT course information.

#### **l) Edit menu module**

This module is only a web page that consists of hyperlinks such as Edit students' information link, Edit administrators' information link, Edit registration fee and exam fee link, Edit MIT course information link, Edit MSE course information link, Edit MCS course information link, Edit MLIS course information link, Edit PHD course information link, Students access list, Administrators access list link, and Course registration reports link.

#### **m) Edit students' information module**

Administrators can search, edit, update, add or delete students' record in this module.

#### **n) Edit administrators' information module**

Administrators like department head and treasury officer can search, edit, update, add or delete administrators' record in this module.



u) Students access list module

**o) Edit registration fee and exam fee module**

Administrators can change the fee's figure every semester.

**p) Edit MIT course information module**

Administrators can edit, delete and insert the MIT course information.

**q) Edit MSE course information module**

Administrators can edit, delete and insert the MSE course information.

**r) Edit MCS course information module**

Administrators can edit, delete and insert the MCS course information.

**s) Edit MLIS course information module**

Administrators can edit, delete and insert the MLIS course information.

**t) Edit PHD course information module**

Administrators can edit, delete and insert the PHD course information.

#### **u) Students access list module**

Administrators can check students access list if they access this system and register their course. The list will show the login time of all students.

#### **v) Administrators access list module**

Administrators can check administrators access list if they access this system. The list will show the login time of all administrators.

#### **w) Course registration reports module**

In this module, there is a table that shows all the courses that registered by students. Besides that, administrators can also search the total of students that register a particular course.

### **4.1.3 Database Design**

#### **4.1.3.1 Entity-Relationship (E-R) Model**

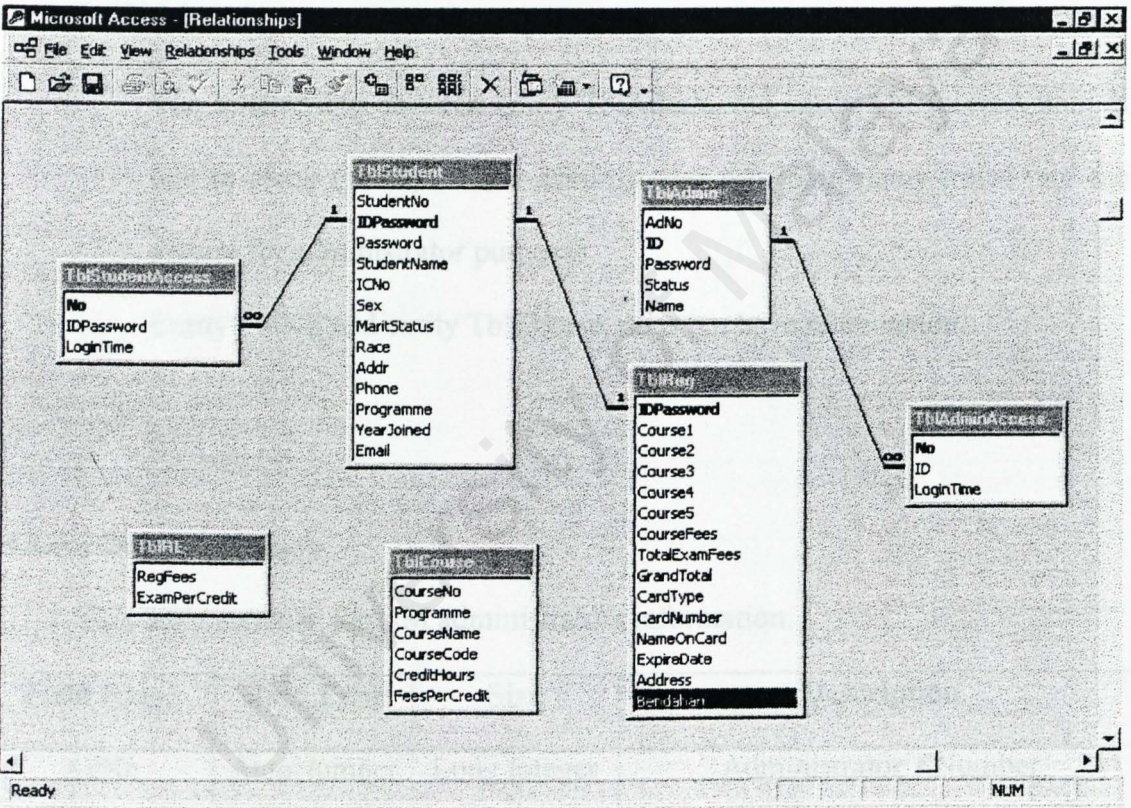
An E-R model is a logical representation of the data. It illustrates the relationship between entities in the related environment and the attributes of both the entities and their relationship. (Connolly, T. & Begg, 2000). It is commonly used to:

- ✓ Translate different view of data among administrators, users and programmers to fit into a common framework.



- ✓ Define data processing and constraint requirement to help meeting the different views.
- ✓ Help implement the database.

E-R Diagram for Online Postgraduate Course Registration and Payment System is shown in Figure 6.



**Figure 6: E-R Diagram for Online Postgraduate Course Registration and Payment System**

- i. The relationship between entity TblStudent and entity TblStudentAccess is one to many (1: M). Each student can login many times into the system for course registration and payment.
- ii. The relationship between entity TblStudent and entity TblReg is one to one (1: 1). Each student can register course many times and the records will keep updating every time.
- iii. The relationship between entity TblAdmin and entity TblAccessAdmin is one to many (1: M). Each administrator can login many times into the system for administrator purposes.
- iv. Entity TblIRE and entity TblCourse, does not have relationship.

#### 4.1.3.2 Data Dictionary

- i. This table contains the UM administrators information.

Field Name	Data Type	Size	Description
AdNo	AutoNumber	Long Integer	Administrator's Number
ID	Text	9	Administrator's ID
Password	Text	10	Administrator's Password
Status	Text	50	Administrator's Level
Name	Text	50	Administrator's Name

**Table 2: UM Administrators Information**



ii. This table contains FSKTM postgraduate students information.

Field Name	Data Type	Size	Description
StudentNo	AutoNumber	Long Integer	Student's Number
IDPassword	Text	9	Student's Matrix Number
Password	Text	10	Student's Password
StudentName	Text	50	Student's Name
ICNo	Text	12	Student's IC Number
Sex	Text	10	Student's Sex
MaritStatus	Text	20	Student's Marital Status
Race	Text	10	Student's Race
Addr	Text	100	Student's Address
Phone	Text	30	Student's Phone Number
Programme	Text	10	Student's Programme
YearJoined	Text	4	Year Join The Programme
Email	Text	50	Student's Email Address

**Table 3: FSKTM Postgraduate Students Information**

iii. This table contains courses particulars.

Field Name	Data Type	Size	Description
CourseNo	Text	10	Course Number

Programme	Text	5	Programme Name
CourseName	Text	200	Course Name
CourseCode	Text	8	Course Code
CreditHours	Number	Integer	Credit Hours Per Course
FeesPerCredit	Currency	6	Fees Per Credit Hours

**Table 4: Courses Particulars**

iv. This table contains the courses that selected, total fees and payment information.

Field Name	Data Type	Size	Description
IDPassword	Text	9	Student's Matrix Number
Course1	Text	200	Course 1 Register By Student
Course2	Text	200	Course 2 Register By Student
Course3	Text	200	Course 3 Register By Student
Course4	Text	200	Course 4 Register By Student
Course5	Text	200	Course 5 Register By Student
CourseFees	Currency	6	Course Fees That Excluding Registration Fee and Examination Fee
TotalExamFees	Currency	6	Exam Fee Per Credit Hours Multiply By Total Credit Hours
GrandTotal	Currency	6	Total Course Fee + Registration Fee + Total Examination Fee
CardType	Text	50	Type of Credit Card
CardNumber	Text	20	Credit Card Number



NameOnCard	Text	50	Name of Card Holder
ExpireDate	Text	5	Expire Date of Credit Card
Address	Text	100	Address On Credit Card
Bendahari	Text	50	Payment Information to UM Bursar' Office

**Table 5: Course(s) Selected Information and Payment Information**

v. This table contains amount of registration fee and examination fee per credit hour.

Field Name	Data Type	Size	Description
RegFees	Currency	6	Registration Fee
ExamPerCredit	Currency	6	Examination Fee Per Credit Hour

**Table 6: Registration Fee & Examination Fee Per Credit Hour**

vi. This table contains the record of students' login time.

Field Name	Data Type	Size	Description
No	AutoNumber	Long Integer	Number
IDPassword	Text	9	Student's Matrix Number
LoginTime	Date/ Time	General Date	Login Date and Time

**Table 7: Students' Login Time**

#### 4.1.4.1 Principles of Interface Design

Interface design is an area in which few rules exist. There are well-established

vii. This table contains the record of administrators' login time.

Field Name	Data Type	Size	Description
No	AutoNumber	Long Integer	Number
ID	Text	9	Administrators' ID
LoginTime	Date/ Time	General Date	Login Date and Time

**Table 8: Administrators' Login Time**

#### 4.1.4 User Interface Design

User interface design is based on the Graphical User Interface (GUI) approach. All the design is created using HTML and JavaScript. A good user interface can be designed by following some structured guidelines as follows:

- ✓ Planning the overall structures of the application
- ✓ Planning the content of the application
- ✓ Planning the interactive behavior
- ✓ Planning the look and feel of the application

- What You See Is What You Get — the interface should not hold any surprise for the user or exhibit strange behavior. There is no point to establishing exceptions and other aids to help support interaction and then removing user confidence in them by implementing them in an inconsistent manner. It is



#### 4.1.4.1 Principles of Interface Design

Interface design is an area in which few rules exist. There are well-established guidelines and principles that can be used to help improve the potential for success when designing an interface. A few of the more common are considered here.

- Direct Manipulation --- the user will feel in more control of an interactive experience if their physical actions result in physical, or apparently physical, results. This is one of the reasons we see features such as moveable icons and interface elements such as sliders on scrollbars.
- Aesthetic Integrity --- confusing and unattractive interfaces only serve to confuse and annoy the user. The interface should be clear and uncluttered.
- Feedback and Dialogue --- users need feedback in order to know what's happening, allowing them to feel comfortable in using the system. We have to make feedback timely, concise and make it understandable to the user.
- What You See Is What You Get --- the interface should not hold any surprise for the user or exhibit strange behavior. There is no point to establishing metaphors and other aids to help support interaction and then removing user confidence in them by implementing them in an inconsistent manner. If an

element requires the inclusion of odd, or magic, properties, then it probably is not appropriate for the use being made of it.

- Frequency of User Versus Accessibility of Functionality --- make frequently used functionality easy to access and place the bits and pieces that are seldom used out of the way, to help simplify the interface. It is also useful to place dangerous functions where they will not be triggered accidentally.



4.1.4.2 Main Page Design

The interface design of the main page is shown below.

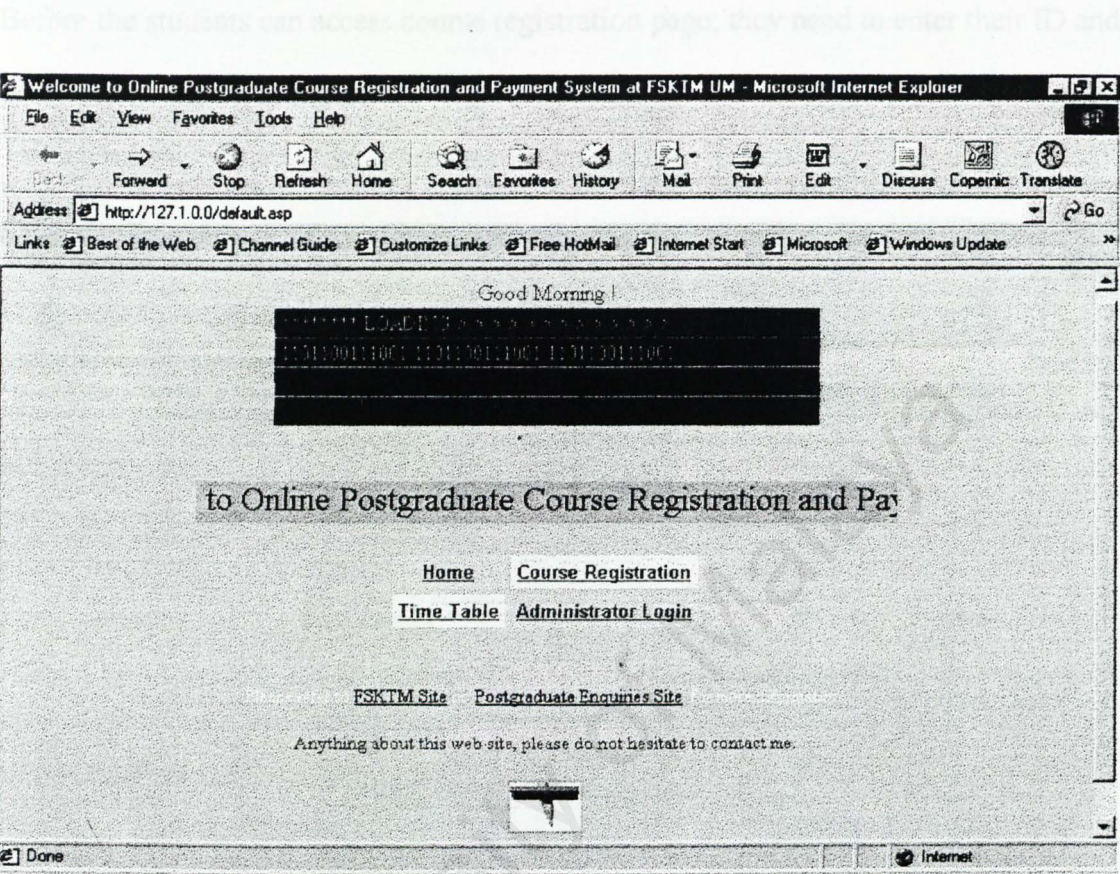


Figure 7: Main Page Design

Figure 8: Student Login Page Design

This page consists of a menu that includes *Home*, *Course Registration*, *Time Table*, and *Administrator Login*. It allows users to select the module they require. Besides that, there are two links that will bring users to *FSKTM web site* and *Postgraduate Enquires web site*. Furthermore, there is an email icon at the button of the page allow users to send email to the person in charge if they have any questions regarding the system.



4.1.4.3 Course Registration Pages Design

Before the students can access course registration page, they need to enter their ID and password as shown in Figure 8.

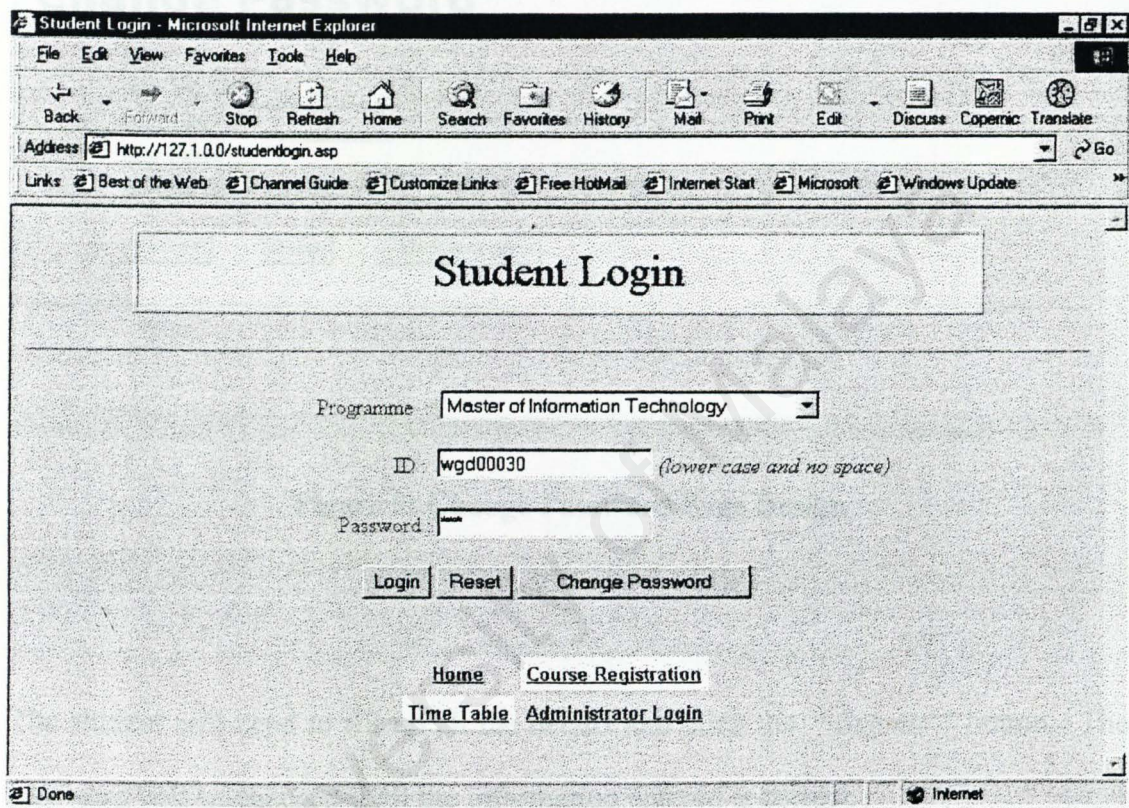


Figure 8: Student Login Page Design

In the student login page (Figure 8), once the student successfully login, they will go

If the student wants to change their password, they can click on the “ *Change Password* ” button and they will go to the web page as shown in Figure 9.



Change Password - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Refresh Home Search Favorites History Mail Print Edit Discuss Copernic Translate

Address <http://127.1.1.0/studentchangepassword1.asp> Go

Links [Best of the Web](#) [Channel Guide](#) [Customize Links](#) [Free HotMail](#) [Internet Start](#) [Microsoft](#) [Windows Update](#)

---

## Change Password

ID :

Password :

New Password :

Re-type New Password :

Done Internet

**Figure 9: Change Password Page Design**

The student will need to fill up all the details and click the “*Submit*” button. The password will be changed if all the information entered is correct.

In the student login page (Figure 8), once the student successfully login, they will go to Figure 10 and Figure 11.

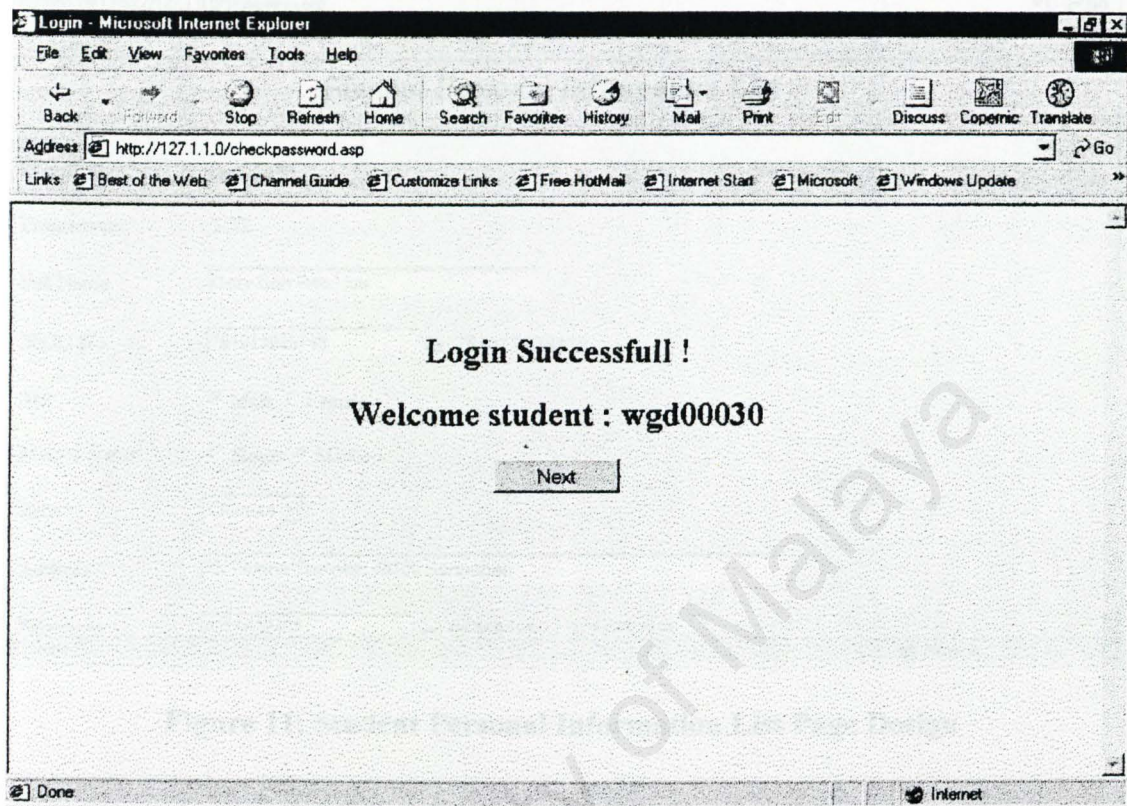


Figure 10: Successful Login Page Design



Personal Information List - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Refresh Home Search Favorites History Mail Print Edit Discuss Copernic Translate

Address <http://127.1.0.0/studentinfo.asp> Go

Links [Best of the Web](#) [Channel Guide](#) [Customize Links](#) [Free HotMail](#) [Internet Start](#) [Microsoft](#) [Windows Update](#)

### Student Personal Information List

(Please correct the information if it is wrong !)

Matrix No : wgd00030

Programme : MIT

Full Name :

NRIC No :  ex: 761013478139

Sex : ☒ Male ☐ Female

Marital Status : ☐ Single ☒ Married

Race :

Address :

Telephone :  ex: 0676089900 / 0126277899

Done Internet

**Figure 11: Student Personal Information List Page Design**

As shown in Figure 11, students can update their information. After that, they will go to course selections page (Figure 12 and Figure 13).

Course Selections - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Refresh Home Search Favorites History Mail Print Edit Discuss Copernic Translate

Address <http://127.1.1.0/coursestable.asp> Go

Links [Best of the Web](#) [Channel Guide](#) [Customize Links](#) [Free HotMail](#) [Internet Start](#) [Microsoft](#) [Windows Update](#)

### MIT Course Selections

Course Name	Course Code	Credit Hours	Fees Per Credit (RM)	Total Fees (RM)
Visual Programming: Concept and Application	WXGE6104	3	210	630
Multimedia Technology	WXGE6309	3	210	630
Web Commerce	WXGE6306	3	210	630
Knowledge Management	WXGE6308	3	210	630
Database Concepts and Implementation	WXGE6101	3	210	630
Software				

Done Internet

Figure 12: Course Selections Page Design



Course Selections - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Refresh Home Search Favorites History Mail Print Edit Discuss Copernic Translate

Address http://127.1.1.0/coursestable.asp Go

Links Best of the Web Channel Guide Customize Links Free HotMail Internet Start Microsoft Windows Update

Practice

Virtual Reality Environment	WXGE6310	3	210	630
Operating Systems	WXGE6314	3	210	630
Network Security and Management	WXGE6404	3	210	630
Dissertation	WXGE6899	6	210	1260

Course 1 : WXGE6309 Multimedia Technology

Course 2 : WXGE6314 Operating Systems

Course 3 : Nil

Course 4 : Nil

Course 5 : Nil

Next ! Reset

Done Internet

Figure 13: (Continue from Figure 12)

In this course selections page, there is a table consists of courses' names, courses' codes, number of credit hours per course, fee per credit hour and total fee per course. Each student must at least choose one course and up to a maximum of five courses.

After the selection, this system will calculate all the fees as shown in Figure 14.

http://127.1.1.0/selectedcourse.asp - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Refresh Home Search Favorites History Mail Print Edit Discuss Copernic Translate

Address http://127.1.1.0/selectedcourse.asp Go

Links Best of the Web Channel Guide Customize Links Free HotMail Internet Start Microsoft Windows Update

**MIT Selected Course(s)**

Course Name	Course Code	Credit Hours	Fees Per Credit (RM)	Total Fees (RM)
Multimedia Technology	WXGE6309	3	RM210	RM630
Operating Systems	WXGE6314	3	RM210	RM630
Total	---	6credits	---	RM1260
Registration Fees	---	---	---	RM277
Examination Fees	---	6credits	RM10	RM60
Grand Total	---	---	---	RM1597

OK

Done Internet

**Figure 14: Selected Course(s) Page Design**

Here, the total fee will be calculated. The formula for the total fee is shown below:

$$\text{Total Fee} = \text{Total Course Fee} + \text{Registration Fee} + \text{Examination Fee}$$

After that, the student will proceed to payment page as shown in Figure 15.



Payment - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Refresh Home Search Favorites History Mail Print Discuss Copernic Translate

Address http://127.1.1.0/um/payment.asp Go Links

Payment Information

Credit Card Payment

Card Type	Mastercard
Card Number	454553-4381431
Cardholder Name	Goh Ren Huu
Expiry Date	0303
Address	85, Taman Templer, 70200 Seremban
<input type="button" value="Submit"/> <input type="button" value="Reset"/>	

Other Payment Type

<input type="button" value="OK"/>
-----------------------------------

Done Internet

**Figure 15: Payment Page Design**

Here, the students need to select their mode of payment. They can either pay their fees via visa or master card or pay cash to UM Bursar's Office. If they want to pay by visa or master card, they need to complete all the entries like card type, card number, name on card, expiry date of the card, and card holder address, as shown in Figure 15. Then, click "Submit" button. All this information will receive by UM Bursar's Office. After that, UM Bursar's Office will contact the bank electronically and verify that credit card information students have provided matches the credit card information where their credit cards are mailed. If the information they have provided is different from information that bank has provided to UM Bursar's Office, the course registration will be delayed or canceled.

However, if they want to pay cash to UM Bursar' Office, just click " OK ". Then the record will be stored in the database. Students need to settle their fees within two weeks after the course registration.

4.1.4.4 Time Table Design

The timetable selection page for MIT, MSE, MCS, MLIS, and PHD is shown here.

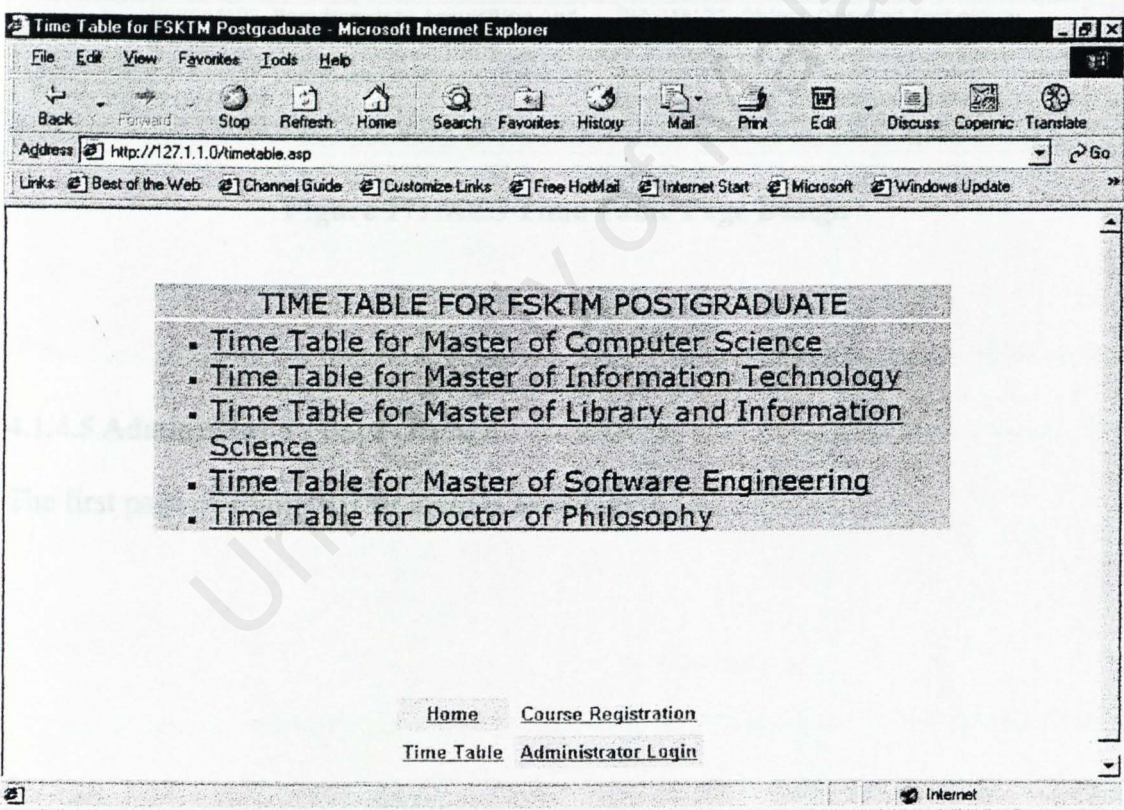


Figure 16: Timetable Selections Page Design

The following is an example of MCS timetable page design.



Master of Computer Science - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Refresh Home Search Favorites History Mail Print Edit Discuss Copernic Translate

Address: http://127.1.1.0/timetablesmcs.asp Go

Links: Best of the Web Channel Guide Customize Links Free HotMail Internet Start Microsoft Windows Update

**Time Table for MCS Semester 2, Session 2001/2002**

Last Update : 1 December, 2001

Master of Computer Science		
	6.00-9.00pm	6.30-9.30pm
<b>Monday</b>	WXGA6302 <b>Neural Network</b> (Room A1) Puan Hanizah WRGA6302 <b>VLSI Technology</b> (BK3A) Teh Ying Wai	WXGA6101 <b>Advanced Issues In Programming</b> (Room A2) Dr. Sellappan WRGA6304 <b>High Performance Computer Systems</b> (BK2) Pn Hannyzzura Pal @ Affal
<b>Tuesday</b>	WXGA6303 <b>Digital Signal Processing</b> (Auditorium) Miss Laiha Mat Kiah	
<b>Wednesday</b>		WXGA6301 <b>Expert System</b> (BK1B) Dr Syed Malek Fakar Duani
<b>Thursday</b>	WXGA6102 <b>Requirements Acquisition And Modelling</b> (Auditorium) Dr. Sellappan	WXGA6103 <b>Object-Oriented Techniques</b> (BK3A) Teh Ying Wai
<b>Friday</b>	WXGA6104 <b>Distributed Systems</b> (Room A1) Puan Hanizah	WRGA6301 <b>Parallel And Concurrent Processing</b> (BK3A) Mazliza Othman

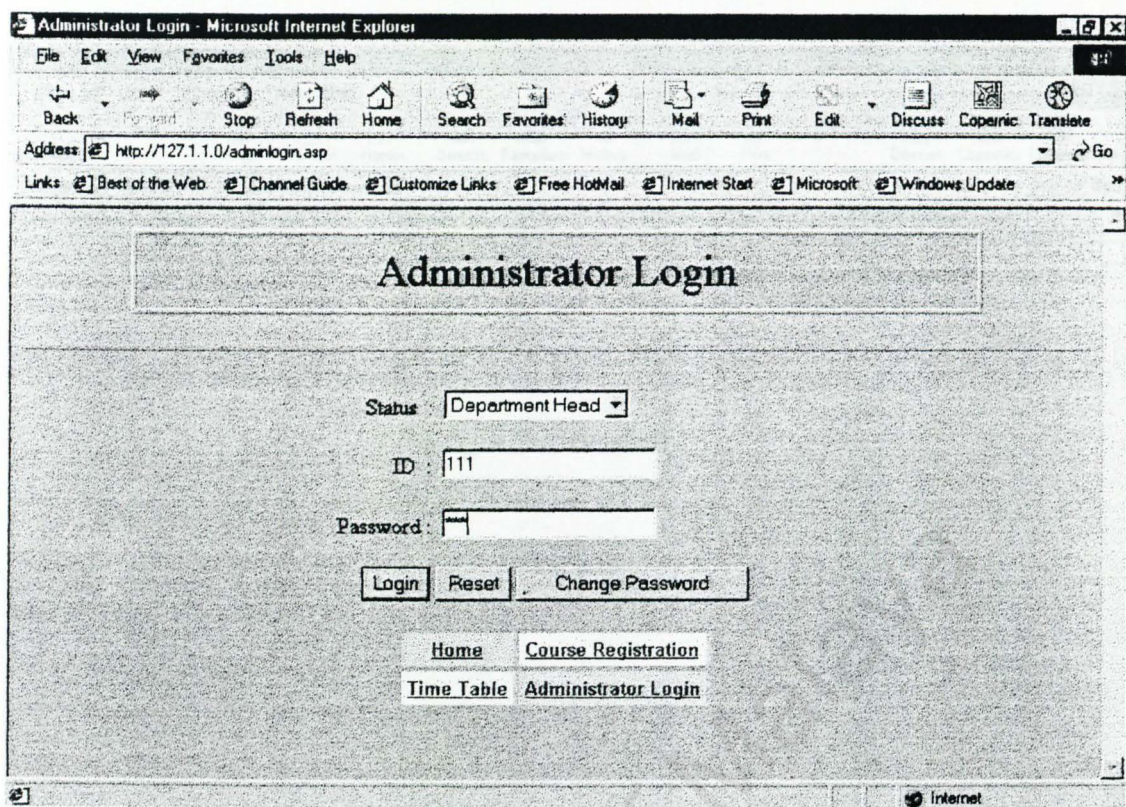
Done Internet

**Figure 17: MCS Time Table Page Design**

#### 4.1.4.5 Administrators Pages Design

The first page of administrator login is as in Figure 18.



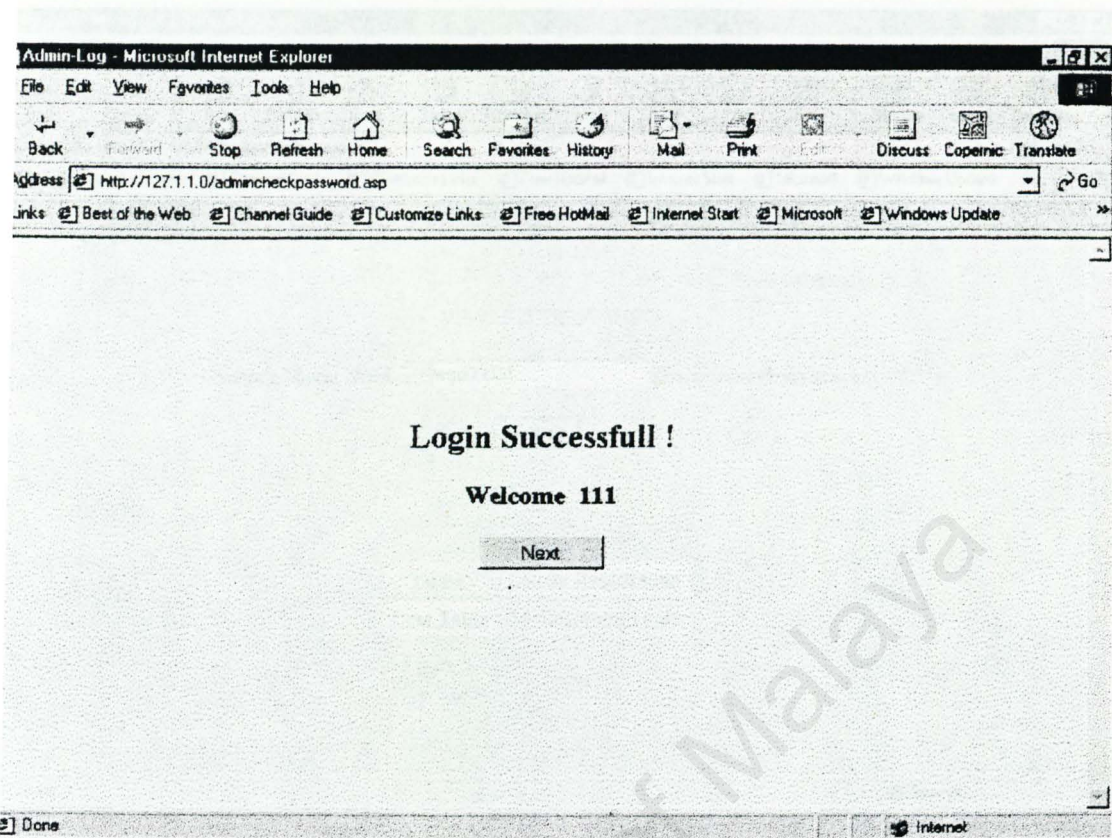


**Figure 18: Administrator Login Page Design**

If the administrators would like to change their password, they need to click on the “*Change Password*” button, a page look like the one in Figure 9 will appear. The steps to change password is exactly the same as for the students.

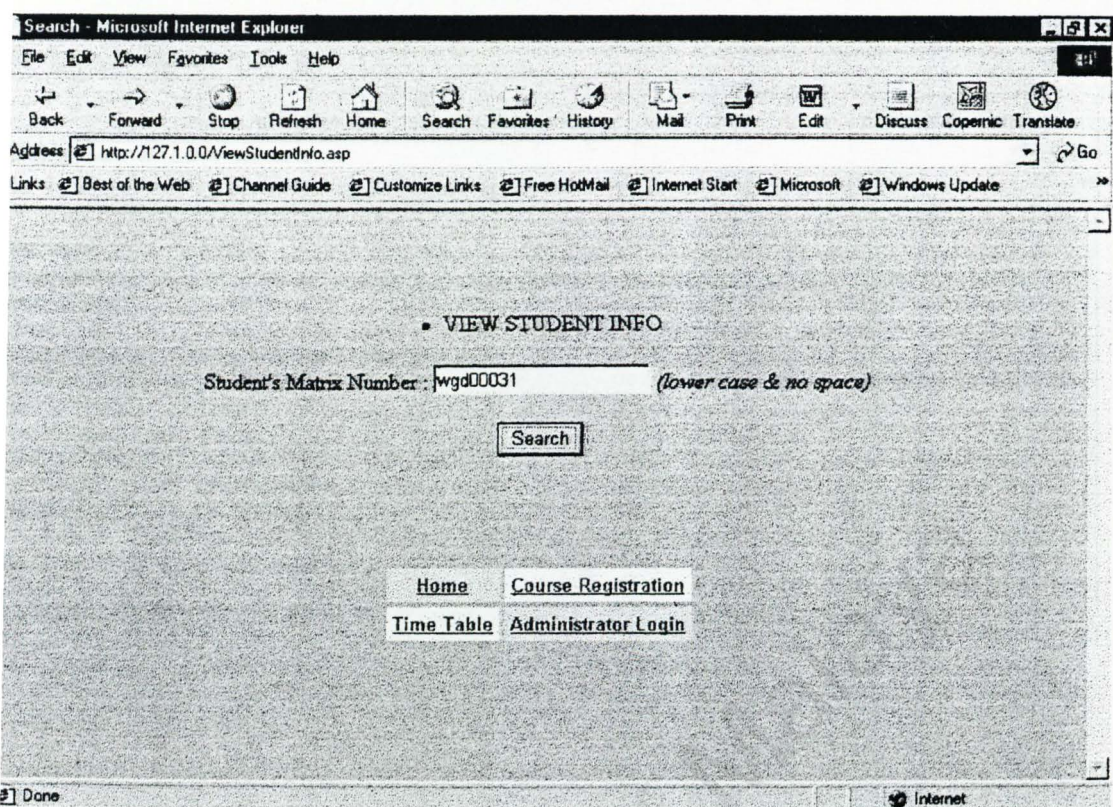
For login into the administrator login page, administrators have to fill in the *Status*, *ID*, and *Password* text box as shown in Figure 18. There are four type of status for administrators: Department Head, Treasurer, Lecturer and Staff. If *Status*, *ID*, and *Password* is not match, then access is denied. If the login is successful, then the page as shown in Figure 19 will appear.





**Figure 19: Successful Login Page Design**

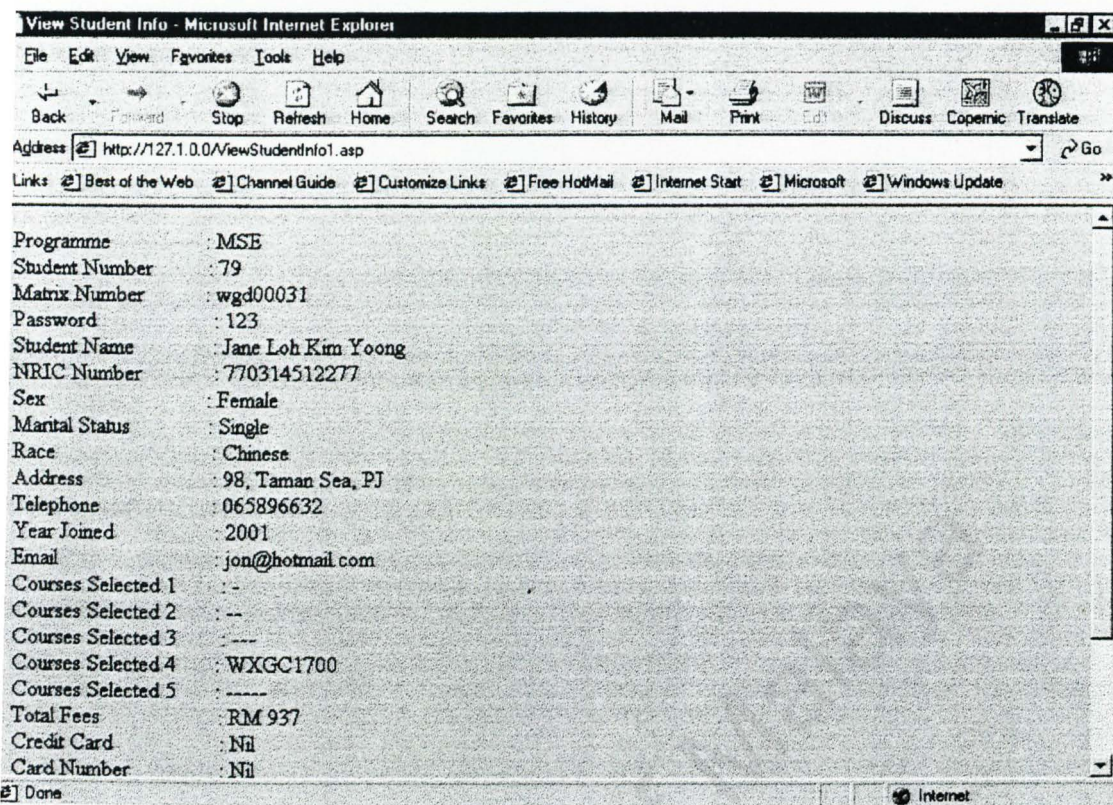
If the administrator is a lecturer or staff, they will proceed to the page as shown in Figure 20. Here, they can only view students' records but cannot edit it. As shown in Figure 20, they need to key in the student's matrix number. For example, wgd00030.



**Figure 20: View Student' Record Page Design**

If the matrix number is in the database record, then the page as shown in Figure 21 will display all the information of that particular student.

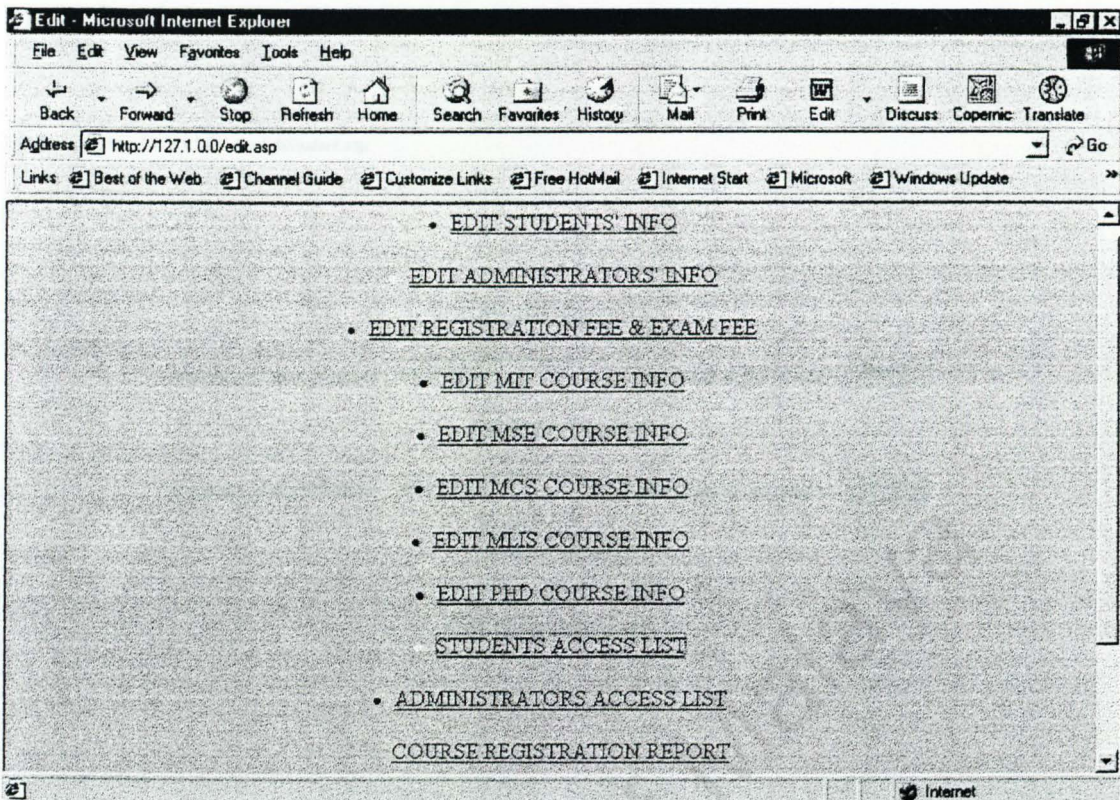




**Figure 21: Student's Record Page Design**

If the administrator is a Department Head or Bursar's Officer, they will proceed to Figure 22 after successfully login into the system. There are 11 choices for them to select; edit students' info, edit administrators' info, edit registration fee and exam fee, edit MIT course info, edit MSE course info, edit MCS course info, edit MLIS course info, edit PHD course info, students access list, administrators access list and course registration report.



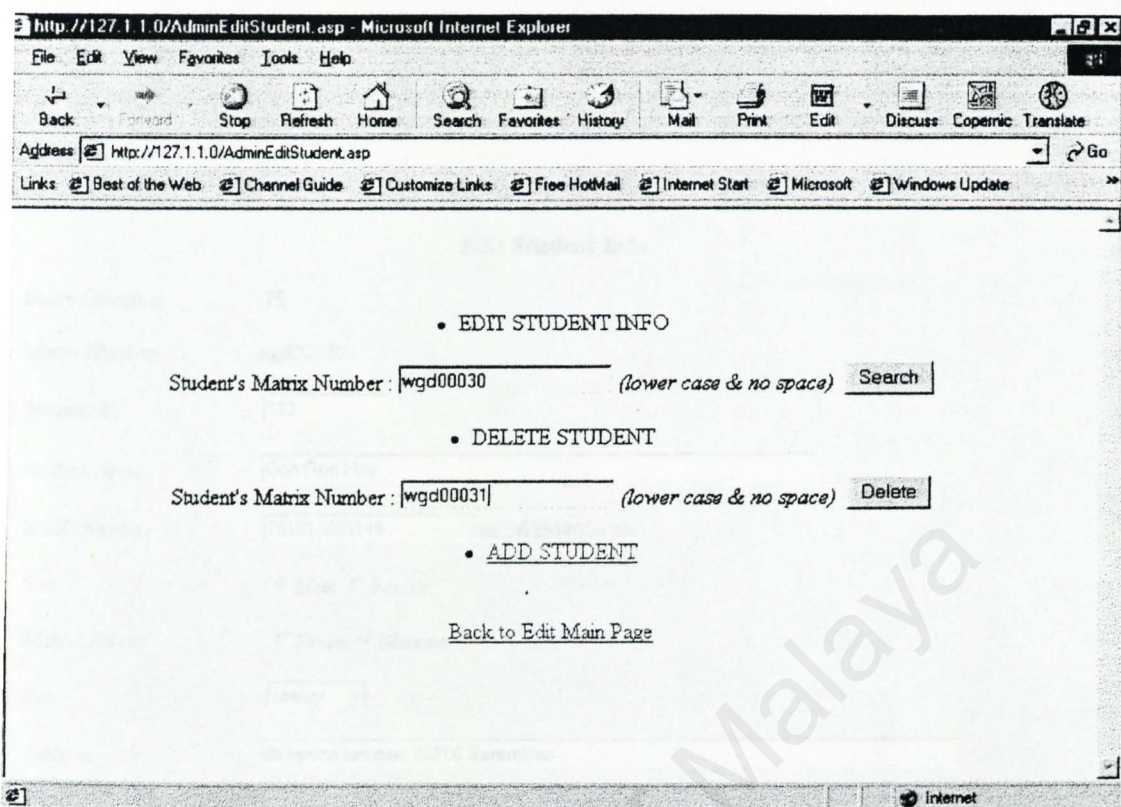


**Figure 22: Edit Menu Page Design**

#### **i. Edit Students' Information Page Design**

Click on “*Edit Student*”, the page as shown in Figure 23 will appear.





**Figure 23: Edit Student's Record Page Design**

If administrators want to edit or update students' records, they must enter the matrix number in the text box and click the "Search" button as shown in Figure 23. Then, Figure 24 will appear. Click the "Update" button after editing.

If they want to delete the student's record, just enter the student's matrix number and click delete.

Edit Student Info - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Refresh Home Search Favorites History Mail Print Edit Discuss Copernic Translate

Address <http://127.1.1.0/editstudentinfo.asp> Go

Links [Best of the Web](#) [Channel Guide](#) [Customize Links](#) [Free HotMail](#) [Internet Start](#) [Microsoft](#) [Windows Update](#)

### Edit Student Info

Student Number : 78

Matrix Number : wgd00030

Password : 123

Student Name : Goh Ren Huu

NRIC Number : 761013055149 ex: 768544054896

Sex : ☒ Male ☐ Female

Marital Status : ☐ Single ☒ Married

Race : Malay

Address : 85,taman templer, 70200 Seremban

Done Internet

**Figure 24: Student's Record Page Design**

If they want to add the information of a new student, they must click “*Add Student*”, then a page as shown in Figure 25 will appear.



**Add New Student**

Matrix No :

Password :

Programme :

Full Name :

NRIC No :  ex 761013478139

Sex : ☐ Male ☐ Female

Marital Status : ☐ Single ☐ Married

Race :

Address :

**Figure 25: Add New Student Page Design**

They must click the “*Add*” button after completing all the entries.

## ii. Edit Admin Page Design

They must click on the “*Edit Admin*” in the page shown by Figure 22, the system will bring them to the page as shown in Figure 26.

Admin Personal Info - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Refresh Home Search Favorites History Mail Print Edit Discuss Copernic Translate

Address http://127.1.1.0/AdminEditAdmin.asp Go

Links Best of the Web Channel Guide Customize Links Free HotMail Internet Start Microsoft Windows Update

### Administrator Personal Information

Admin No	ID	Password	Status	Name
4	111	111	Department Head	Mohammad Ali
5	222	222	Lecturer	Fatimah binti Mohammad Nor
6	333	333	Staff	Siti Hajar
7	444	444	Department Head	Sharizan bin Mohammad Nor
8	555	555	Treasurer	Debbie Goh Tian Yee

ID:

ID:

ID:

Password:

Name:

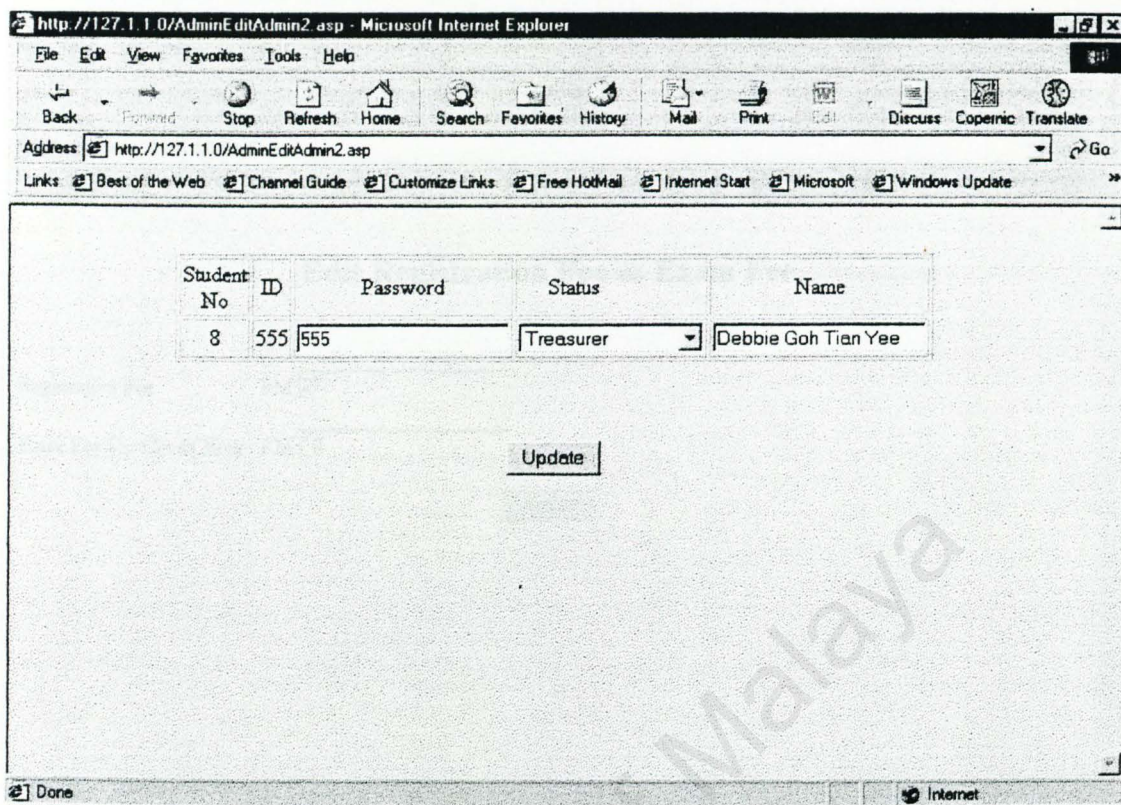
Status:

Done Internet

**Figure 26: Edit Administrators Records Page Design**

If administrators want to edit or update their own records, they must enter the administrator number in the text box and click the “ *Edit* ” button. Then, a page shown by Figure 27 will appear for them to update their records.



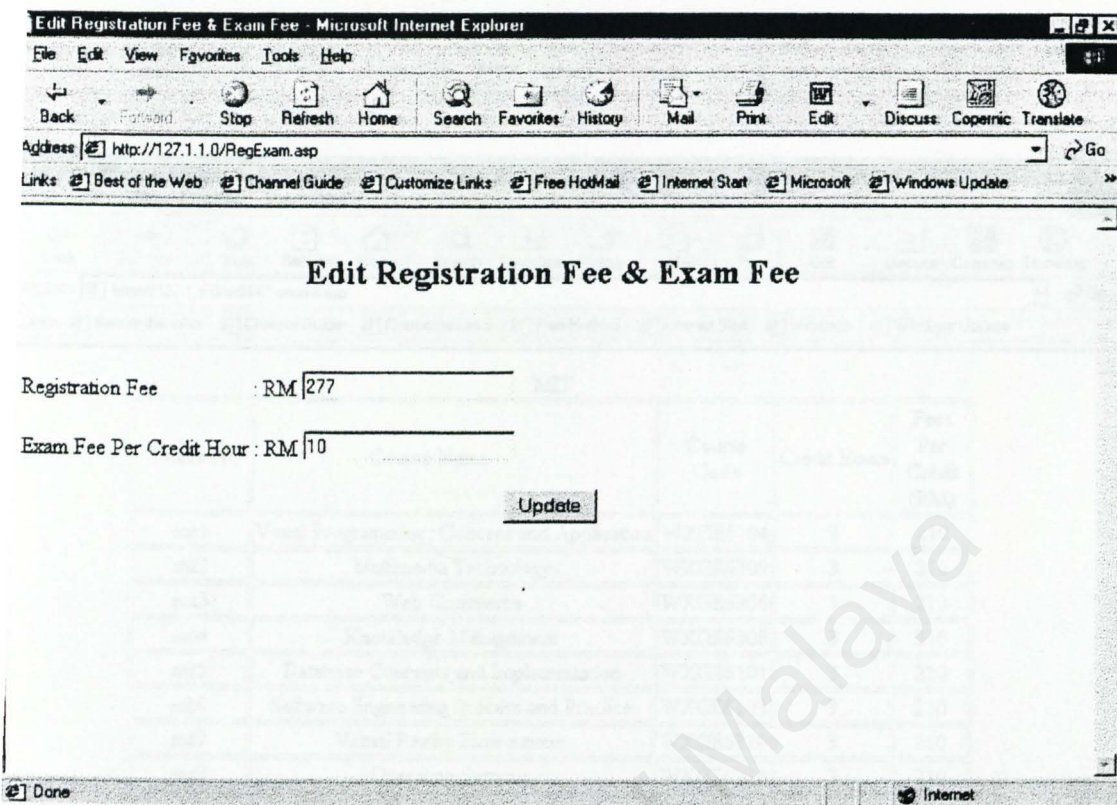


**Figure 27: Edit/Update Record Page Design**

If administrators want to delete a record, they must enter the administrator number and click the “Delete” button as shown in Figure 26. If they want to add the information of a new administrator, they must complete the 5 entries as shown in Figure 26 and click the “Insert” button. The record will be added in database.

### iii. Edit Registration Fee and Examination Fee Page Design

The administrators must click on “Edit Registration Fee and Examination Fee” as shown in Figure 22. Then they will see the page as shown in Figure 28.



**Figure 28: Edit Registration Fee and Examination Fee Page Design**

Then, they can change the figure in the text box and click the “*Update*” button. The record will be updated.

**iv. Edit MIT Course Info/ Edit MSE Course Info/ Edit MCS Course Info/ Edit MLIS Course Info/ Edit PHD Course Info Page Design**

Actually, the page design for all the course information is the same for different programmes. The MIT Course Info design page is shown next (Figure 29 & Figure 30).



Edit MIT - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Refresh Home Search Favorites History Mail Print Edit Discuss Copernic Translate

Address <http://127.1.1.0/editMITcourse.asp> Go

Links [Best of the Web](#) [Channel Guide](#) [Customize Links](#) [Free HotMail](#) [Internet Start](#) [Microsoft](#) [Windows Update](#)

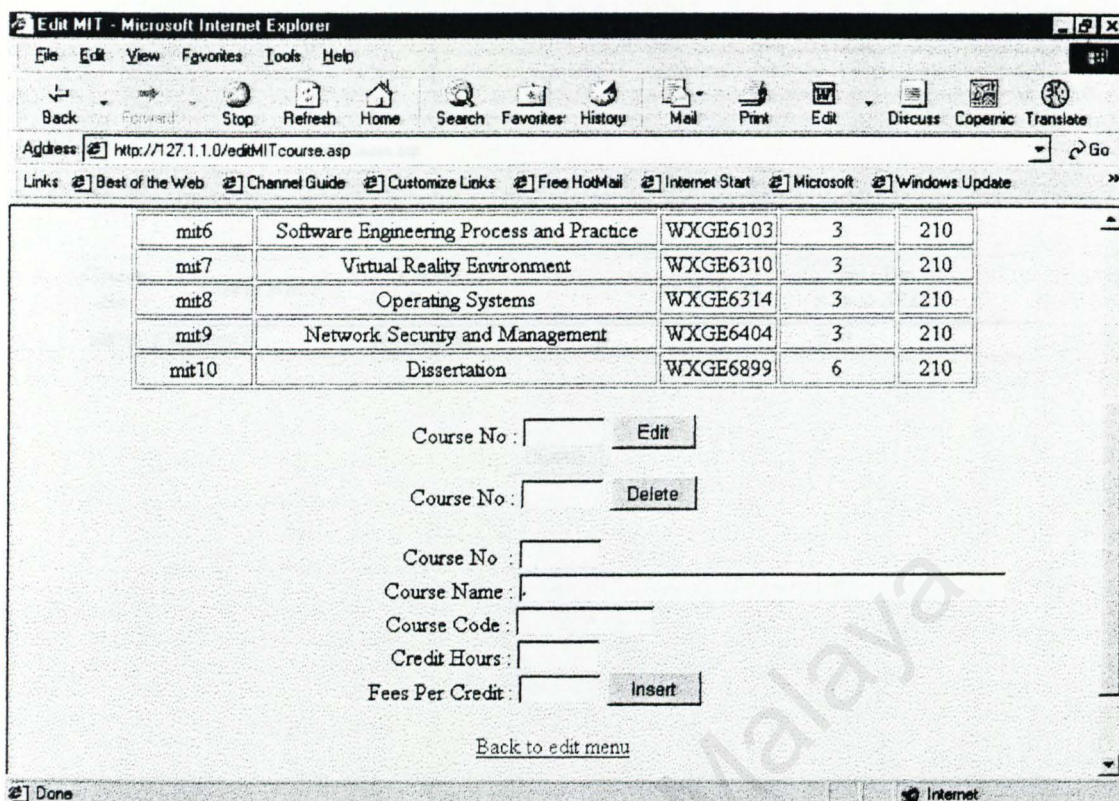
MIT

Course No	Course Name	Course Code	Credit Hours	Fees Per Credit (RM)
mit1	Visual Programming : Concept and Application	WXGE6104	3	210
mit2	Multimedia Technology	WXGE6309	3	210
mit3	Web Commerce	WXGE6306	3	210
mit4	Knowledge Management	WXGE6308	3	210
mit5	Database Concepts and Implementation	WXGE6101	3	210
mit6	Software Engineering Process and Practice	WXGE6103	3	210
mit7	Virtual Reality Environment	WXGE6310	3	210
mit8	Operating Systems	WXGE6314	3	210
mit9	Network Security and Management	WXGE6404	3	210
mit10	Dissertation	WXGE6899	6	210

Course No :  Edit

Done Internet

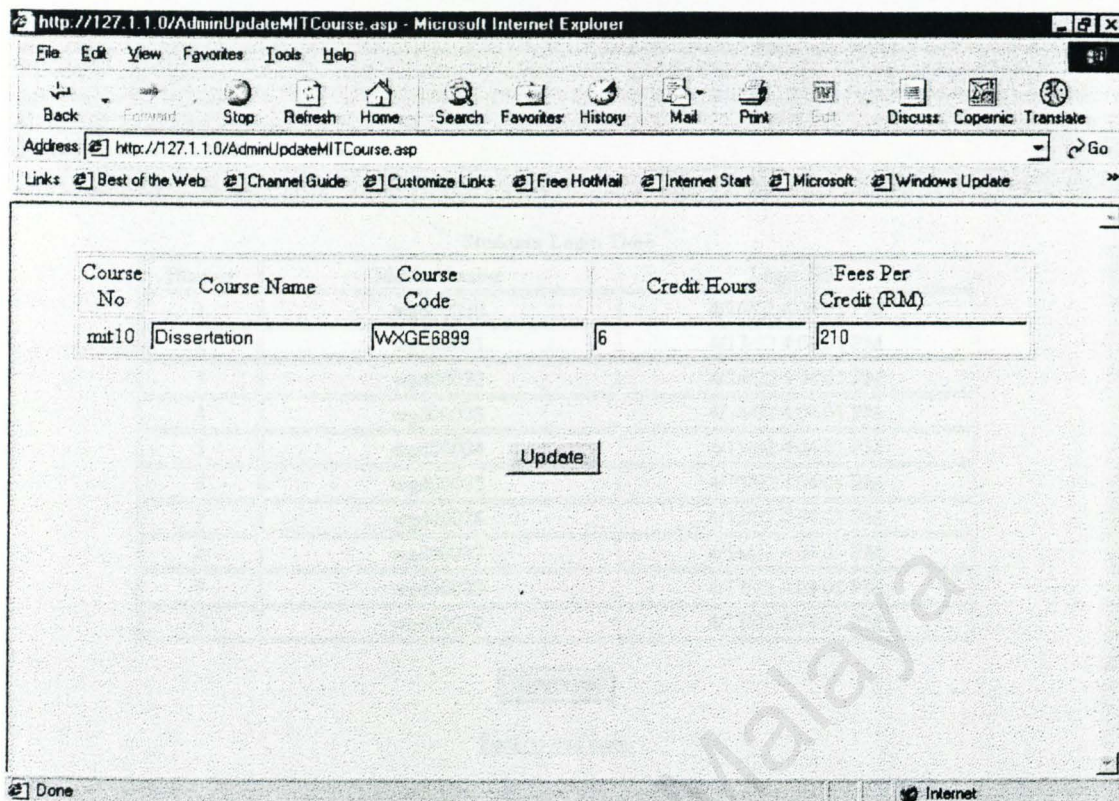
Figure 29: Edit MIT Course Page Design



**Figure 30: (Continue from Figure 29)**

To edit the course records, the administrators need to enter the course number in the text box and click the “*Edit*” button. The design page like Figure 31 will appear.



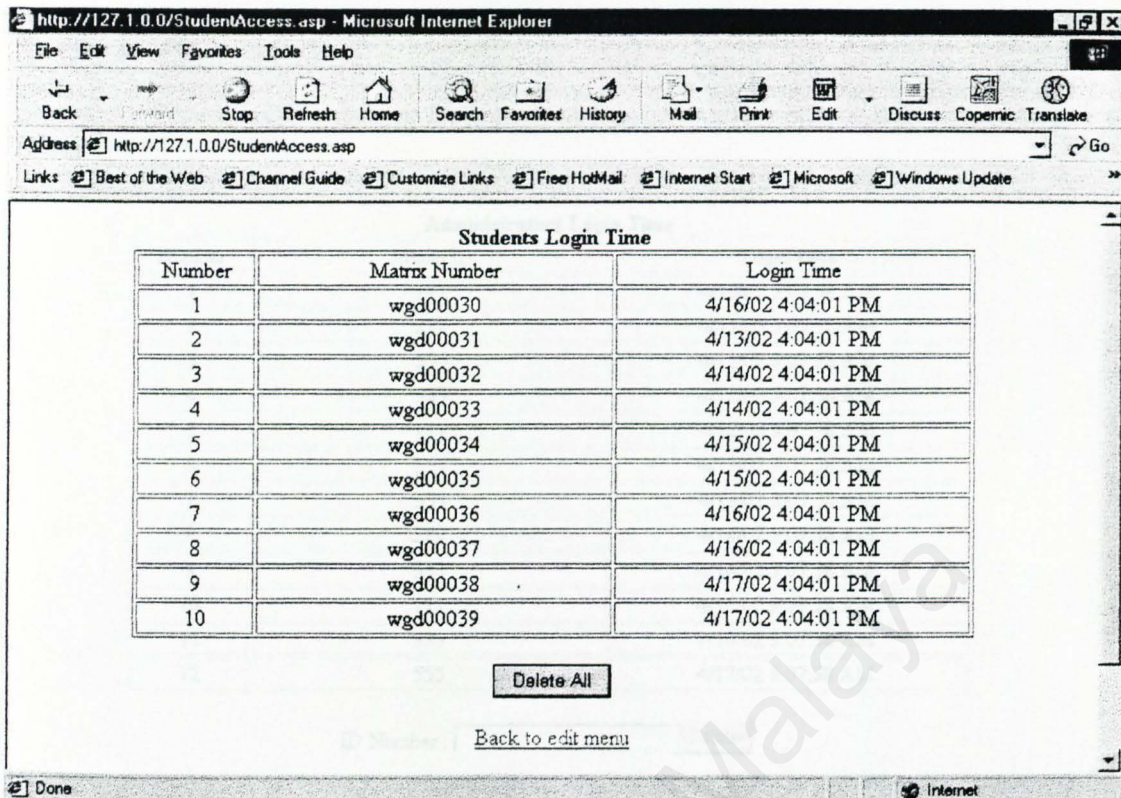


**Figure 31: Update Course Information Page Design**

To delete the course, they need to enter the course number and click the “Delete” button. The course will then be deleted. To add a new course, complete the five text boxes and click “Insert” button as shown in Figure 30. The new course will be inserted in the record.

#### v. Students Access List Page Design

If the administrators want to check the students’ login time, they can go to this page. (Figure 32)



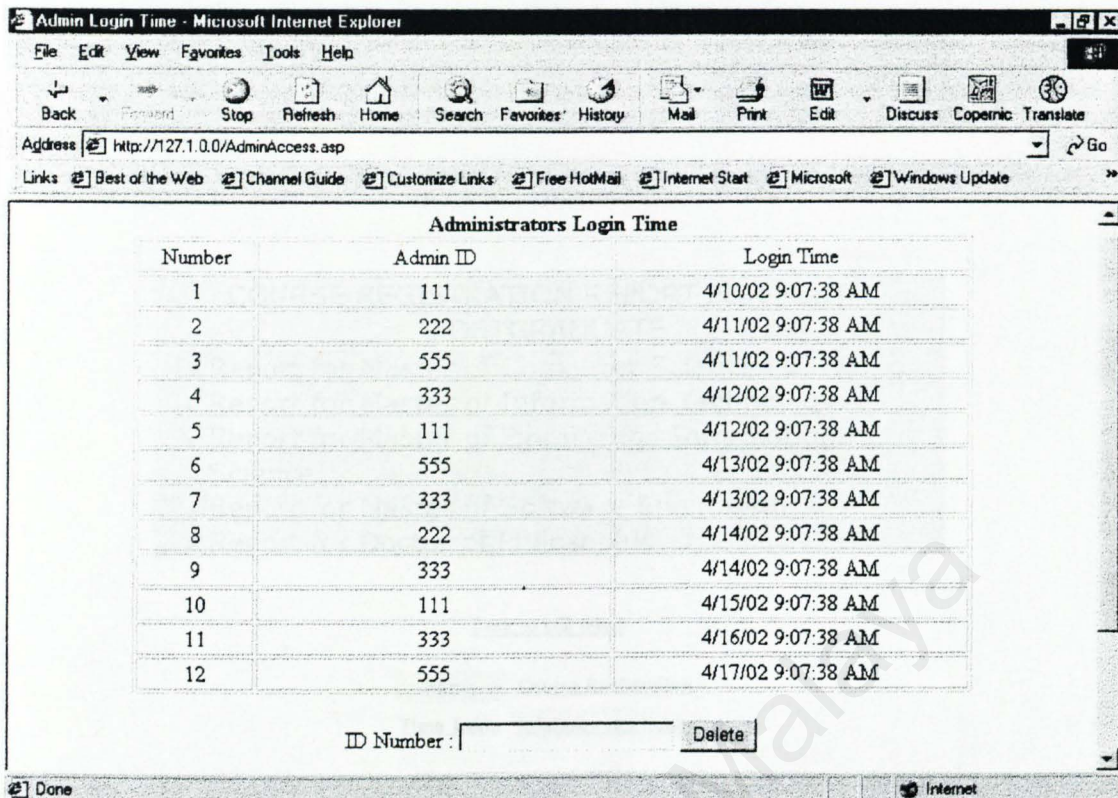
**Figure 32: Students' Login Time**

The UM Administrators can delete the records at the end of semester.

#### vi. Administrators Access List Page Design

If administrators want to check the admin staffs' login time, they can access into the page as shown in Figure 33.



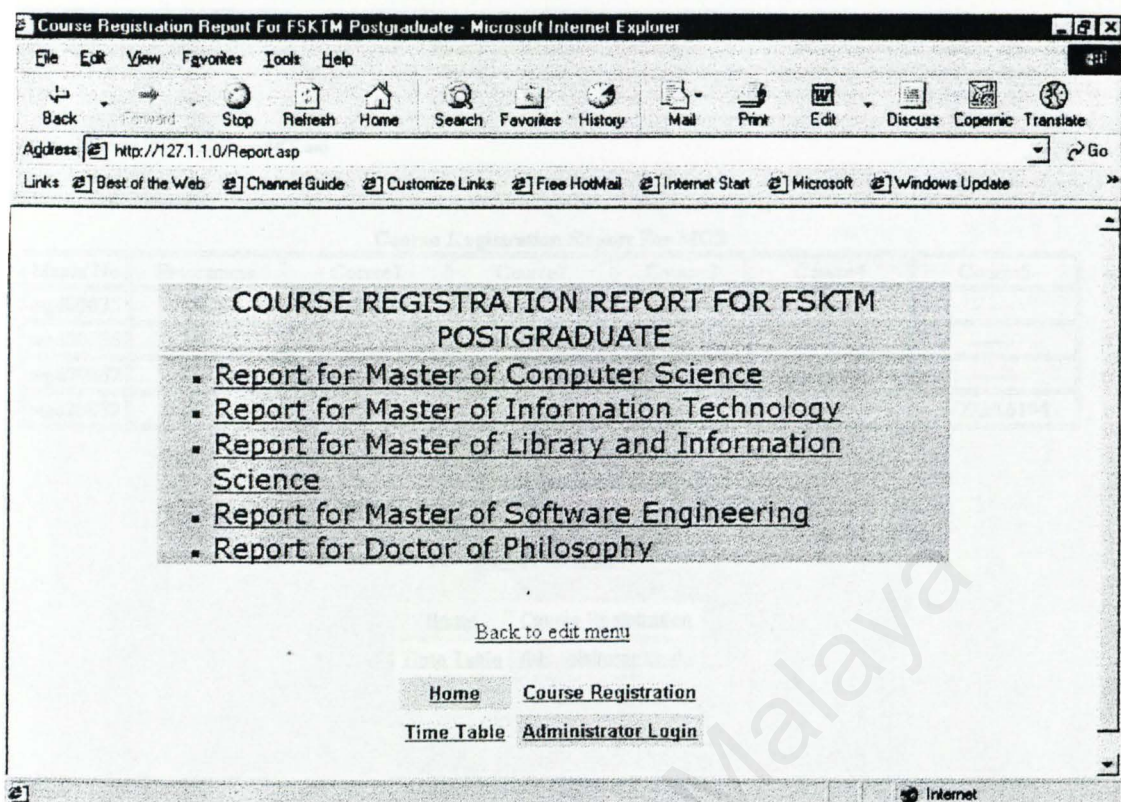


**Figure 33: Administrators' Login Time**

The administrators' login time can be deleted. Enter the ID number and click the "Delete" button, the record of the particular administrator will be deleted.

## vii. Course Registration Report

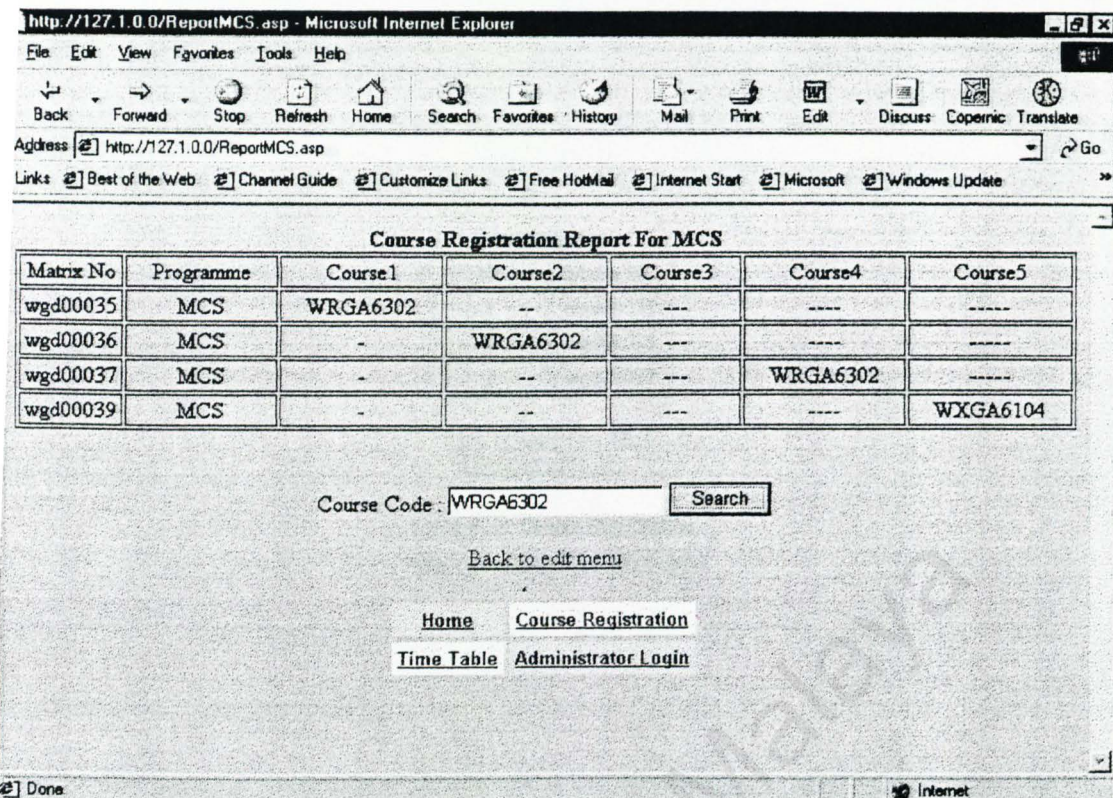
The course registration report main page is shown in below (Figure 34).



**Figure 34: Course Registration Report for FSKTM**

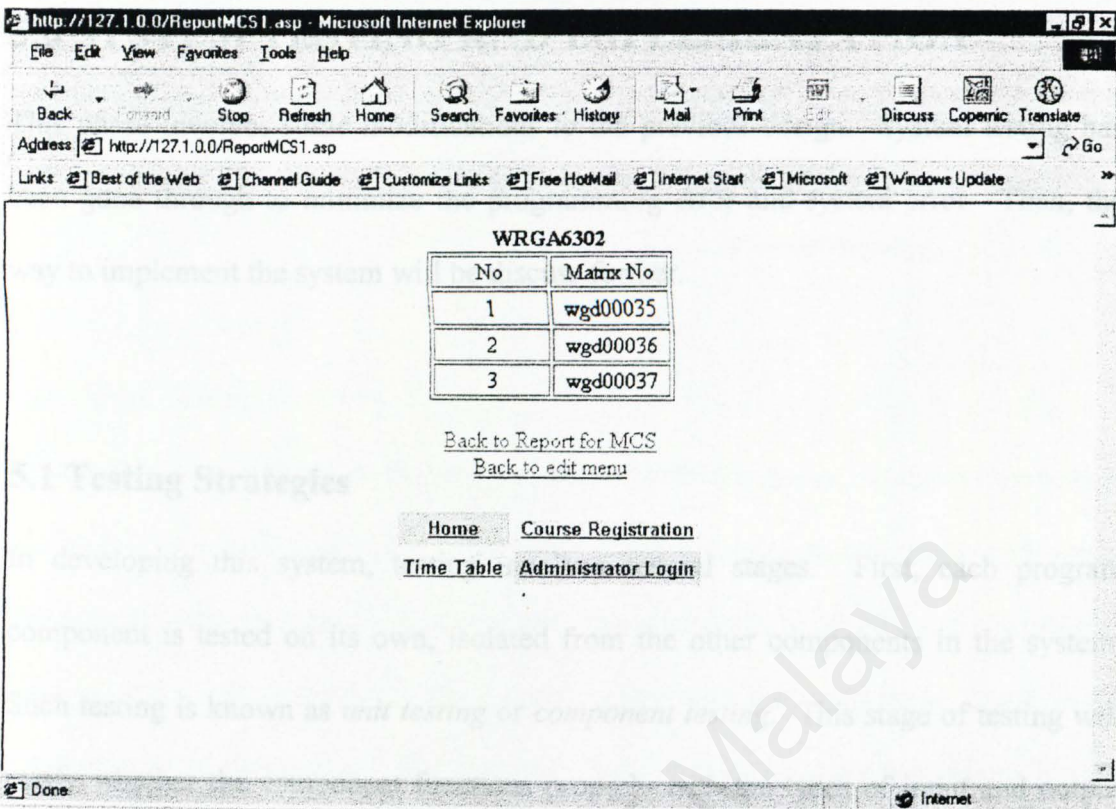
The page design for five different programmes (MIT, MSE, MCS, MLIS & PHD) is the same. Therefore, only Report for MCS is chosen for discussion. Figure 35 is an example of the page design for the MCS Report.





**Figure 35: Report for MCS**

In Figure 35, there is a table that shows all the courses registered by MCS students. Besides that, UM administrators can check which student and the number of students that register for a particular course. For example, if the UM administrators wants to check the number of students who registered for course WRGA6302, he need to type in the code for the course in the text box and click " Search " button. Then, the page as shown in Figure 36 will appear.



**Figure 36: Students that Register Course WRGA6302**

All the students that register course WRGA6302 will be shown in the table.



## 5.0 SYSTEM TESTING AND IMPLEMENTATION

This phase involves some modifications to the previous design. System testing has been gone through to minimize the programming error and system error. Then, the way to implement the system will be discuss further.

### 5.1 Unit Testing

#### 5.1 Testing Strategies

In developing this system, testing involves several stages. First, each program component is tested on its own, isolated from the other components in the system. Such testing is known as *unit testing* or *component testing*. This stage of testing will verify whether the component functions properly with the types of input and output expected from studying the component's design. After each component has been tested, the interaction between these components is tested again to ensure that the components can be integrated. (Dr.P.Sellapan, 2000)

(Kendall & Kendall, 1999)

When all components have been unit-tested, the next step is ensuring that the interfaces among the components are defined and handled properly. This step is called *integration testing*, also known as *module testing*, which verifies that the entire component can work together as described in the module or system design specifications.

Debugging would be carried out to identify the error before testing the unit again.

Finally, system testing is performed to ascertain that the whole system works according to the requirement description and specifications. System modification will be carried out if the requirements as described in the specification are not met.

## 5.2 Module Testing

### 5.1.1 Unit Testing

Unit testing concentrates on the smallest unit of software design, which is the module. A module is a collection of components, which are independent from each other. After each component unit has been tested, the interaction between these components will be tested. Unit testing is done concurrently with the prototyping phase in the development of this project. For example, "*edit postgraduate student record*" in the record is one of the sub-modules with many sub-functions. The insertion of student particulars and other functions (*delete & update*) are tested to ensure that the module is error free and the records can be saved and retrieved from database correctly. (Kendall & Kendall, 1999)

In the development of this system, unit testing is done after the development of each module and not the end of the whole system development. The object and programming codes are thoroughly checked and tested to ensure that the function and data are implemented properly as indicated in the design. If there were any error, debugging would be carried out to identify the error before testing the unit again.



Other unit that is independently unit-tested such as opening and closing of connection to the database are also important.

### 5.1.2 Module Testing

The objective of module testing (also known as integration testing) is to take unit-tested modules and build program structure that encapsulates all of the related modules. This testing will ensure that the module calling sequence in this project is systematic.

In module testing, two or more units in which either unit that use output data from or provide input data for another unit are tested in collection. These units have related characteristics to perform a common goal or function. (Kendall & Kendall, 1999)

All the ASP objects were integrated and tested to ensure the success of integration. Multiple values of test data were entered through the interface to ensure that values were inserted correctly into the database.

### 5.1.3 System Testing

The last testing procedure is *system testing*. Once all the modules are tested, they are tested in concert to verify that all the elements are functioning and interfacing with each other properly. The testing result will show whether or not the entire system specifications and objectives are achieved. (Kendall & Kendall, 1999)

## 5.2 System Testing

System testing to the applications that have been developing in the system is to satisfy the scope and the objectives. Modules that involved are:

- Main page module
- Student login module
- Student change password module
- Student's personal information list module
- Course selections module
- Course selected module
- Fees payment module
- Timetable module
- Administrator login module
- Administrator change password module
- View students' information module
- Edit menu module
- Edit students' information module



- Edit administrators' information module
- Edit registration fee and exam fee module
- Edit MIT course information module
- Edit MSE course information module
- Edit MCS course information module
- Edit MLIS course information module
- Edit PHD course information module
- Students access list module
- Administrators access list module
- Course registration reports module

### 5.2.1 Main page module testing

Testing on the link function of each hyperlinks to make sure the navigations are correct.

### 5.2.2 Student login module testing

In this module, enter the correct and false student's programme, ID and password to check whether can access to the next page. Then, the functionality of the buttons also been checked. The error message box will prompt out if the user makes an error.

### **5.2.3 Student change password module testing**

In this module, the correct student's programme, ID, existing password and new password have been entered to make sure the password is successfully changed. The system will prompt out the words " Password Successfully Changed " if the existing password has been changed.

### **5.2.4 Student's personal information list module testing**

Enter the new data in this module and make sure all the data can be updateable. Error message box will prompt out if user makes an error.

### **5.2.5 Course selections module testing**

This module is tested by select the same courses, and then, no course is selected. The error message box will prompt out and show users the error they have made.

### **5.2.6 Course selected module testing**

Check whether the course name and course code are the same as the one that have been selected. Then, check whether the total of course fees, credit hours and fees per credit hours are correct.



### **5.2.7 Fees payment module testing**

In this module, try to enter the false information in the text box to see whether the error message box will prompt out. If all the information that entered is correct, then the system will store all the information in the database. The system will tell the users that they have successfully done the registration.

### **5.2.8 Timetable module testing**

Testing on the links to every timetable to make sure all the links will go to the correct timetables.

### **5.2.9 Administrator login module testing**

In this module, enter the correct and false administrator's status, ID and password to check whether can access to the next page. Then, the functionality of the buttons also been checked.

### **5.2.10 Administrator change password module testing**

In this module, the correct administrator's status, ID, existing password and new password have been entered to make sure the password is successfully changed.

#### **5.2.11 View students' information module testing**

The correct and false students' matrix number have been entered to test whether the system is showing the correct students record.

#### **5.2.12 Edit menu module testing**

Testing on the link function of each hyperlinks to make sure the navigations are correct.

#### **5.2.13 Edit students' information module testing**

In this module, make sure the particular student's record can be updated and deleted. Besides that, make sure add the new student function is workable. The new student's record must be store in the database. Retrieve the information (new student record) back after the adding process to check whether the new record is correct. Make sure the correct error message box prompt out when make mistake.

#### **5.2.14 Edit administrators' information module testing**

In this module, make sure the particular administrator's record can be updated, deleted and added. Make sure the correct error message box prompt out when make mistake.



#### **5.2.15 Edit registration fee and exam fee module testing**

Enter the different figures of registration fee and exam fee in the text boxes and update the information. Then check in the course-selected module whether the figure of registration fee and exam fee are the same.

#### **5.2.16 Edit MIT course information module testing**

In this module, check the courses can be edited, updated, deleted and added. Make sure the correct error message box prompt out when make mistake.

#### **5.2.17 Edit MSE course information module testing**

In this module, check the courses can be edited, updated, deleted and added. Make sure the correct error message box prompt out when make mistake.

#### **5.2.18 Edit MCS course information module testing**

In this module, check the courses can be edited, updated, deleted and added. Make sure the correct error message box prompt out when make mistake.

#### **5.2.19 Edit MLIS course information module testing**

In this module, check the courses can be edited, updated, deleted and added. Make sure the correct error message box prompt out when make mistake.

#### 5.2.20 Edit PHD course information module testing

In this module, check the courses can be edited, updated, deleted and added. Make sure the correct error message box prompt out when make mistake.

#### 5.2.21 Students access list module

In this module, check whether the access time to the system is correct. Then, test the delete record function is workable.

#### 5.2.22 Administrators access list module

In this module, check whether the access time to the system is correct. Then, test the delete record function is workable.

#### 5.2.23 Course registration reports module

In this module, the link functions of every hyperlink are tested. Make sure the link functions will link to the correct pages. Then, the workable of the search function in the module has been tested. Enter the course code and then check whether the correct students are registering that particular course.



### 5.3 System Implementation Registration and Payment System Main Page

System implementation is a process that converts the system requirements and designs into program codes. Online Postgraduate Course Registration and Payment System is developed using the top-down approach which involves building the high-level system modules which are further refined into function and procedures. Therefore, this system can be implemented by a few module and sub module.

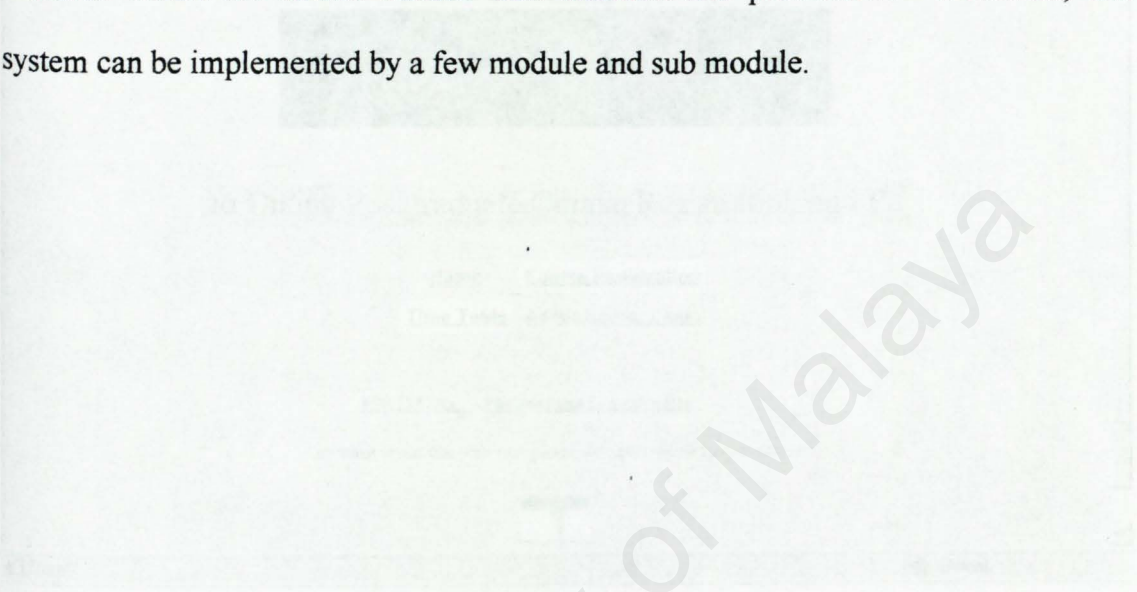
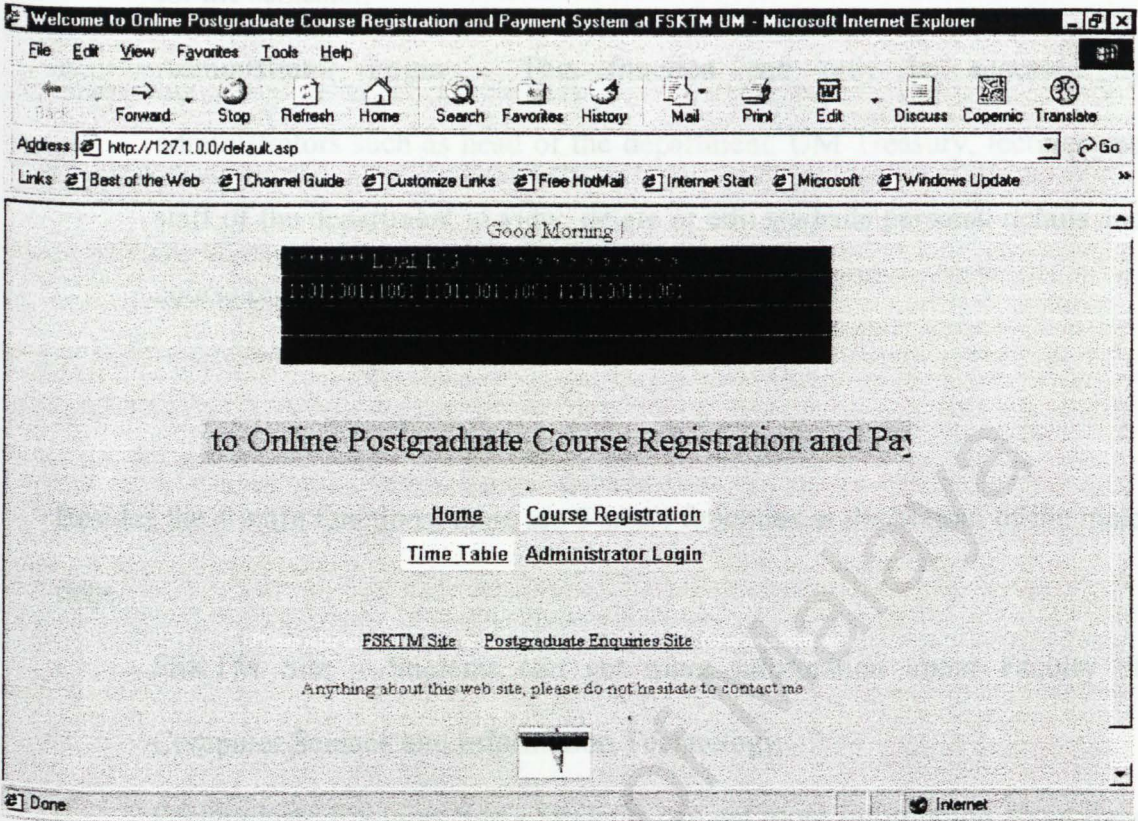


Figure 37: Online Postgraduate Course Registration and Payment System Main

The main page shows a menu that contains 4 main functions that can be chosen.

- I. Home - This function allows the students to return to the main page of the Online Postgraduate Course Registration and Payment System.
- II. Course Registration - This function allows the students to register for the courses they want to study for a particular semester. Students can also update their personal details. After registration, students can pay their fees online.

### 5.3.1 Online Postgraduate Course Registration and Payment System Main Page



**Figure 37: Online Postgraduate Course Registration and Payment System Main Page**

The main page shows a menu that contains 4 main functions that can be chosen.

- i. Home – This function allows the students to return to the main page of the Online Postgraduate Course Registration and Payment System.
- ii. Course Registration – This function allows the students to register for the courses they want to study for a particular semester. Students can also update their personal details. After registration, students can pay their fees online.



- iii. Time Table – This function allows the students to view the latest timetable for the semester.
- iv. Administrator Login – This function can only be accessed by administrators such as head of the department, UM Treasury, lecturer and staff of the department to view, delete or edit students personal details and course registration information.

Besides the 4 main functions, there are 3 small functions at the bottom of the main page.

- i. FSKTM Site – Students can get more information about Faculty of Computer Science and Information Technology.
- ii. Postgraduate Enquires Site - Students can get more information about their program in this site.
- iii. Email Icon – Students can email to that email account if they have any questions.

### **5.3.2 Course Registration**

To start the course registration, click on the word “ *Course Registration* ” at the menu.

The following page will appear.

Student Login - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Refresh Home Search Favorites History Mail Print Edit Discuss Copernic Translate

Address <http://127.1.0.0/studentlogin.asp> Go

Links: [Best of the Web](#) [Channel Guide](#) [Customize Links](#) [Free HotMail](#) [Internet Start](#) [Microsoft](#) [Windows Update](#)

## Student Login

Programme:

ID:  (lower case and no space)

Password:

Login:

[Home](#) [Course Registration](#)  
[Time Table](#) [Administrator Login](#)

Internet

**Figure 38: Student Login**

### 5.3.2.1 Change Password

- i. To change old password, click on the “*Change Password*” button.



ii. User will see a change password page as shown below.

Change Password - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Refresh Home Search Favorites History Mail Print Edit Discuss Copernic Translate

Address <http://127.1.1.0/studentchangepassword1.asp> Go

Links [Best of the Web](#) [Channel Guide](#) [Customize Links](#) [Free HotMail](#) [Internet Start](#) [Microsoft](#) [Windows Update](#)

## Change Password

ID :

Password :

New Password :

Re-type New Password :

Done Internet

**Figure 39: Change Password**

ii. Fill in the information in the text box and click on “*Submit*” button.

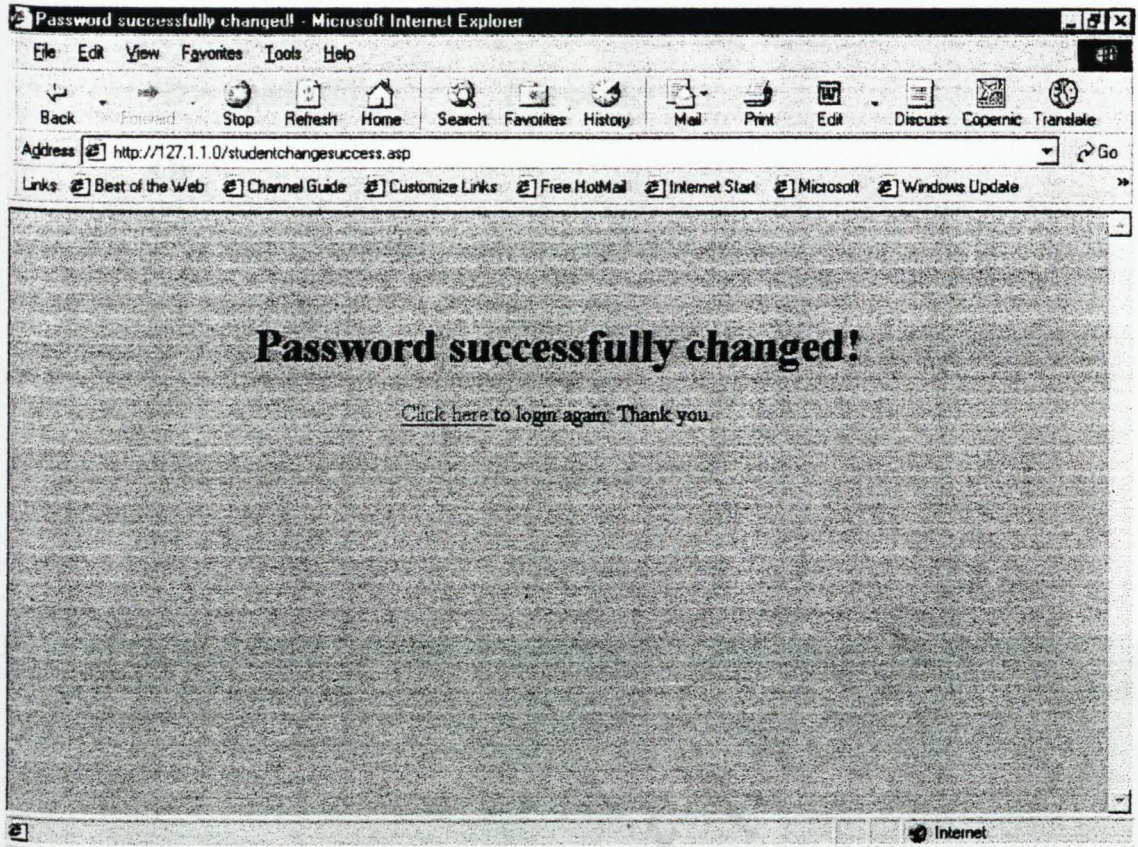
Try using the ID and password below. Then, try to create own new password.

ID = wgd00030 and Password = 123 or

ID = wgd00031 and Password = 123

If all the information entered is correct, then a “Password Successfully Changed” page will appear next.





**Figure 40: Password Successfully Changed**

- iv. Then click on the word “ *Click here* ” to login again with the new password.
- v. If information entered in the text is not complete, the password that entered in the text box does not match with the ID or the two new passwords in the text box are not the same, then have to try again.

**Figure 41: Student Login Page**

#### **5.3.2.2 Student Login**

- i. User have to login with your ID and password as shows in Figure 41. For example,



Programme = Master of Information Technology

ID = wgd00030

Password = 123

Or

Programme = Master of Software Engineering

ID = wgd00031

Password = 123

Or

Programme = Master of Computer Science

ID = wgd00035

Password = 123

Student Login - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Refresh Home Search Favorites History Mail Print Edit Discuss Copernic Translate

Address <http://127.1.0.0/studentlogin.asp> Go

Links [Best of the Web](#) [Channel Guide](#) [Customize Links](#) [Free HotMail](#) [Internet Start](#) [Microsoft](#) [Windows Update](#)

## Student Login

Programme:

ID:  (lower case and no space)

Password:

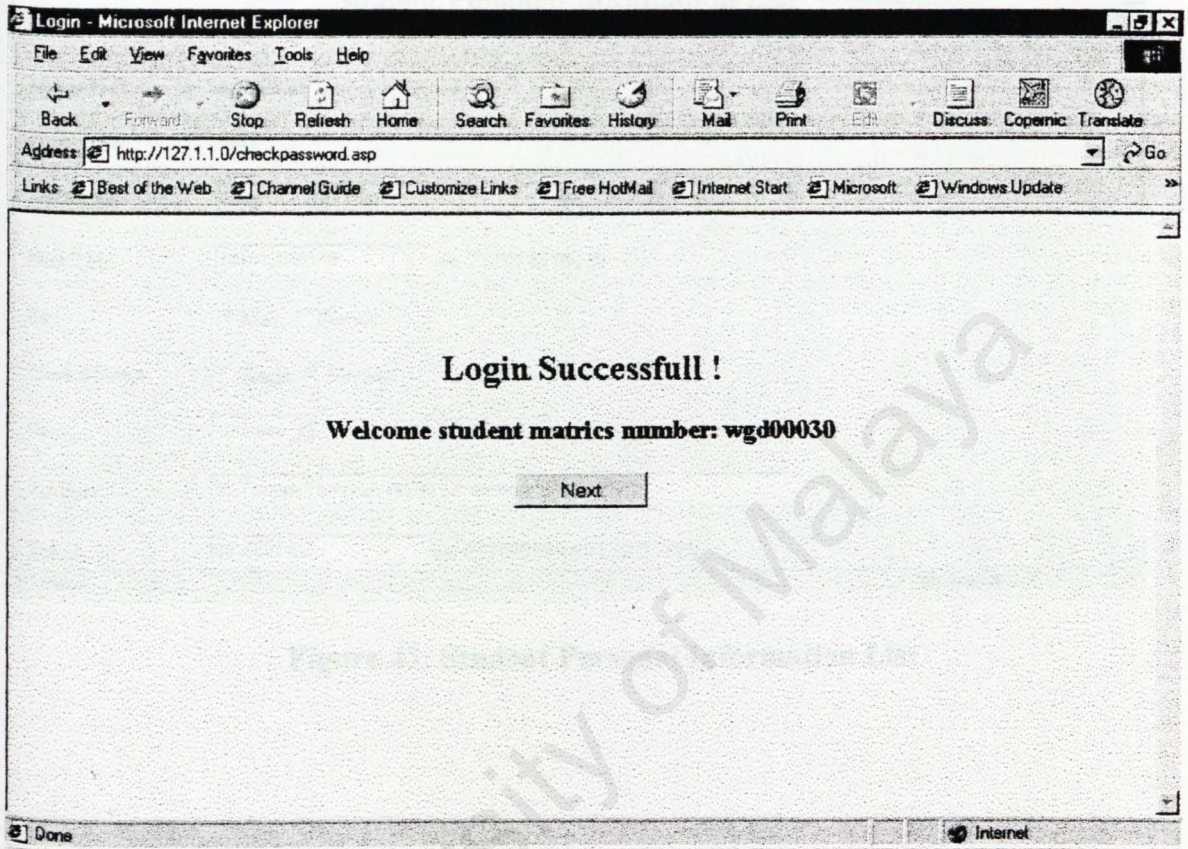
[Home](#) [Course Registration](#)  
[Time Table](#) [Administrator Login](#)

Done Internet

Figure 41: Student Login Page



- ii. If the programme, ID, and password that entered do not match with each other, then access will be denied. If user has entered the correct information, the following page (Figure 42) will appear.



**Figure 42: Successful Login**

- iii. Click on the "Next" button and the page as shown in Figure 43 will appear. This page shows the particular student's personal information. User can update the record if you want to do so. An error message box will prompt out of any mistake that make during updating the record.



Personal Information List - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Refresh Home Search Favorites History Mail Print Discuss Copernic Translate

Address <http://127.1.0.0/studentinfo.asp> Go

Links [Best of the Web](#) [Channel Guide](#) [Customize Links](#) [Free HotMail](#) [Internet Start](#) [Microsoft](#) [Windows Update](#)

### Student Personal Information List

(Please correct the information if it is wrong !)

Matrix No : wgd00030

Programme : MIT

Full Name :

NRIC No :  ex: 761013478139

Sex : ☒ Male ☐ Female

Marital Status : ☐ Single ☒ Married

Race :

Address :

Telephone :  ex: 067698899 / 0126377888

Done Internet

Figure 43: Student Personal Information List

### 5.3.2.3 Student Personal Information

- After that, go to the Course Selections page. Figure 44 and Figure 45 show the courses are offered in the semester.

Course Selections - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Refresh Home Search Favorites History Mail Print Edit Discuss Copernic Translate

Address http://127.1.1.0/courseable.asp Go

Links Best of the Web Channel Guide Customize Links Free HotMail Internet Start Microsoft Windows Update

MIT Course Selections				
Course Name	Course Code	Credit Hours	Fees Per Credit (RM)	Total Fees (RM)
Visual Programming : Concept and Application	WXGE6104	3	210	630
Multimedia Technology	WXGE6309	3	210	630
Web Commerce	WXGE6306	3	210	630
Knowledge Management	WXGE6308	3	210	630
Database Concepts and Implementation	WXGE6101	3	210	630
Software				

Done Internet

Figure 44: Course Selections

### 5.3.2.4 Course Selections

User must choose at least one course up to a maximum of 5 courses. If they did not choose any course, or choose the same courses, then an error message box will appear to inform the error that user has made. Therefore, try again.

After the course(s) is selected, click on "Next" button, the page like Figure 45 will appear. The system will calculate the total course fee automatically. The total course fee is inclusive of registration fee and examination fee.



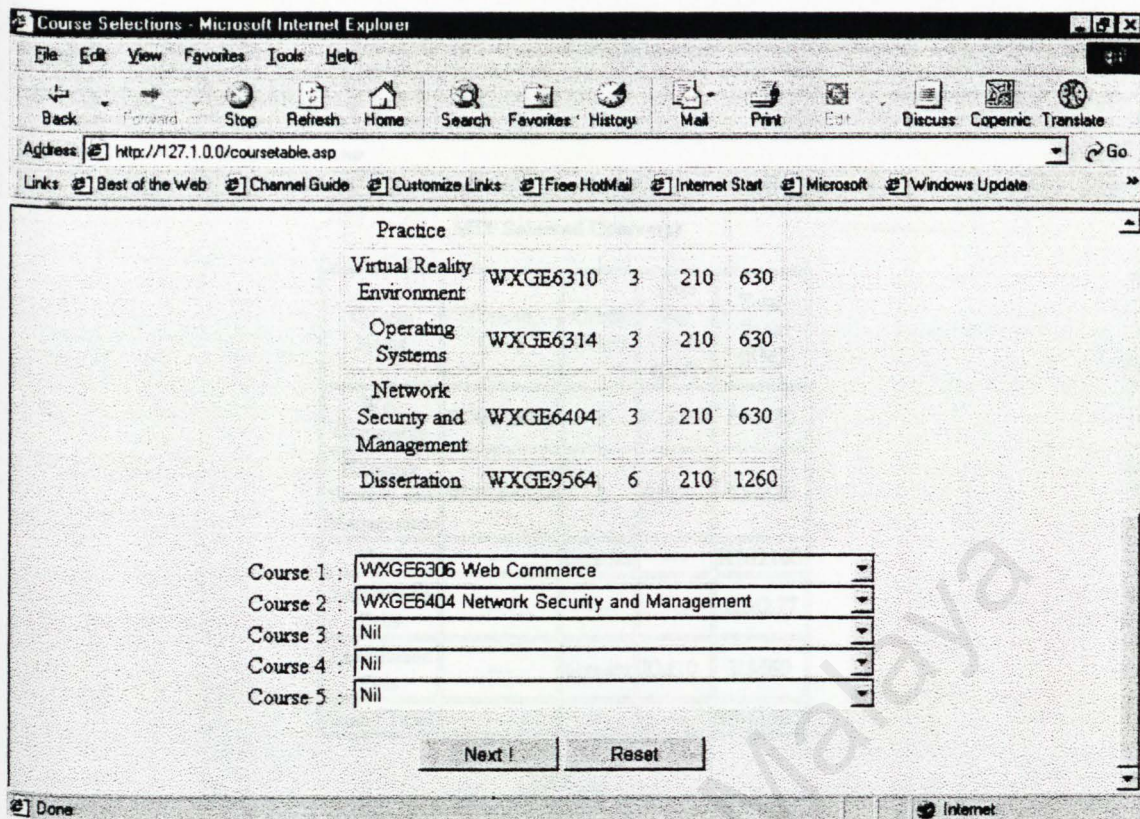
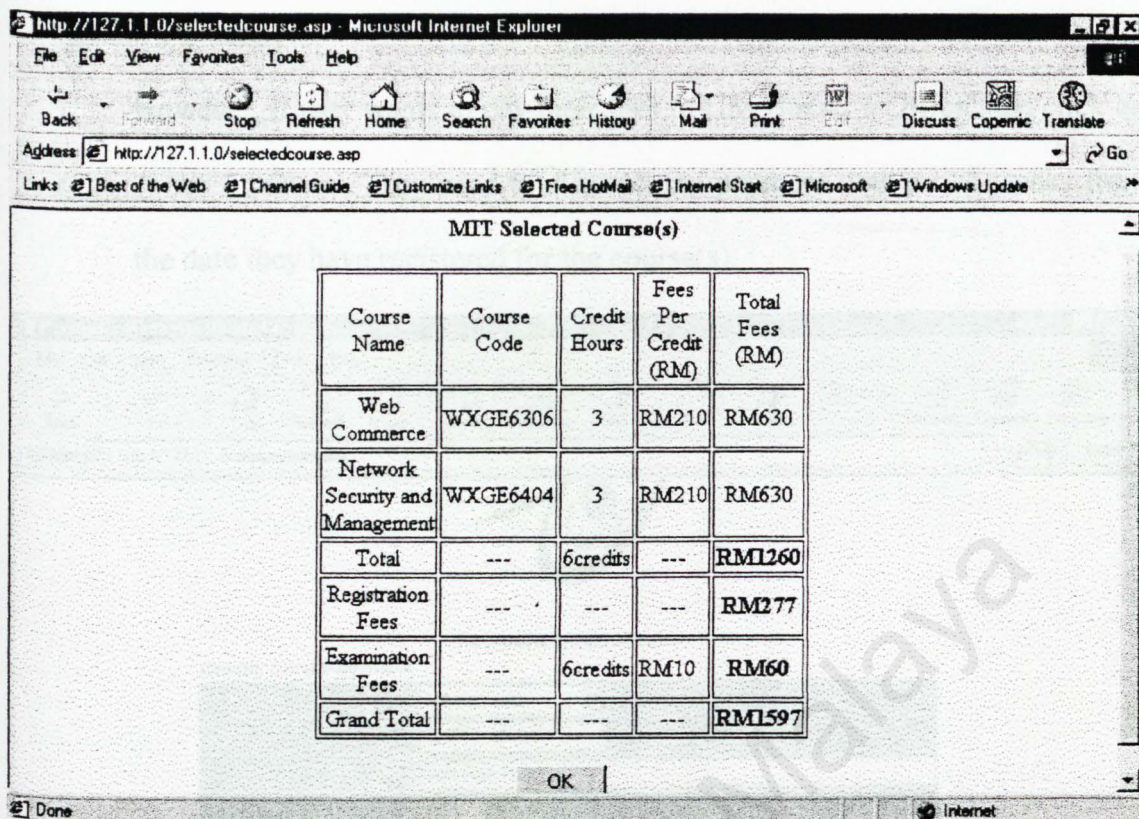


Figure 45: (Continue from Figure 44)

### 5.3.2.4 Course Selections

- i. User must choose at least one course up to a maximum of 5 courses. If they did not choose any course, or choose the same courses, then an error message box will appear to tell the error that user has made. Therefore, try again.
- ii. After the course(s) is selected, click on “Next” button, the page like Figure 46 will appear. The system will calculate the total course fee automatically. The total course fee is inclusive of registration fee and examination fee.





**Figure 46: Selected Course(s)**

### 5.3.2.5 Selected Course(s)

- i. After that, click on the "OK" button, user will go to the payment page as shown in Figure 47. Here, user have two choices; either pay for your fees via visa or master card or pay in cash to the UM Bursar's Office. If they want to pay via visa or master card, then fill in the entries and click on the "Submit" button. An error message box will appear to alert user if they make any error.



5.3.2.6 Payment

- i. However, if user wants to pay their fees to UM Bursar’s Office, just click on “ OK “ button. They need to clear all the payment within two weeks from the date they have registered for the course(s).

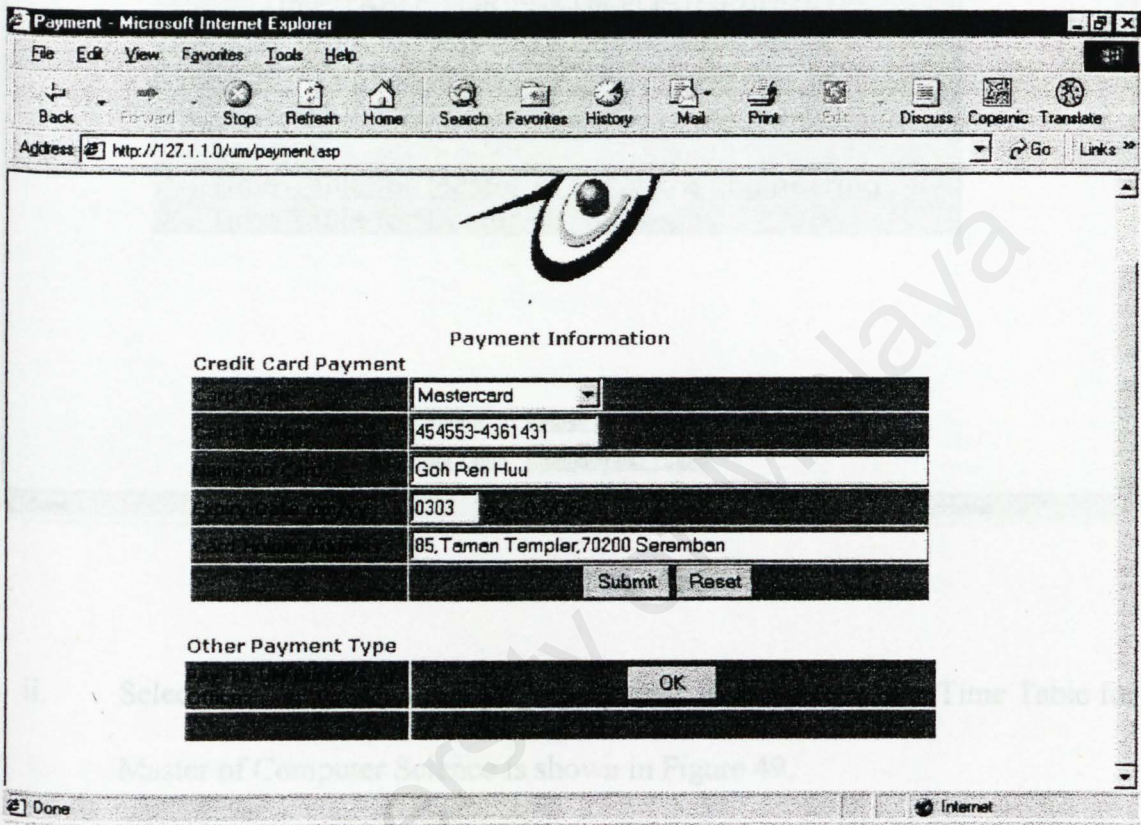
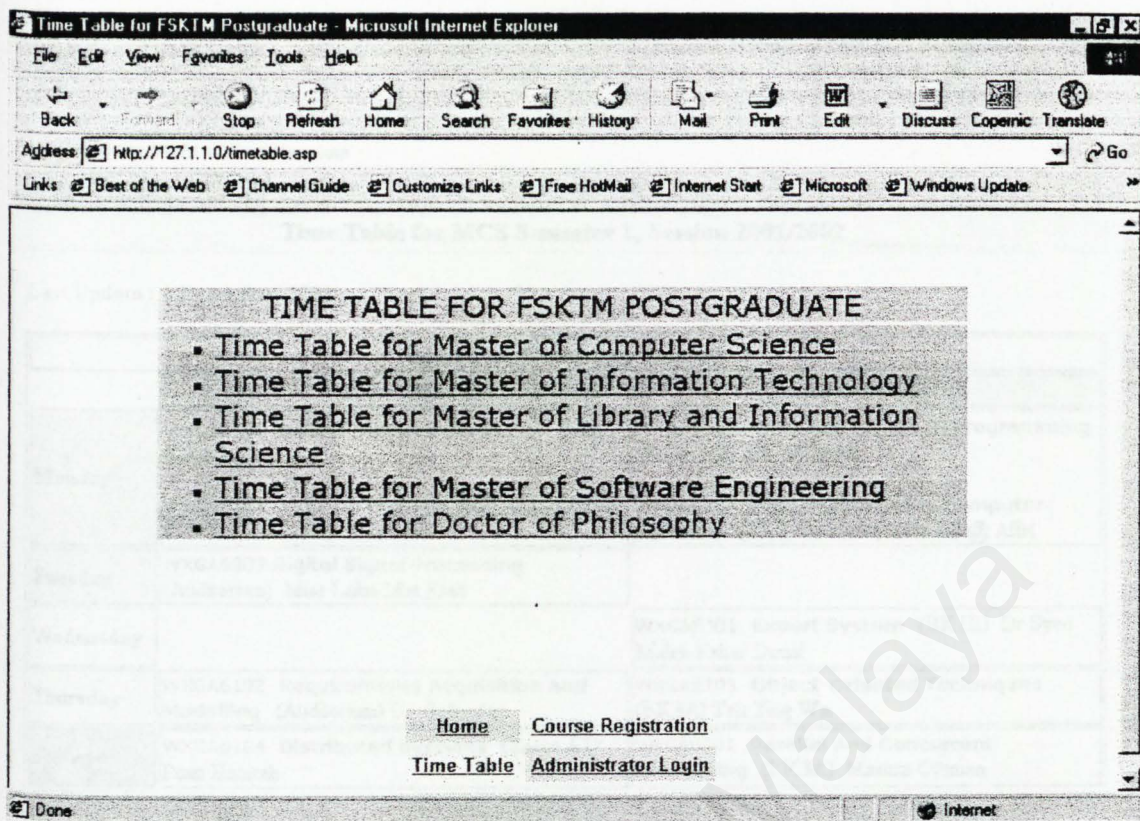


Figure 47: Payment Information

5.3.3 Time Table

- i. To view the latest timetable, click on the word “ Time Table “ on the menu. The following page will appear (Figure 48).



**Figure 48: Time Table**

- ii. Select the timetable that user want to view. For example, the Time Table for Master of Computer Science is shown in Figure 49.



Master of Computer Science - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Refresh Home Search Favorites History Mail Print Edit Discuss Copernic Translate

Address: http://127.1.1.0/timetablencs.asp Go

Links: Best of the Web Channel Guide Customize Links Free HotMail Internet Start Microsoft Windows Update

**Time Table for MCS Semester 2, Session 2001/2002**

Last Update : 1 December, 2001

Master of Computer Science		
	6.00-9.00pm	6.30-9.30pm
<b>Monday</b>	WXGA6302 Neural Network (Room A1) Puan Hanizah WRGA6302 VLSI Technology (BK3A) Teh Ying Wai	WXGA6101 Advanced Issues In Programming (Room A2) Dr. Sellappan WRGA6304 High Performance Computer Systems (BK2) Pn Hannyzura Pal @ Affal
<b>Tuesday</b>	WXGA6303 Digital Signal Processing (Auditorium) Miss Laiha Mat Kiah	
<b>Wednesday</b>		WXGA6301 Expert System (BK1B) Dr Syed Malek Fakar Duani
<b>Thursday</b>	WXGA6102 Requirements Acquisition And Modelling (Auditorium) Dr. Sellappan	WXGA6103 Object-Oriented Techniques (BK3A) Teh Ying Wai
<b>Friday</b>	WXGA6104 Distributed Systems (Room A1) Puan Hanizah	WRGA6301 Parallel And Concurrent Processing (BK3A) Mazliza Othman

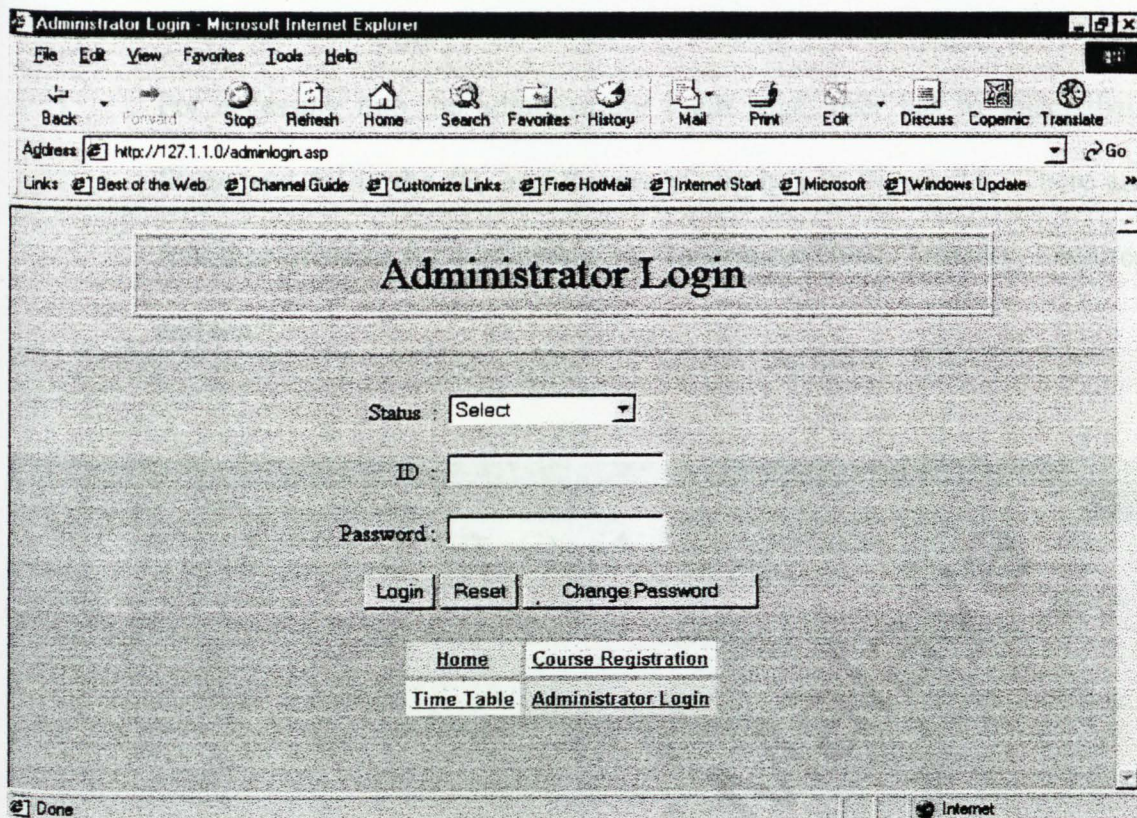
Done Internet

Figure 49: Time Table for Master of Computer Science

### 5.3.4 Administrator Login

For the administrator to login into the system, just click the word “ *Administrator Login* ” at the menu. The following page will appear (Figure 50).





**Figure 50: Administrator Login**

### 5.3.4.1 Change Password

- i. Click on the “*Change Password*” button, the steps to change password for administrators are the same as for the students. Please refer to **5.3.2.1** and **Figure 39**. Try using ID and password given below in the “*Change Password*” page. Then create new password.

ID = 111 and Password=111 or

ID = 222 and Password=222 or

ID = 333 and Password=333



5.3.4.2 Administrator Login

- i. For accessing into the administrator login page, user has to select their *Status* and fill in the *ID*, and *Password* text box in Figure 51. There are four type of status for administrators: Department Head, Treasury, Lecturer and Staff.

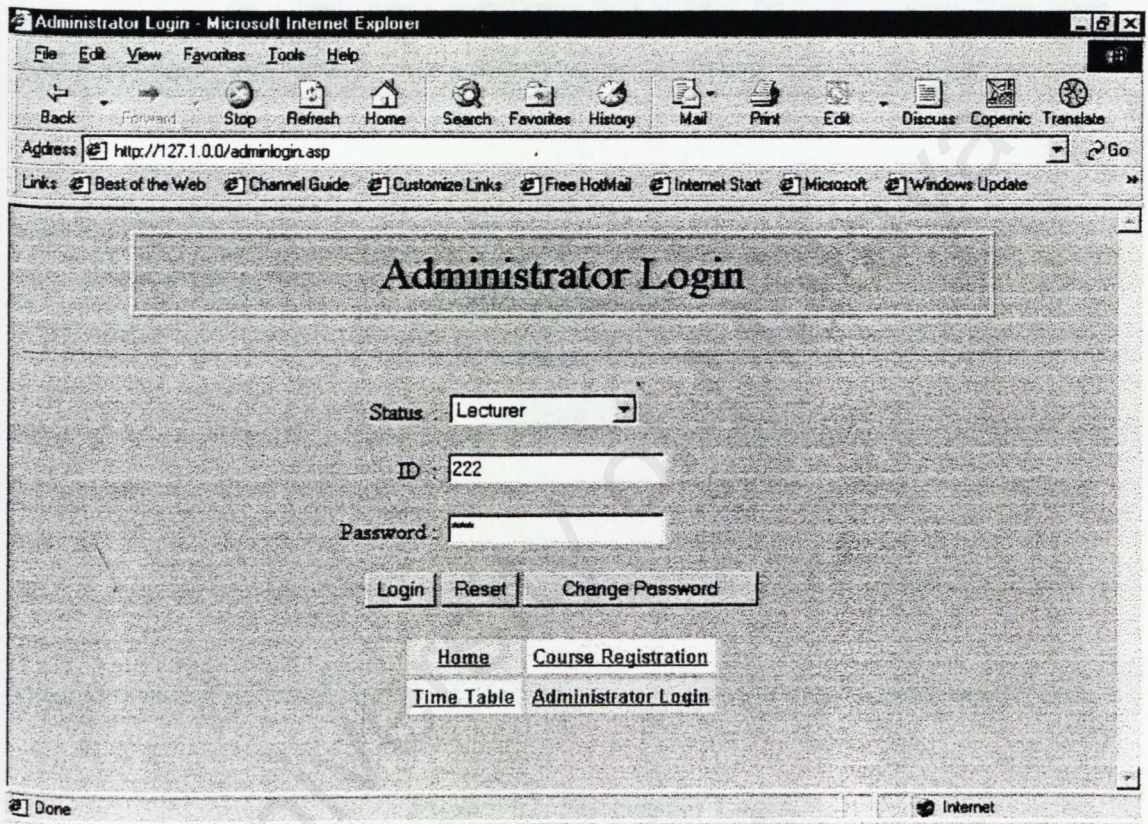


Figure 51: Administrator Login

- ii. Please login using the information given below.

Status: Department Head	ID: 111	Password: 111
Status: Treasury	ID: 555	Password: 555
Status: Lecturer	ID: 222	Password: 222



Status: Staff

ID: 333

Password: 333

- iii. If user's *Status*, *ID*, and *Password* do not match, access will be denied. If all the information is correct, Figure 52 will appear.

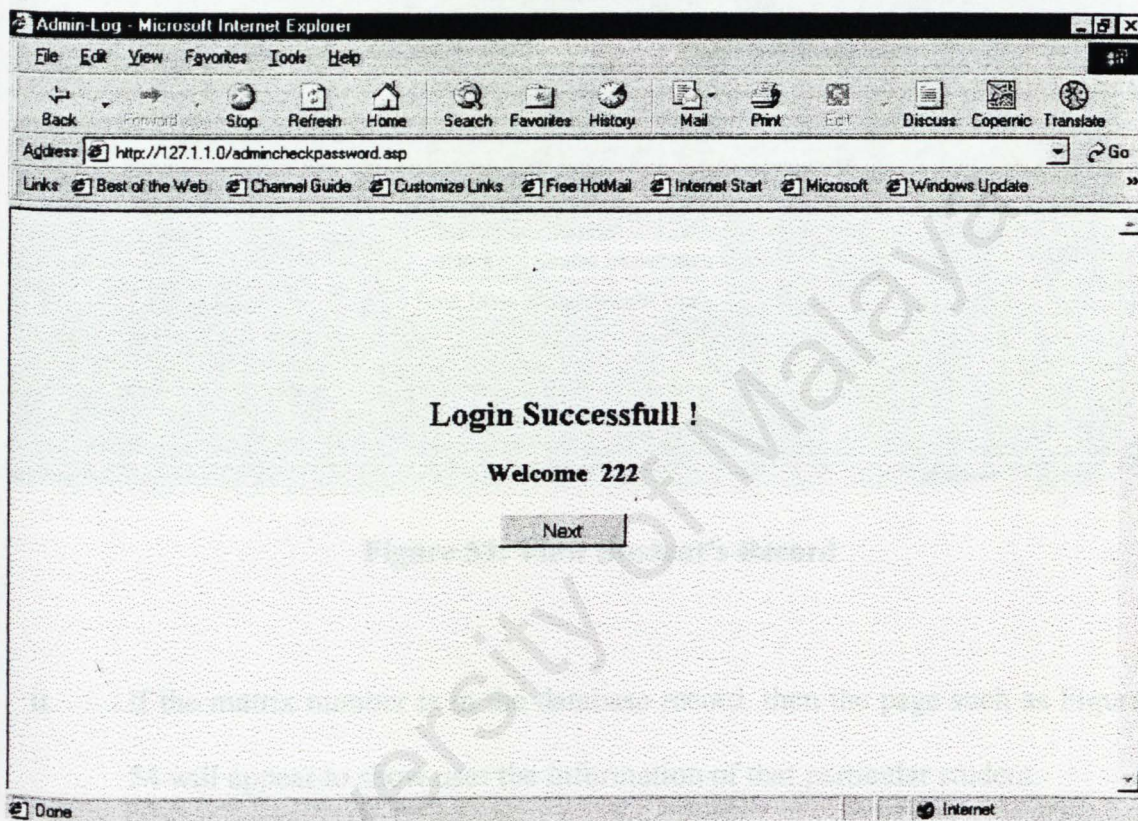
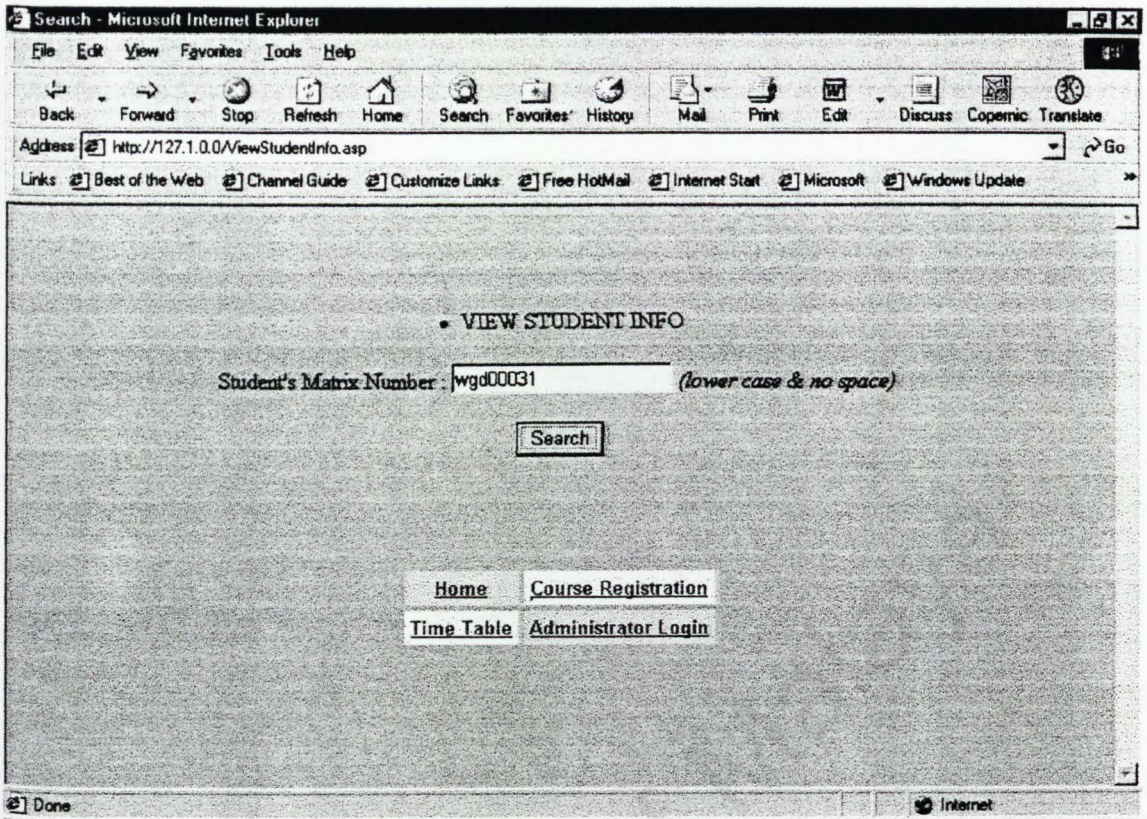


Figure 52: Successful Login

#### 5.3.4.3 View Student's Record

- i. If user is a lecturer or staff, they will go to the page shown in Figure 53. Here, they can only view the student record but cannot edit it. Key in the student matrix number for the records of the student they which to see. For example, wgd00030.

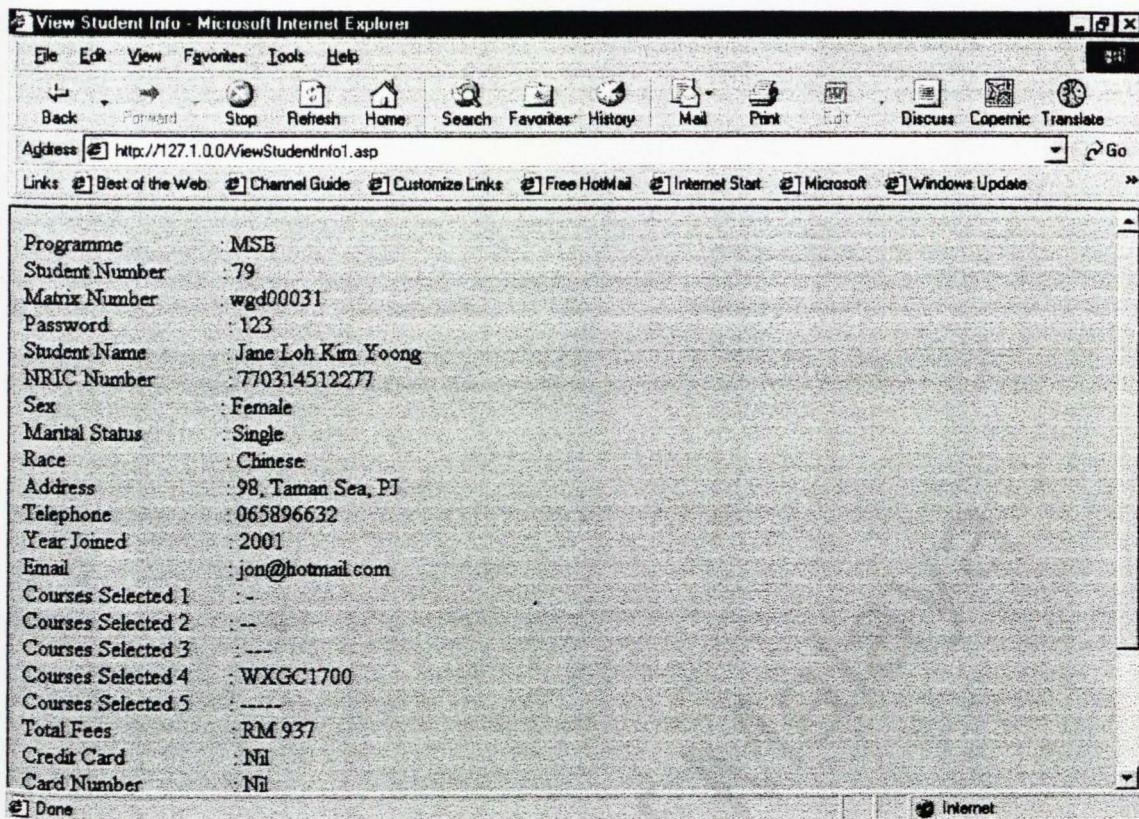




**Figure 53: View Student's Record**

- ii. If the matrix number is in the database record, then the page such as Figure 54 will appear to show user the information of that particular student.



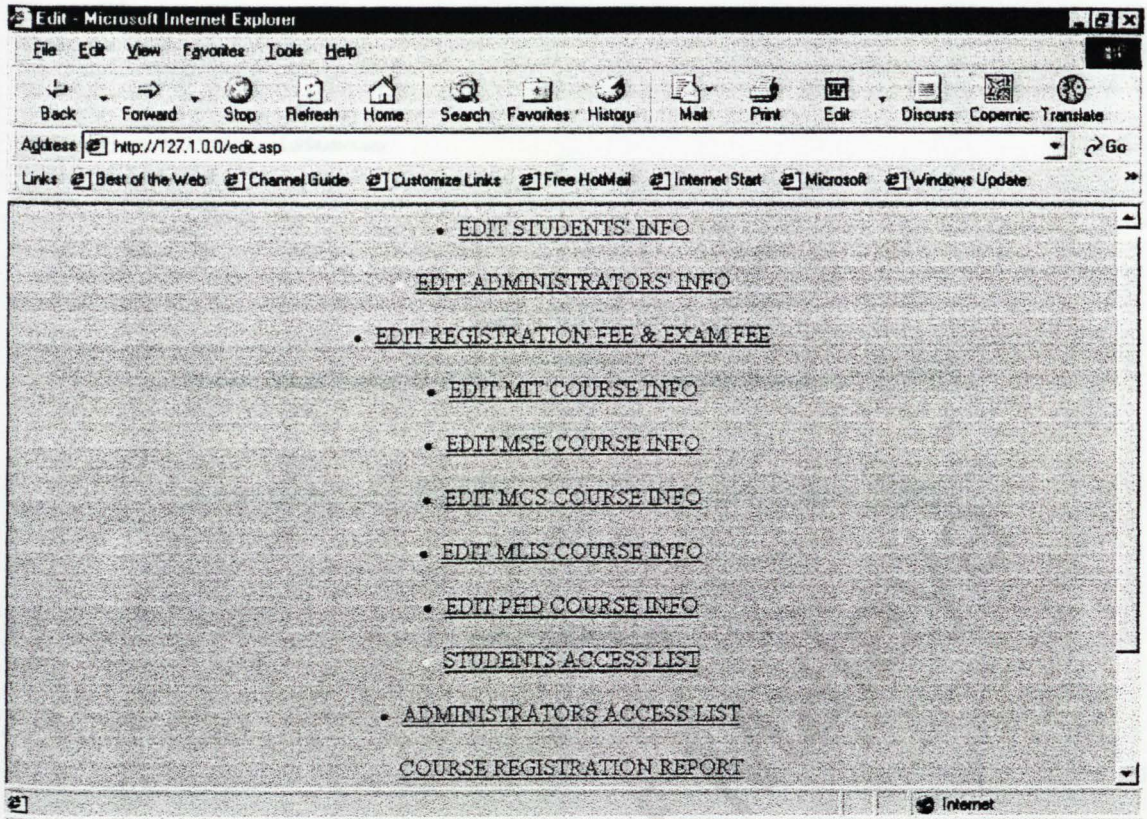


**Figure 54: Student's Record**

#### 5.3.4.4 Edit Menu

- i. If user is a Department Head or UM Bursar's Officer, they will go to the page as shown by Figure 55. There are 11 choices for them to choose; Edit Students' Info, Edit Administrators' Info, Edit Registration Fee and Exam Fee, Course Registration Report, Edit MIT Course Info, Edit MSE Course Info, Edit MCS Course Info, Edit MLIS Course Info, Edit PHD Course Info, Student Access List and Administrator Access List.

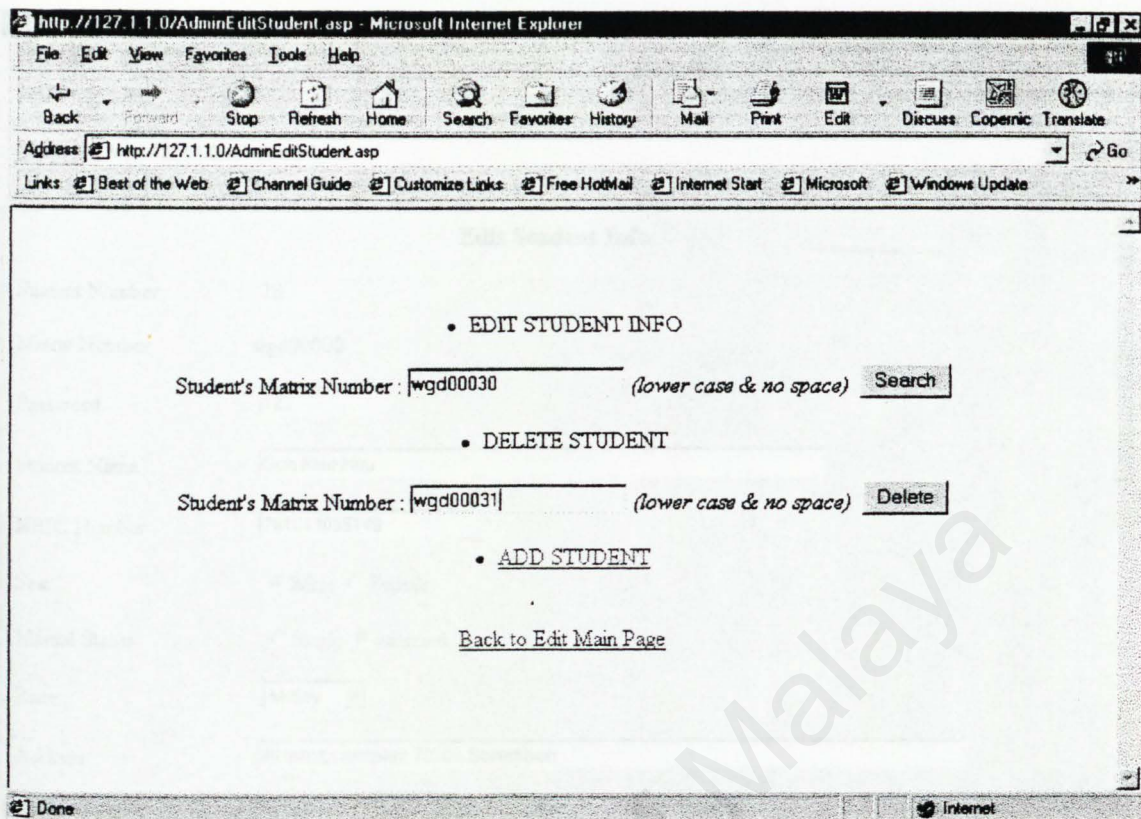




**Figure 55: Edit Menu**

#### **5.3.4.4.1 Edit Students' Information**

- i. If user clicks on " *Edit Student* ", they will go to Figure 56.



**Figure 56: Edit Student's Record**

- ii. If they want to edit or update student record, then enter the student's matrix number in the text box and click on the " *Search* " button as shown in Figure 56. Then, the page shown by Figure 57 will appear.



**Edit Student Info**

Student Number : 78

Matrix Number : wgd00030

Password : 123

Student Name : Goh Ren Huu

NRIC Number : 761013055149

Sex : ☒ Male ☐ Female

Marital Status : ☐ Single ☒ Married

Race : Malay

Address : 85, taman templer, 70200 Seremban

**Figure 57: Student's Record**

- iii. After editing, click on the “*Update*” button to update the record.
- iv. If they want to delete student record, then enter student matrix number in the text box and click on the “*Delete*” button as shown in Figure 56. The particular student's record will be deleted.
- v. However, if they want to add a new student in the record, click on the word “*Add Student*” as shown in Figure 56. User will come to Figure 58.



The screenshot shows a Microsoft Internet Explorer window titled "Add New Student - Microsoft Internet Explorer". The address bar displays "http://127.1.1.0/AddStudent.asp". The browser's menu bar includes File, Edit, View, Favorites, Tools, and Help. The toolbar contains icons for Back, Forward, Stop, Refresh, Home, Search, Favorites, History, Mail, Print, Edit, Discuss, Copernic, and Translate. The Links bar shows shortcuts to Best of the Web, Channel Guide, Customize Links, Free HotMail, Internet Start, Microsoft, and Windows Update. The main content area is titled "Add New Student" and contains the following form fields:

- Matrix No:
- Password:
- Programme:
- Full Name:
- NRIC No:
- Sex: ☐ Male ☐ Female
- Marital Status: ☐ Single ☐ Married
- Race:
- Address:

The status bar at the bottom shows "Done" and "Internet".

Figure 58: Add New Student

- vi. Click on the "Add" button after user completed all the entries.

#### 5.3.4.4.2 Edit Administrators' Information

- i. Click on the "Edit Admin" at Figure 55, user will come to Figure 59.



Admin Personal Info - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Refresh Home Search Favorites History Mail Print Edit Discuss Copernic Translate

Address http://127.1.1.0/AdminEditAdmin.asp Go

Links Best of the Web Channel Guide Customize Links Free HotMail Internet Start Microsoft Windows Update

### Administrator Personal Information

Admin No	ID	Password	Status	Name
4	111	111	Department Head	Mohammad Ali
5	222	222	Lecturer	Fatimah binti Mohammad Nor
6	333	333	Staff	Siti Hajar
7	444	444	Department Head	Sharizan bin Mohammad Nor
8	555	555	Treasurer	Debbie Goh Tian Yee

ID:

ID:

ID:

Password:

Name:

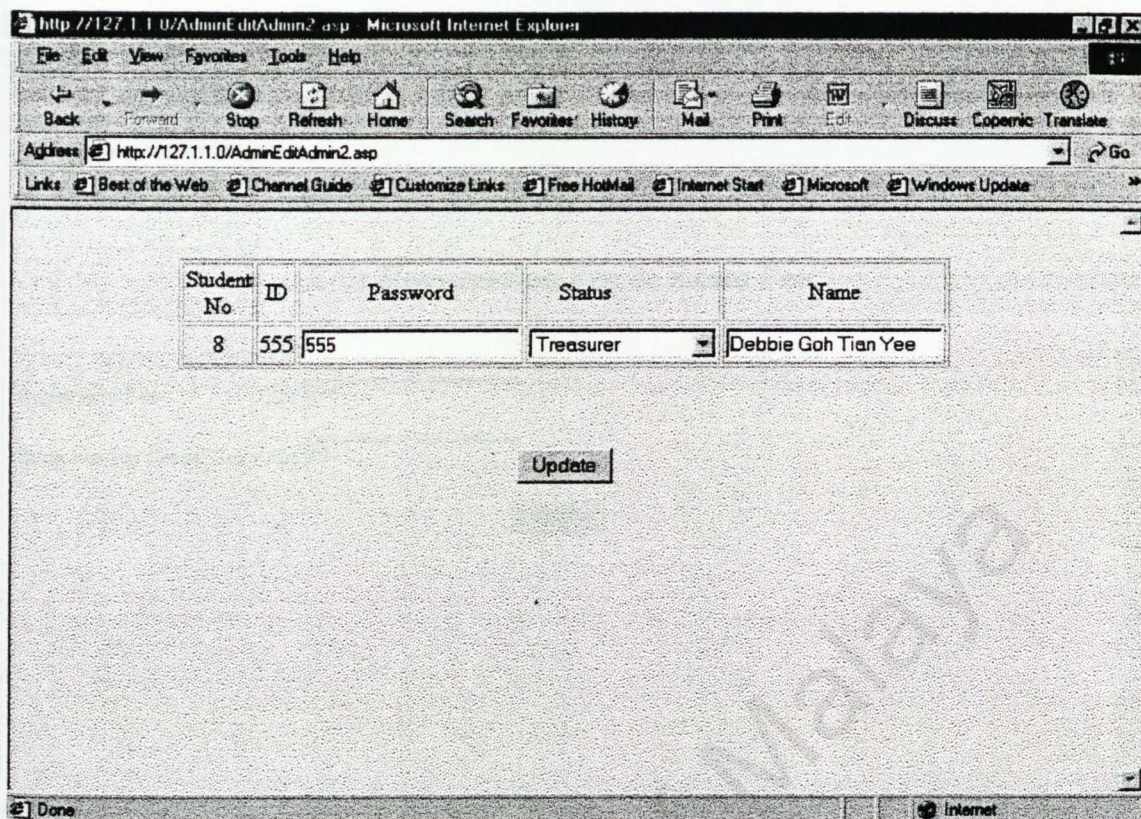
Status:

Done Internet

**Figure 59: Administrators' Records**

- ii. If they want to edit or update the record, enter the admin ID in the text box and click on the "Edit" button. They will go to Figure 30.





**Figure 60: Edit/Update Record**

- iii. After editing / updating record, click on the “*Update*” button.
- iv. If they want to delete a record, enter the admin ID and click the “*Delete*” button as shown in Figure 59.
- v. If they want to add a new administrator, complete the 5 entries as shown in Figure 59 and click the “*Insert*” button. The record will add in database.

#### 5.3.4.4.3 Edit Registration Fee and Examination Fee

- i. Click on “*Edit Registration Fee and Examination Fee*” as shown in Figure 55. Then they will go to Figure 61.



**Edit Registration Fee & Exam Fee**

Registration Fee : RM 277

Exam Fee Per Credit Hour : RM 10

**Figure 61: Edit Registration Fee and Examination Fee**

- ii. Change the figure in the text box and then click the “ *Update* ” button.  
The record will be updated.

#### **5.3.4.4.4 Edit MIT Course Info/ Edit MSE Course Info/ Edit MCS Course Info/ Edit MLIS Course Info/ Edit PHD Course Info**

- i. The steps to edit the course info are the same for all *programmes*. Therefore, only the steps for “*Edit MIT Course Info* ” will be discussed.
- ii. Click on “*Edit MIT Course Info* ” as shown in Figure 55. Then, user will see Figure 62 and Figure 63 (Continue from Figure 62).



MIT

Course No	Course Name	Course Code	Credit Hours	Fees Per Credit (RM)
mit1	Visual Programming : Concept and Application	WXGE6104	3	210
mit2	Multimedia Technology	WXGE6309	3	210
mit3	Web Commerce	WXGE6306	3	210
mit4	Knowledge Management	WXGE6308	3	210
mit5	Database Concepts and Implementation	WXGE6101	3	210
mit6	Software Engineering Process and Practice	WXGE6103	3	210
mit7	Virtual Reality Environment	WXGE6310	3	210
mit8	Operating Systems	WXGE6314	3	210
mit9	Network Security and Management	WXGE6404	3	210
mit10	Dissertation	WXGE6899	6	210

Course No :

**Figure 62: Edit MIT Course**



mit4	Knowledge Management	WXGE6308	3	210
mit5	Database Concepts and Implementation	WXGE6101	3	210
mit6	Software Engineering Process and Practice	WXGE6103	3	210
mit7	Virtual Reality Environment	WXGE6310	3	210
mit8	Operating Systems	WXGE6314	3	210
mit9	Network Security and Management	WXGE6404	3	210
mit10	Dissertation	WXGE6899	6	210

Course No :

Course No :

Course No :

Course Name :

Course Code :

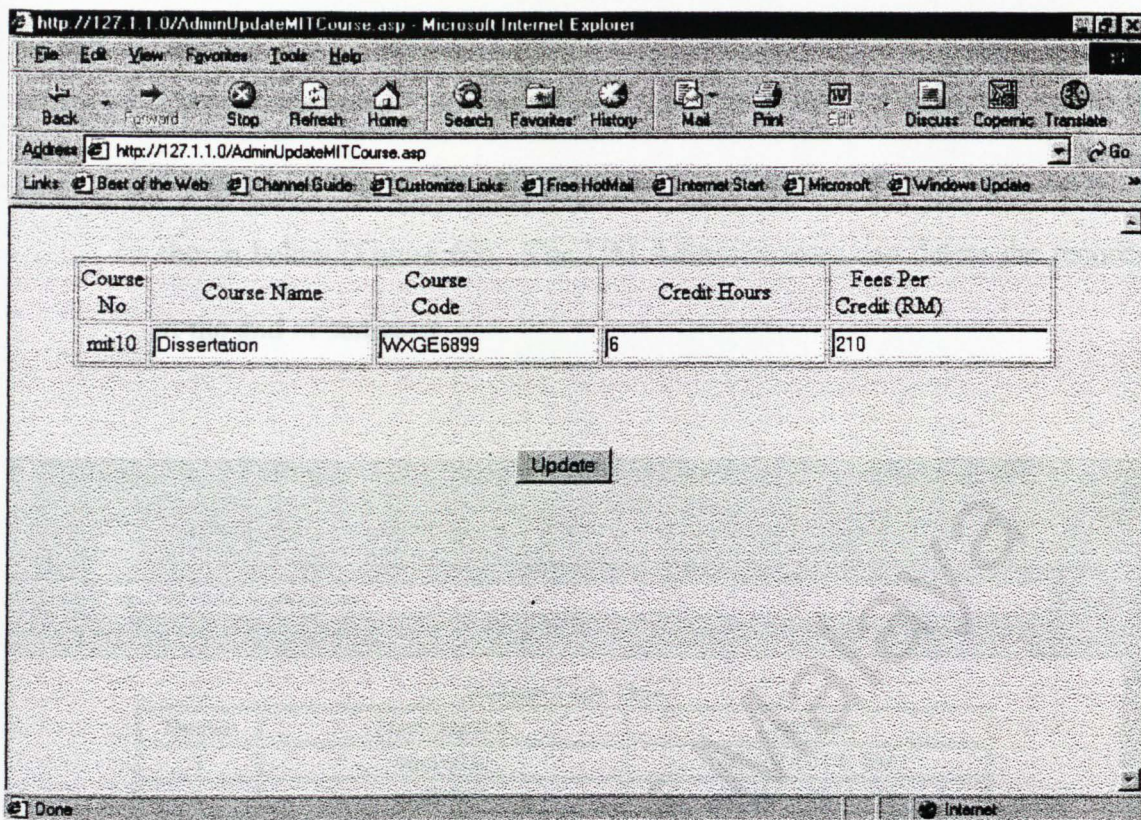
Credit Hours :

Fees Per Credit :

**Figure 63: (Continue from Figure 62)**

- iii. To edit course record, enter the course number in the text box and click the "Edit" button. User will see Figure 64.





**Figure 64: Edit Course Information**

- iv. Click the “*Update*” button after editing. The record will be updated.
- v. To delete the course, enter the course number and click the “*Delete*” button as shown in Figure 63. The course will be deleted.
- vi. To add a new course, complete the 5 text boxes and click the “*Insert*” button as shown in Figure 63. The new course will insert in the record.



5.3.4.4.5 Students Access List

- i. Click on the word “ *Student Access List* ” as shown in Figure 55. Students Access List allows the administrators to check students’ login time as shown below (Figure 65). The UM Administrators can delete the records at the end of semester.

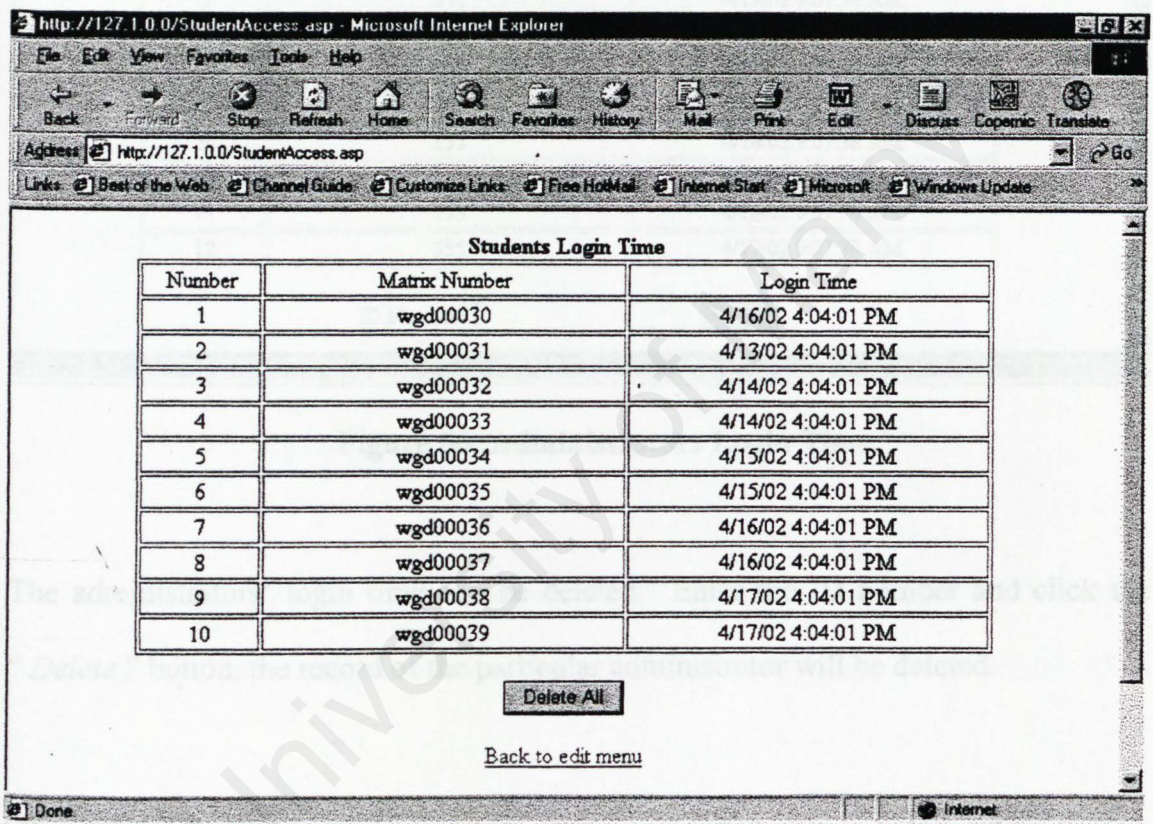
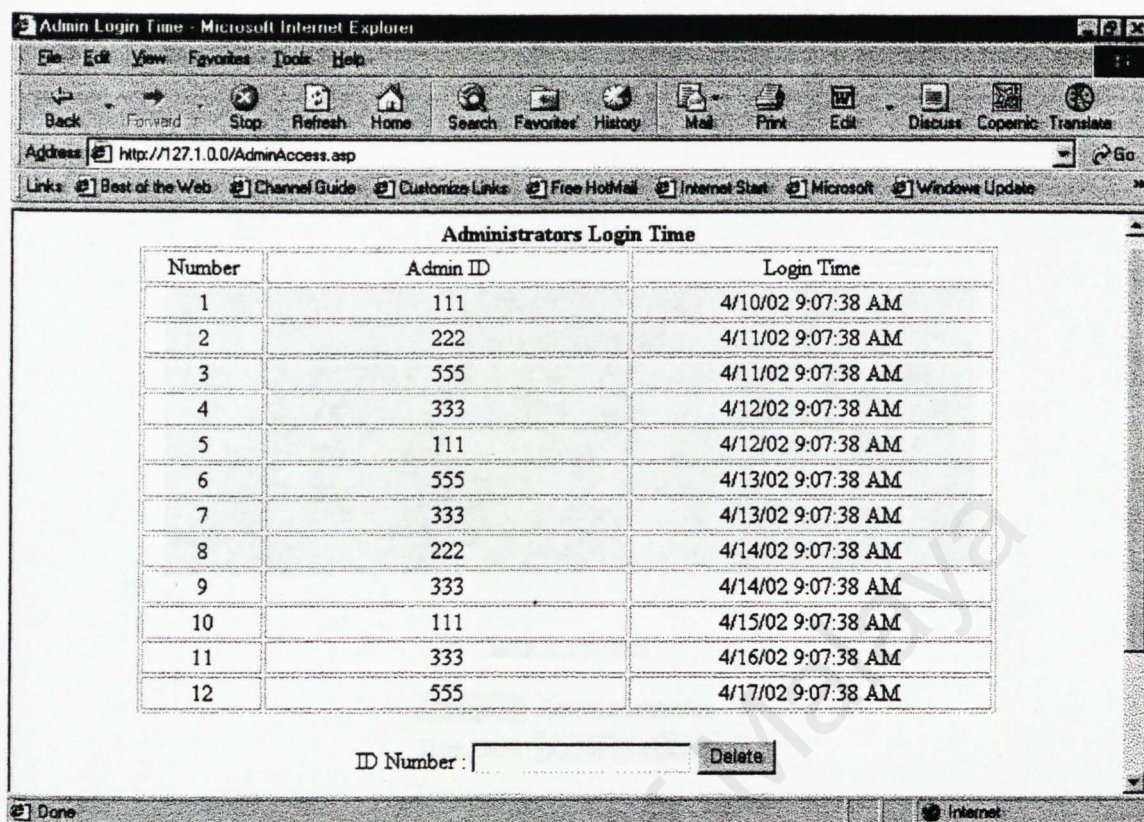


Figure 65: Students Login Time

5.3.4.4.6 Administrators Access List

- i. Click on the “ *Admin Access List* ” as shown in Figure 55. Administrator Access List will allow administrators to check others administrators’ login time as shown in Figure 66.



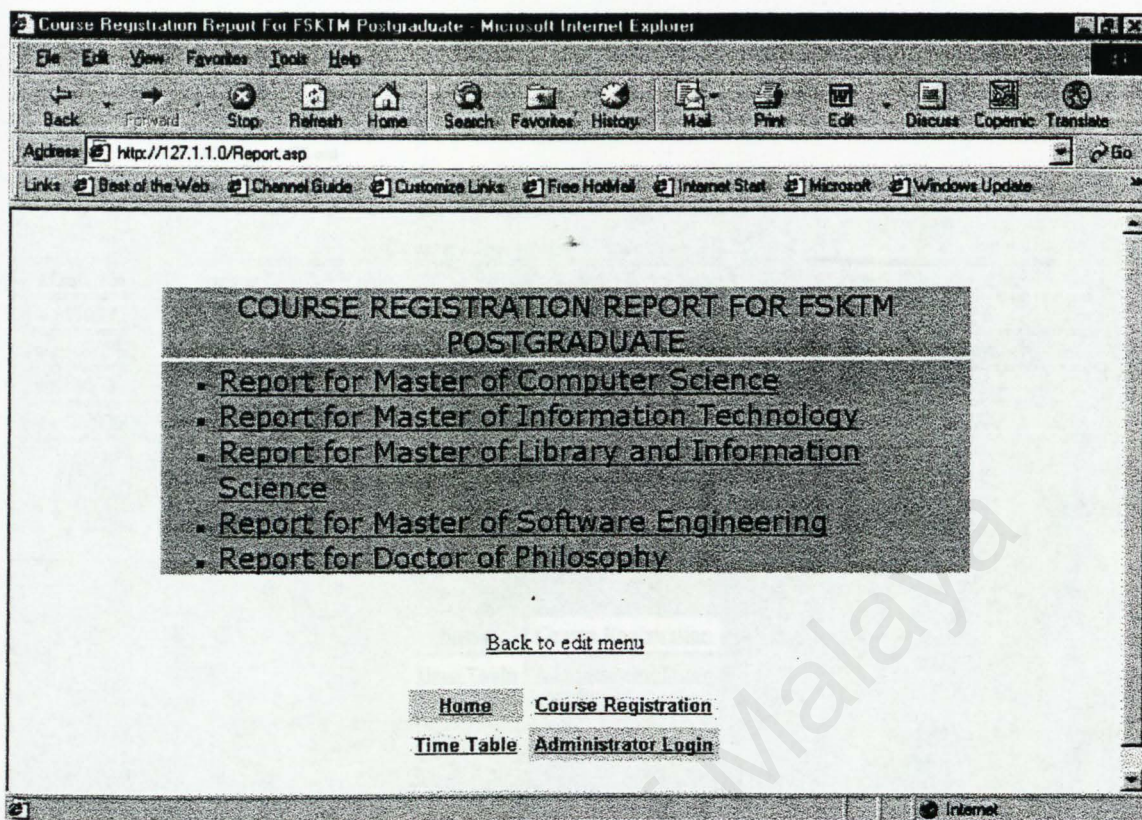
**Figure 66: Administrators Login Time**

The administrators' login time can be deleted. Enter the ID number and click the "Delete" button, the record of the particular administrator will be deleted.

#### 5.3.4.4.7 Course Registration Report

The course registration report main page is shown below (Figure 67).





**Figure 67: Course Registration Report for FSKTM**

The page design for the five different programmes is the same. Therefore, only Report for MCS is chosen for discussion. Figure 33 is the page design for the MCS Report.



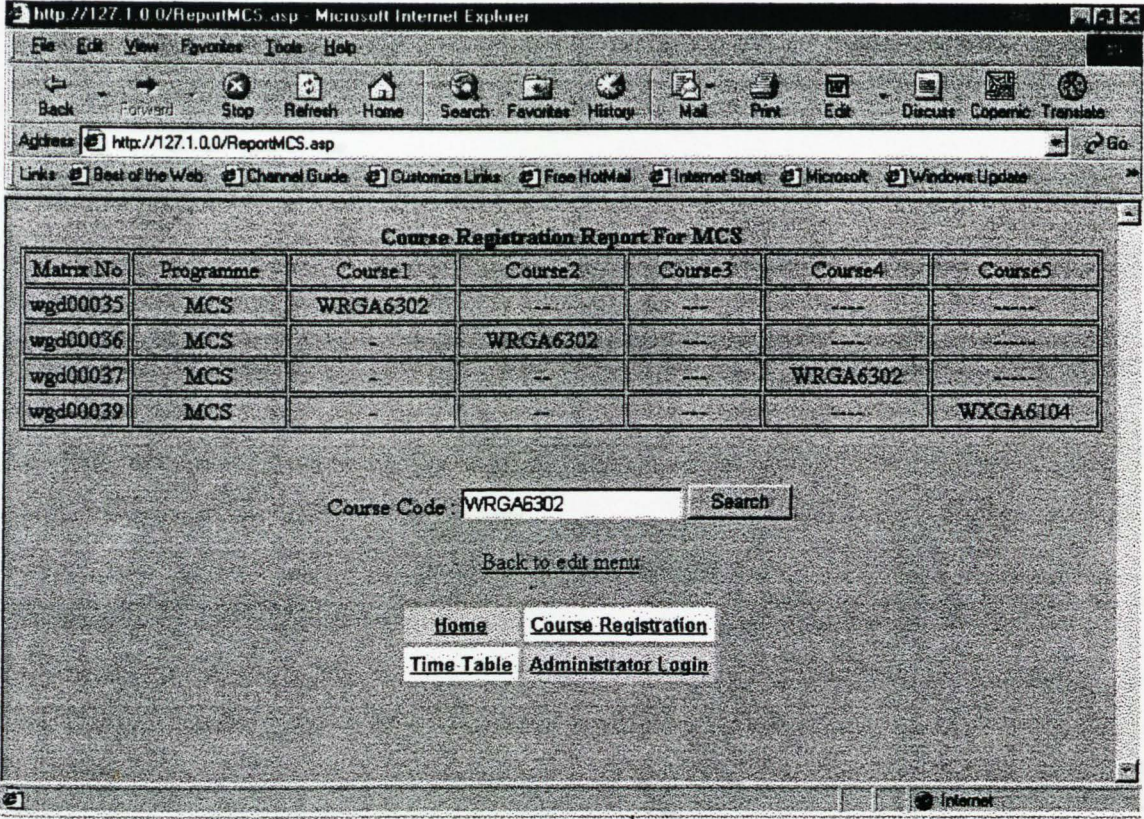
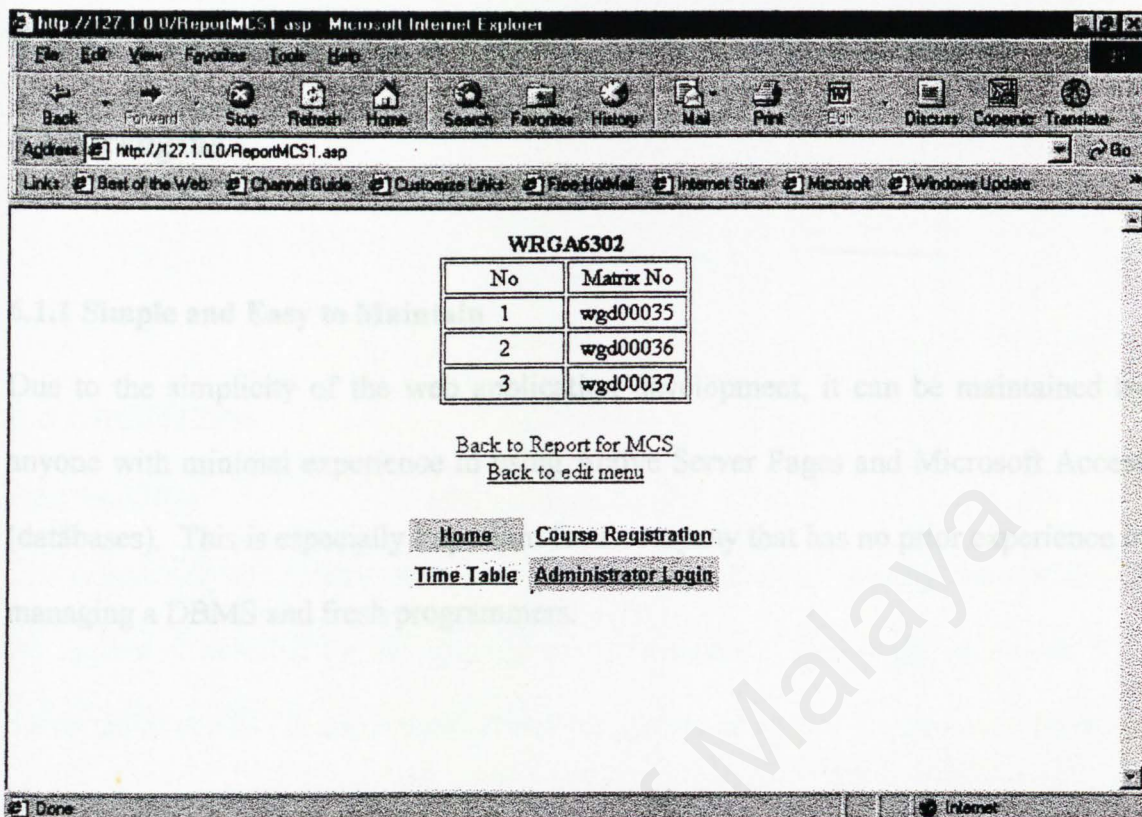


Figure 68: Report for MCS

In Figure 68, there is a table showing all the courses that are registered by the MCS students. User can also check which student and the number of students that registers for the particular course. For example, if they want to check the number of students who registered for course WRGA6302, just type in the course code in the text box and click the “*Search*” button. Then, Figure 69 will appear.





**Figure 69: Students that Register Course WRGA6302**

All the students who registered for course WRGA6302 will appear in the table.

## **6.0 SYSTEM EVALUATIONS AND CONCLUSIONS**

### **6.1 Strengths**

#### **6.1.1 Simple and Easy to Maintain**

Due to the simplicity of the web application development, it can be maintained by anyone with minimal experience in using Active Server Pages and Microsoft Access (databases). This is especially important for a company that has no prior experience in managing a DBMS and fresh programmers.

#### **6.1.2 Low Cost of Ownership**

The system is easy to deploy, manage and upgrade as technology evolves. This is achieved through application level programming interface that supports various development needs. Microsoft Access 2000 was choosing to be the database of this project because the cost is reasonable and easy to use. It is suitable for developing the small and medium system.

#### **6.1.3 High Level of Upgrade**

The system can be easily upgraded. The database initially created using Microsoft Access 2000 can be migrated to the Microsoft SQL Server 7.0 easily using the Access Upsizing Wizard.



#### **6.1.4 Navigation of the System**

Users can navigate the system easily. This is because the navigation buttons and icons in the system are user friendly and easy to understand.

### **6.2 Limitations**

#### **6.2.1 Stability**

The system maybe unstable once we upload it onto the web. Usually, the stability of the system is affected by the stability of the database. Therefore, Microsoft SQL Server 2000 or DB2 is encouraged to use for system database. It is because there are more stable.

#### **6.2.2 Low Recoverability of Stored Data**

Due to the integrated nature of the Microsoft databases files (.mdb), if the database file is corrupted, it is usually difficult to recover the stored data.

#### **6.2.3 Manual Backup of Database**

There is no automatic backup mechanism for the .mdb file and backup would have to be done manually.

#### **6.2.4 Storage Capability**

Microsoft Access has a limit on the amount of data that can be stored in the database. Once the records exceeded a certain quantity, new records cannot be store anymore. The capability will be low when the workloads increase.

#### **6.2.5 Interface Design**

Interface designs in this system are not attractive and static. But, the major purpose of application development of this system is to make the system more users friendly. Therefore, the animation factor has been neglected.

#### **6.2.6 Security**

Even though the system only can be accessed using User ID and Password, but the security of the data transactions and steps to prevent system hacking have been neglected. This is because the lack of securities software and equipments and the duration of developing this system is short.

### **6.3 Future Enhancements to Upgrade the Quality of the System**

The applications of the system are prototype that can be upgraded to the new functions and securities features as below:



### 6.3.1 Security Features

Online security is a major concern for those who are contemplating online payment. This is likely due to the high profile website hackings and online security breaches that occasionally make the news headlines.

If an organization really wants to use this system, I suggest that they have to deal with the service provider regarding the security features in the web site. Most service providers will try to allay our fears by explaining the measures they take to protect transactions.

All Internet service providers use Secure Socket Layer (SSL) technology, which encrypts (scrambles) data so only certain people can read it. Some providers also use Secure Electronic Transaction (SET) technology as an added security measure.

A secure connection is very important whenever for any transaction online. Depending on what the browser that is use, either a closed padlock icon (Internet Explorer) or a key icon (Netscape Navigator) at the bottom of the web page.

This indicates that the transaction is being encrypted, and the users should not worry about someone else eavesdropping on his account. Besides that, the Internet service

provider should be using high-end encryption. For example, 128-bit encryption. To check this out, refer to the text next to the padlock or key icon. It should be “RC4-128”. (Kalakota, Ravi & Whinston A, 1996)

Nevertheless, evaluating the security of the website is beyond the scope of this project.

### 6.3.2 Designing and Deploying Web-Enabled Applications: A Changing Architecture for Changing Needs

With many forces influencing how the Internet behaves and operates, utilizing global network in the same way as it is used in private network presents many challenges. To transform the global network into a reliable application platform, the original architecture of the web must be enhanced to meet the needs that were taken for granted when developing traditional applications. The enhanced architecture to develop and deploy the Web-enabled applications can be seen in Figure 70. (Poo Kuan Hoong, 2000)

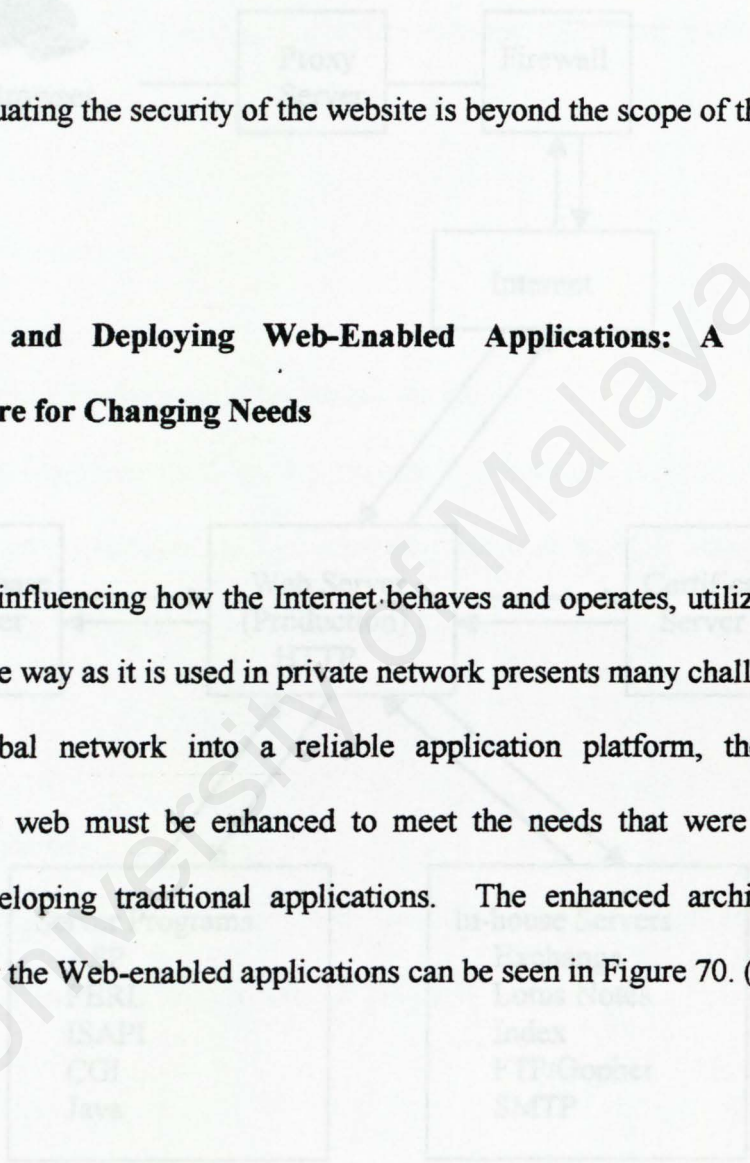
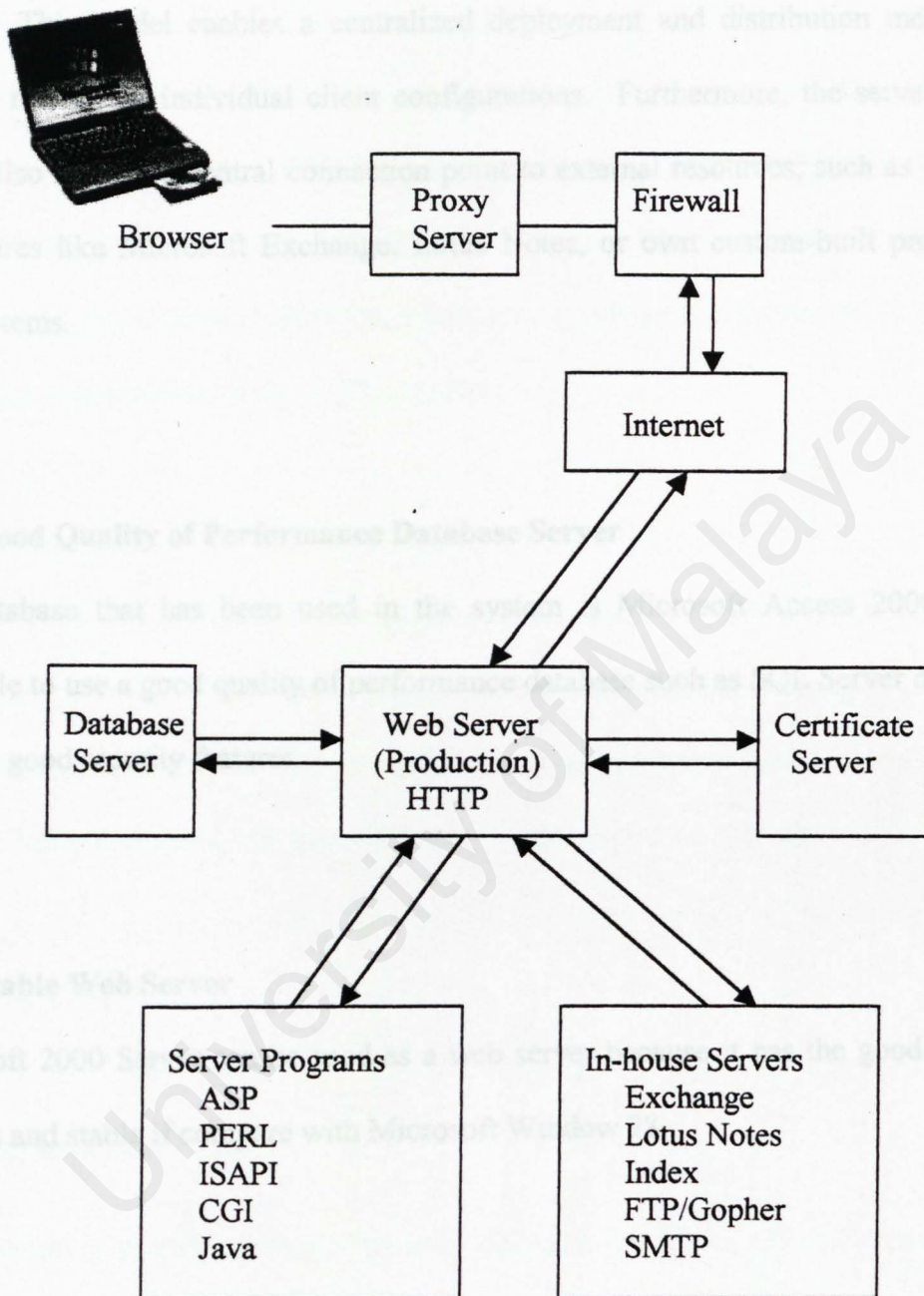


Figure 70: The evolution of Web architecture to support the basic needs of a Web-enabled application





**Figure 70: The evolution of Web architecture to support the basic needs of a Web-enabled application**

The Web-based architecture represents a shift back to the server-centric deployment model. This model enables a centralized deployment and distribution mechanism without relying on individual client configurations. Furthermore, the server-centric model also enables a central connection point to external resources, such as in-house data stores like Microsoft Exchange, Lotus Notes, or own custom-built proprietary data systems.

### **6.3.3 Good Quality of Performance Database Server**

The database that has been used in the system is Microsoft Access 2000. It is advisable to use a good quality of performance database such as SQL Server or Oracle that has good security features.

### **6.3.4 Stable Web Server**

Microsoft 2000 Server can be used as a web server because it has the good security features and stable if compare with Microsoft Window 98.

### **6.3.5 Implement Firewall**

Implementation of firewall can avoid the system hacking and viruses attack to the system. The leak of information after system hacking will affect the reliability of the system from the users. The lost of data after virus attack is a cost to the UM administrators as well as the students.



### 6.3.6 Cisco

Cisco is good because it contents switching solutions ensuring secure and fast online course registration. Cisco 11000 Series Content Services Switch (CSS) and the Cisco 11000 Series Secure Content Accelerator (SCA) can enhance the performance of online course registration application, guaranteeing 100 percent server availability and ensuring secure and successful online registration.

Cisco provides a hardware/firmware solution that is able to handle a higher volume of users. The Cisco CSS 11000 Series also provides automatic fail over between servers for increased availability as well as denial-of-service protection for increased security. The flexibility and simplicity in configuring the hardware are the advantages of using Cisco.

## 6.4 Conclusion

E-commerce is a fast moving area internationally in terms of opportunities and technologies. It is predominantly leads by industry and user sector with Government playing a catalytic role providing pro-active and facilitating support and critical inputs for growth.

In addition, the Internet is radically changing the way consumers shop for goods and services. Credit providers, and especially retailers, are more than willing to satisfy their appetite to buy whatever they need, whenever they need it, without leaving the comfort of office or home. Unfortunately, the Internet is also opening up fresh vistas for a new generation of technologically savvy criminals to steal with greater anonymity, and sending creditors and merchants scrambling to find new products and technologies to protect themselves — and their customers — from fraud. Therefore, to help combat crime in cyberspace, the credit fraud prevention industries have to provide a number of new products and marketing those tools to credit and retail businesses.

Finally, I wish the objectives of this project can be achieved and the system will greatly improve the operational performance in course registration and fees payment by postgraduate students and operational performance of the related tasks by UM administrators. The use of this system could be also expanded to whole UM students.



# GLOSSARY

## Data Flow Diagram (DFD)

A type of graphical representation of flows and data processes in a system. In their original state DFD depicts the broadest possible overview of the system's inputs, processes, and output, in which correspond to data movement in the system.

## Entity-Relationship Model

A detailed logical representation of entities, attributes, and relationship for a company.

## Open Database Connectivity (ODBC)

A standard protocol for database servers. If a database has an ODBC driver which is used to connect to ODBC-compliant data sources.

## ActiveX Data Objects (ADO)

A set of ActiveX components designed to access ODBC data.

## OLEDB

A specification that defines a set of standard interfaces for accessing data. The ActiveX

Data Objects (ADO) implement the OLEDB standard.

## **Data Definition Language (DDL)**

Used by the DBMS to physically establish record types, fields, and structural relationships. Additionally, the DDL defines views of the database. Views restrict the portion of a database that may be used or accessed by different users and programs. DDLs record the definitions in a permanent data repository.

## **Data Manipulation Language (DML)**

Used to create, read, update and delete records in the database and to navigate between different records and types of records—for example, from a customer record to the order records for that customer. The DBMS and DML hide the details concerning how records are organized and allocate to the disk.



## REFERENCES

1. Andrew Aird. (2000). E-commerce in Higher Education: can we afford to do nothing?. Ariadne Issue 26
2. Arvind Panagariya. (1999). E-commerce, WTO & Developing Countries.
3. Cisco System. (2002). Cisco Content Switching Solution Ensures Secure and Fast Online Course Registration for Duke University Students. Cisco System, Inc.
4. Common Gateway Interface. (2002)  
<http://internetconnection.net/support/techcgi.shtml>
5. Connolly, T. & Begg, C. (2000). Database Solution: A step-by-step guide to building databases. Harlow: Addison-Wesley.
6. Dr.P.Sellapan. (2000). Software Engineering: Management & Methods (1<sup>st</sup> Edn). PJ: Sejana Publishing.
7. Dynamic HTML in Netscape Communicator  
<http://developer.netscape.com/library/documentation/communicator/dynhtml/index.html>
8. E-Commerce Perspectives from different parts of the World, IT Pro. Nov/Dec 1999. IEEE Publication.
9. Eric A.Smith, (2000). Active Server Pages Bible. IDG Books Worldwide, Inc.
10. Hari Krishnan A/L Sanmugam Pilai. (1999). Sistem Pengurusan Sijil Ujian Produk Komputer (TC Manager). Fakulti Teknologi & Sains Maklumat, Universiti Kebangsaan Malaysia.
11. Hypertext Transfer Protocol  
[http://searchwebservices.techtarget.com/sDefinition/0,,sid26\\_gci214004,00.html](http://searchwebservices.techtarget.com/sDefinition/0,,sid26_gci214004,00.html)
12. Jeff Kozoris. (1997) Cascading Style Sheets and Dynamic HTML. Network Notes #51 ISSN 1201-4338. Information Analysis and Standards National Library of Canada

13. Jennings, R. (1997). Special edition using Access 97 (2<sup>nd</sup> Edn). Indianapolis, IN: Que Corporation.
14. Johnson, S. & Ballinger, K. & Chapman, D. (1997) Using Active Server Pages: Special Edition. Century Old Style and ITC Franklin Gothic: Que Corporation
15. Kalakota, Ravi & Whinston A. (1996). Frontier Of Electronic Commerce. Addison Wesley Long Man Inc.
16. Kathy Brown. (2000). Ask-an-Expert: e-Commerce Payment Gateways. Biz-Zone Internet Group, Inc
17. Kendall, K.E. & Kendall, J.E. (1999). System analysis and design. Upper Saddle River, NJ: Prentice-Hall.
18. Lawrence A. Rowe. University of California, Berkeley. URL: <http://www.BMRC.Berkeley.EDU/~larry>
19. Levitt, J.(April 7, 2000) "Rift Over HTML's Dynamics," Information Week, , Issue 625
20. Maznee Abdul Halim. (1999). Sistem Pendaftaran Program P & O Melalui Internet --- Kajian Kes Di NPC. Fakulti Teknologi & Sains Maklumat, Universiti Kebangsaan Malaysia.
21. Microsoft Measures Itself Against Netscape  
<http://www.microsoft.com/workshop/author/dhtml/>
22. Nadav Savio (1997). What is Dynamic HTML?  
[http://hotwired.lycos.com/webmonkey/dynamic\\_html/](http://hotwired.lycos.com/webmonkey/dynamic_html/)
23. Netscape Communicator Feature Demos  
[http://home.netscape.com/comprod/products/communicator/beta\\_features.html](http://home.netscape.com/comprod/products/communicator/beta_features.html)
24. Overview of Microsoft's Dynamic HTML  
<http://www.microsoft.com/workshop/prog/aplatfrm/dynhtml.html>
25. Poo Kuan Hoong. (2000). Sistem Laman Web E-Hospitaliti.Net. Fakulti Teknologi & Sains Maklumat, Universiti Kebangsaan Malaysia.
26. Sommerville, I. (2002). Software Engineering (5<sup>th</sup> Edn). Reading, MA: Addison-Wesley.



27. Steve Patient. (2002). Reducing Online Credit Card Fraud. Jupitermedia Group, Inc.
28. Teo Siew Hway. (2000). Pembangunan Sistem Maklumat Pelajar Untuk Program Diploma Secara Kawal Selia Mutu Bagi Fakulti Teknologi & Sains Maklumat. Fakulti Teknologi & Sains Maklumat, Universiti Kebangsaan Malaysia.
29. W3C Cascading Style Sheets, Level 1 Syntax  
<http://www.w3c.org/pub/WWW/TR/REC-CSS1>
30. W3C Positioning HTML Elements Syntax for Layers  
<http://www.w3c.org/pub/WWW/TR/WD-positioning>
31. Zelnick, N. "Client: Dynamic HTML Battle Brewing," Web Week, April 21, 1997, vol. 3, issue 11, p.25.