

FACULTY OF COMPUTER SCIENCE  
AND  
INFORMATION TECHNOLOGY  
UNIVERSITY OF MALAYA  
KUALA LUMPUR

Perpustakaan SKTM

## STUDENT ATTENDANCE SYSTEM

MOHD SAIFULNIZAM BIN ABU BAKAR  
WEK 990338

SESSION 2003/2004

This project is submitted to the Faculty of Computer Science and Information Technology, University Of Malaya in partial fulfillment of the requirements for the Bachelor Of Computer Science (Hons).



**SISTEM KEHADIRAN PELAJAR**  
**STUDENT ATTENDANCE SYSTEM**



...needs up to date and accurate and to look at to upgrade their paper based record  
...system to a digital one. Realizing the potential and the need of a new system, I  
...developed a digital Student Attendance System for lecturer use.

The system is a combination of web based and client server architecture  
...system. Core of the system are ASP in server and client side. This  
...system is especially developed to meet the need in Student Attendance  
...management (SAS) with a main focus on the security.

The system is developed using Active Server Pages, Microsoft Access 2000,  
...and Microsoft Internet Explorer MX.

# Abstract & Acknowledgement

## **ABSTRACT**

Most of the attendance records are still using the old manual filling system or a semi-computerized system. This method faced the challenge of keeping its student records up-to-date and accurate and so looked to upgrade their paper-based record systems to an automated one. Realizing the potential and the need of a new system, I have decided to develop this Student Attendance System for lecturer use.

The system is a combination of web based and client server architecture application system. Users of this system are able to access this system online. This system is especially developed to meet the need in efficiency of attendance management in FCSIT with a main focus on the lecturers of the faculty.

The system is developed using Active Server Pages, Microsoft Access 2000, as well as VBScript and Macromedia Dreamweaver MX.

## ACKNOWLEDGEMENT

First of all, I would like to convey my heartfelt gratitude to my supervisor, Mrs Fazidah Othman. Her guidance and encouragement throughout the development of this project is greatly appreciated.

To Ms Rafidah Md Noor, the moderator of this project, your brilliant ideas and suggestions meant a lot to this project. Your valuable guidance helped me a great deal during the development period.

My deep appreciation also goes to my beloved family who has given me a lot of support throughout the project. Also thanks you to all of my friends who has been with me and giving valuable ideas and support all along the completion of this project.

There are other who I missed, please accept my humblest apologies and my assurances that your help is greatly appreciated.

THANK YOU!

## TABLE OF CONTENTS

<b>ABSTRACT</b>	ii
<b>ACKNOWLEDGEMENT</b>	iii
<b>TABLE OF CONTENT</b>	iv
<b>LIST OF FIGURES</b>	viii
<b>LIST OF TABLES</b>	ix

### CHAPTER 1 INTRODUCTION

<b>1.1 PROJECT OVERVIEW</b>	1
<b>1.2 OBJECTIVES</b>	3
<b>1.3 SCOPE</b>	4
<b>1.4 PROJECT EXPECTATION</b>	5
<b>1.5 PROJECT SCHEDULE</b>	6
<b>1.6 PROJECT OUTLINE</b>	7

### CHAPTER 2 LITERATURE REVIEW

<b>2.1 PURPOSE OF LITERATURE REVIEW</b>	9
<b>2.2 TECHNIQUES ON CONDUCTING LITERATURE REVIEW</b>	9
<b>2.3 EXISTING MANUAL SYSTEM IN FCSIT</b>	11
<b>2.4 CURRENT SYSTEM</b>	12
2.4.1 MyGradebook.com	12
2.4.2 Student Tracking System (DUBAI WOMEN'S COLLEGE)	15
2.4.3 Faculty Senate Attendance System	18
<b>2.5 TECHNOLOGY CONSIDERATION</b>	21

2.5.1	Client-Server Computing Technology	21
2.5.2	Types Of Client/Server Computing	25
2.5.3	Web Server	34
2.5.4	Web Application Development	36
2.5.5	Programming / Scripting Languages	39
2.5.6	Database Management System	41
2.5.7	Data Access Models	43
2.6	SUMMARY	46

### **CHAPTER 3 METHODOLOGY**

3.1	INTRODUCTION	47
3.2	SOFTWARE DEVELOPMENT MODEL	47
3.2.1	WATERFALL MODEL	48
3.2.2	THE PROTOTYPING MODEL	50
3.2.3	THE WATERFALL MODEL WITH PROTOTYPING	54

### **CHAPTER 4 SYSTEM ANALYSIS**

4.1	DEVELOPMENT CONCEPT	
4.1.1	System Architecture Concept	55
4.1.2	Operating System	55
4.1.3	Web Programming Language	56
4.1.4	Database Management System	58
4.1.5	Scripting Language	59
4.1.6	ASP Editor	60

4.2	SYSTEM REQUIREMENT	61
4.2.1	Functional Requirements	62
4.2.2	Non-Functional Requirements	64
4.3	SOFTWARE REQUIREMENT	66
4.4	HARDWARE REQUIREMENT	67
<b>CHAPTER 5 SYSTEM DESIGN</b>		
5.1	ARCHITECTURE DESIGN	69
5.2	LAYOUT DESIGN	71
5.3	DATABASE DESIGN	73
5.4	SYSTEM FUNCTIONAL DESIGN	75
<b>CHAPTER 6 SYSTEM IMPLEMENTATION</b>		
6.1	SYSTEM IMPLEMENTATION	81
6.2	SYSTEM CODING	81
6.2.1	Coding Style	81
6.2.2	Code Documentation	82
6.3	CODING EXAMPLE	84
6.4	CONCLUSION	85
<b>CHAPTER 7 SYSTEM TESTING</b>		
7.1	INTRODUCTION	86
7.2	TESTING	86
7.3	UNIT TESTING	
7.3.1	Unit Test Cases	87

7.3.2	Unit Test Results	88
7.4	INTEGRATION TESTING	
7.4.1	Bottom Up Approach	89
7.4.2	Integration Test Cases	89
7.5	SYSTEM TESTING	91
7.6	CONCLUSION	93

## **CHAPTER 8 SYSTEM EVALUATION & CONCLUSION**

8.1	INTRODUCTION	94
8.2	PROBLEMS ENCOUNTERED AND SOLUTIONS	94
8.3	SYSTEM STRENGTHS	96
8.4	SYSTEM LIMITATIONS	97
8.5	EVALUATION BY END USER	96
8.6	FUTURE ENHANCEMENT	98
8.7	KNOWLEDGE AND EXPERIENCE GAINED	99
8.8	CONCLUSION	100

## **REFERENCES**

## **APPENDIX**

### **A) USER MANUAL**



## LIST OF FIGURES

Figure 1-1 Project Schedule For Student Attendance System	6
Figure 2-1 MyGradebook.com Reports	13
Figure 2-3 Main screen Of Student Tracking System	16
Figure 2-4 Menu Screen Of Student Tracking System	16
Figure 2-5 Check the attendance for the student in the same course and other courses	17
Figure 2.6 Attendance Amendment	18
Figure 2-7 Inquire Student Attendance	18
Figure 2-8 Faculty Senate Attendance System	19
Figure 2.9 Faculty Senate Attendance System (Member List)	20
Figure 2-10 : Two Tier Client Server Architecture Design	26
Figure 2-11: Three tier distributed client/server architecture depiction	29
Figure 2-12: N tier distributed client/server architecture depiction	32
Figure 3-1 The Waterfall Model	48
Figure 3-2 The Prototyping Model	52
Figure 3-3 The Waterfall Model with Prototyping	54
Figure 4-1 Screenshot of Microsoft Visual Interdev 6.0	60
Figure 5-1 System tier of Student Attendance System	70
Figure 5.2 Main Menu For Student Attendance System	72
Figure 5-3 Context Diagram For Student Attendance System	76
Figure 5-4 Zero-Level Diagram For Student Attendance System	77
Figure 5-5 Administrator Module's Flow Chart	79
Figure 5-6 Lecturer Module's Flow Chart	80
Figure 7-1 Levels of Testing	86

## LIST OF TABLES

Table 2-1 N-Tier	33
Table 4-1 Functional requirement	63
Table 5-1 Symbols using Gane and Sarson	76
Table 5-2 Flow Chart Component	78

University of Malaya

CHAPTER 1

Introduction



## SISTEM KEHADIRAN PELAJAR STUDENT ATTENDANCE SYSTEM



more on their own activities. Lecturers require computers, and are usually motivated by a desire to spend more time in the lecture hall and less time doing the administrative side of their work.

The administrative situation has become more complicated in recent years since universities have a greater number of students. So, a significant and increasing percentage of a lecturer's time is now spent ensuring that accurate records are kept of student attendance, academic and behavioral performance.

The information must be recorded, organized and stored efficiently among other positions of staff, if it is to be useful. These associated administrative demands make the delivery of quality education difficult and place lecturers under ever increasing stress.

Many universities already have a student administration system to store and analyze student information. Unfortunately, these are often systems which are cumbersome and difficult to use. In faculty of a thousand students or more, this task is extremely very time-consuming for lecturers, as

lecturer's first concern should be that it be tailored accordingly and permit an incorporation needed by their work. As a

# CHAPTER 1 Introduction

### 1.1 Project Overview

Many professionals want to be able to spend less time on administration and more on their core activities. Lecturers are no exception, and are usually motivated by a desire to spend more time in the lecture hall and less time filling out forms or performing administration duties.

The administration situation has become more complicated in recent years since universities have a greater number of students. So, a significant and increasing proportion of a lecturer's time is now spent ensuring that accurate records are kept of student attendance, academic and behavioral performance.

The information must be recorded, managed and shared efficiently among other members of staff, if it is to be useful. These associated administrative demands make the delivery of quality education more difficult and place lecturers under ever increasing stress.

Many universities already have a central administration system to store and analyze student information. Universities that use these systems often use attendance sheet and enter the data manually. In faculty of a thousand students or more, this task is obviously very time-consuming for lecturers, and since this is not the lecturer's first concern the data is likely to be entered infrequently and prone to inaccuracies caused by keying errors, for example

Student attendance system is a system develops for lecturers used for recording, storing, analyze and print out all records regarding student attendance for every semester. This system is developing special purpose for lecturer FSCIT used.

For the time being, the only system used is manual system, which attendances were signed by student in a paper. This document approach is not practical because lecturers need to calculate and analysis all attendance records. It also waste their times and difficult to keep records in bunch of papers.

University of Malaya

This project has several objectives in order of its completion. The main objective of the project is to develop a student attendance system that contains information of the student in the faculty and provide a system that includes student's attendance records and summary. Apart from the main objective, these are the other objectives:

- To develop a computerized and paperless system that will help the administration procedures.
- To increase the efficiency in management operations since the system provide a better decision making tool.
- To reduce data recording and manipulating errors when tasks, which are done manually, will be done automatically by the system itself.
- To design a system which produces full reports with various sorts and formats
- To create a simple, easy to learn and user friendly system. This includes an attractive Graphic User Interface, function, etc.

### 1.3 Scope Expectations

#### Student

The system will store the information of the student in database. Name, matrix number, IC number etc are the information stored.

#### Subject

It contains full list of subject regarding for what the lecturer take during the semester. So lecturers are able to view attendance by subject and get the reports.

#### Reports

Able to generate reports based on lecturers need:

- 1) 80% less attendance per semester
- 2) 3 consecutive absent

#### 1.4 Project Expectation

This project will benefit a lot of people in every aspect. There is several expectation of the outcome even before and after developing the system. Few factors are put in consideration the meet these expectation. Some of the expectations of the project are given below.

- this system will change the existing manual system or semi computerized system into a fully computerized online system
- stability, consistency, user friendly and reliability are the main criteria to be taken when the system performs some basic function.



## 1.5 Project Schedule

Each project will need a project schedule. In order to achieve project objectives, a project schedule has been properly planned to manage the time and tasks that must be completed with a planned development phases. Figure 1-1 below show the project schedule for this development project.

The project will starts at 17<sup>th</sup> Mac 2003 until 14<sup>th</sup> August 2003

ID	Task Name	Start	Finish	Duration	Mar 2003		Apr 2003			May 2003			Jun 2003			Jul 2003			Aug 2003	
					30	31	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	System Study	3/17/2003	3/25/2003	7d	■															
2	Literature Review	3/26/2003	4/8/2003	10d		■	■	■	■	■	■									
3	Requirement Analysis	4/9/2003	4/14/2003	4d				■												
4	System Design	4/15/2003	4/18/2003	4d				■												
5	Database Design	4/18/2003	4/21/2003	2d				■												
6	Layout Design	4/22/2003	4/22/2003	1d					■											
7	Coding	4/24/2003	8/13/2003	80d						■	■	■	■	■	■	■	■	■	■	■
8	Testing	8/14/2003	8/14/2003	1d																■

Figure 1-1 Project Schedule for Student Attendance System

This report is purposely for the documentation of the essential information gathered and implemented throughout the development of the project. It also covers the literature review of the project, analysis and partial of the design of the system. The report can be divided into five main chapters as following:

- Chapter 1: Introduction

This section gives an overview of the overall proposed project. This includes the project aims, objectives, scope, significance, schedule and expectations.

- Chapter 2: Literature Review

This section compiles the studies and researches done, section by section including the analysis and synthesis of the information gathered for the project. This also concludes the survey conducted on the current conventional system.

- Chapter 3: Methodology

This section describes what type of methodology used as an approach to develop the Student Attendance System. Steps and how to manage the development of the system is defined in this section.

- Chapter 4: System Analysis

This section describes how the whole analysis been done since the start of the project. Functional and non-functional requirements of the system based from requirement analysis in system development, also included in this section apart from the methodology used for this project plan.

- Chapter 5: System Design

This section summarizes the consideration for the input design, output design and database implementation into the system development.

- Chapter 6: System Implementation

This section summarizes the way Student Attendance System was being implemented with the translation of program design to program coding.

- Chapter 7: System Testing

This section provides the details on how the system is being tested and the approach used during the phase

- Chapter 8: System Evaluation and Conclusion

This section discusses the strengths and limitations of the system as to evaluate the system and the conclusion of the overall project.

University of Malaya

CHAPTER 2

Literature Review



This chapter will focus on analyzing the existing and similar system in the development process. Each system will be analyzed. This chapter also covers the technologies that can be used for this development project.

## 2.2 Technology on Conducting Literature Review

### 2.2.1 Interview and Questionnaire

One's opinion is important towards the effectiveness in developing the system. Thus, interviews and questionnaires are essential to gather the requirements and information that be expected and required users. Information such as system requirements and problems of existing system can be addressed using this method.

### 2.2.2 Searching on Internet

The internet is useful tool for gathering information for the design and all the resources needed for the research. Sites regarding organizations, government, business, news reviewed thoroughly and analyzed in order to get the development and what system is all about.

## CHAPTER 2

# Literature Review

### 2.1 Purpose of Literature Review

This chapter will focus on analyzing the existing and similar system to the development project. Each system will be analyzed. This chapter also covers the technologies that can be used for this development project.

### 2.2 Technique on Conducting Literature Review

#### 2.2.1 Interview and Questionnaire

User's opinion is important towards the efficiency on developing the system. Thus, interviews and questionnaire are essential to gather the requirements and information from the expected and targeted users. Information such as system requirements and problems of existing system can be addressed using this method.

#### 2.2.2 Surfing the Internet

The internet is a powerful tool for gathering information for the design and all the related information needed for the research. Sites regarding organizations, government bodies etc were reviewed thoroughly and analyzed in order to get the concepts and idea what system is all about.

### **2.2.3 Reading**

Books, magazines, newspapers and journals are excellent resources to get information regarding the system. Most of the sources for information specially books concerning the developing of the system are retrieved from the main library of University of Malaya. Besides, other related issues regarding the system are also retrieved from local bookstores around Klang Valley.

### **2.2.4 FSCIT Document Room**

This room is very important part in my literature reviews because its contains lots of past year's thesis. I can refer any previous project related with my task and make easier to me to get understand clearly. Its help me specially to get a good idea on the documentation side of the project.

### **2.2.5 Discussion**

In discussion, we can exchange information and knowledge within each other.

As information is meant to be shared, it is privileged to conduct discussion among friends about ideas on completing the project.

## **2.3 Existing Manual System In Faculty Science Computer and Information Technology (FCSIT)**

Survey has been carried out to understand how the current management system of attendance. As ISO 9001 has been implement widely especially in administration works, lecturer are encourage to use systematic approach to make sure they keep on track day to day activities such as lecture class, tutorials, test and assignment .According to ISO 9001 rule and regulation, all records and information must keep in document means that all lecturer should have records about all what they done whatever for classes, assignment, tutorial and attendance.

Current system is done manually by lecturers themselves. Mainly there are 4 basic activities managing the students include assignments, tutorial, attendance and exam. So, all lecturers need to record this entire task in proper document.

### **Tutorial, Attendance and Exam**

Students who attend class and tutorial will sign in their attendance sheet. Lecturers will use this attendance sheet for record purpose. They also can find which student not attend class in 3 consecutive times or not fulfill 80% of overall attendance. Attendance sheet need to submit to Office for ISO compliances and make sure that office can produce show cause letter to particular students.

## 2.4 Current Online System

### 2.4.1 My GradeBook.com

URL : <http://www.mygradebook.com>

My Gradebook.com is a comprehensive software program that manages grades and assignment. This system has three different login types

- ✓ Teachers Login
- ✓ Parents and Student Login
- ✓ Group Account Manager

Main features in MyGradeBook includes:

#### 1. Reports

Select from the 20 ready-to-print reports for a quick overview of how the class or an individual student is doing. Instantly ready to share information with school administrators or to discuss progress during parent-teacher conferences. Users can customize the reports to further meet specific needs.

Ready-to-print reports include:

- Grade summaries by class or by student on a specific assignment or for the grading period. Immediately identify missing scores or which students are performing below a specified level.
- Student information summaries, including a record of performance comments you have made about that student.
- Attendance summaries by class or by student for a day or for the grading period.



- An assignment summary indicating due dates and the number of points that may be earned on each assignment

The screenshot displays the MyGradebook.com interface. The top section shows the user is working in the 'Geometry (ID=82197)' class. Navigation tabs include Class, Categories, Assignments, Students, Scores, Attendance, Overall, and Tools. A 'Show:' dropdown is set to 'all students', and the selected student is '1190782 Russos, Diane'. The 'Period' is set to 'All grading periods'. A list of checkboxes allows filtering by 'Overall Summary', 'Category Summary', 'Assignments By Category', 'Assignments By Date Due', 'Show Future Assignments', 'Student Chart', 'Attendance Summary', 'Attendance Details', and 'List All Students'. Below this, the 'Overall Summary' for student ID 1190782, Diane Russos, shows a grade of 'A+ (100.00%)' with 4 assignments and 0 missing. A 'Performance By Category' table is shown, and a 'Grading Period Weights' table lists quarters from 9/01/2001 to 4/16/2002.

The bottom section shows a detailed view for the 'Homework' category. It lists the category weight as 50.00, class start date as 09/01/2001, and class end date as 06/30/2002. It then displays reports for three students:

Student Name	Student ID	User ID	Overall Grade	Citizenship	Effort
Adorney, Carolyn A	1290884	123456	A+ (100.00%)		
<b>Date Due</b>	<b>Assignment Name</b>	<b>Assignment Weight</b>	<b>Points Given</b>	<b>Max Points</b>	<b>% Given</b>
10/22/2001	Chapter 10, Summary 1-25	0.00	100	100.00	100.00%
10/22/2001	Read Chapter 10	0.00	?	10.00	?
Barboza, Chris B	1290665	457867	A (95.00%)		
<b>Date Due</b>	<b>Assignment Name</b>	<b>Assignment Weight</b>	<b>Points Given</b>	<b>Max Points</b>	<b>% Given</b>
10/22/2001	Chapter 10, Summary 1-25	0.00	95	100.00	95.00%
10/22/2001	Read Chapter 10	0.00	?	10.00	?
Bradley, Joe C	1290666	758694	A+ (100.00%)		
<b>Date Due</b>	<b>Assignment Name</b>	<b>Assignment Weight</b>	<b>Points Given</b>	<b>Max Points</b>	<b>% Given</b>
10/22/2001	Chapter 10, Summary 1-25	0.00	100	100.00	100.00%
10/22/2001	Read Chapter 10	0.00	?	10.00	?

Figure 2-1 MyGradebook.com Reports

## 2. Attendance

Track attendance online and take advantage of the option to send email notification to parents of any student who is absent or tardy.

Customize the MyGradeBook attendance system:

- use own codes (A=absent, T=tardy, etc.)
- choose which codes should result in email notification to parents
- show only the days of the week that the class is in session.

mygradebook.com [Help](#) / [Contact Us](#) / [Calendar](#) / [My Account](#) / [Email](#) / [Logout](#)

You are working in:

Class Categories Assignments Students Scores Attendance Reports Tools

Daily Weekly Calendars Meeting Calls

Previous Day **Wednesday, October 17, 2001** Next Day

Student ID	Student Name	Present	Absent	Excused	Holiday	Tardy	Email Sent
<a href="#">1290664</a>	Adorney, Carolyn A.	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	No
<a href="#">1290665</a>	Barboza, Chris B.	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	No
<a href="#">1290666</a>	Bradley, Joe C.	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	No
<a href="#">1290667</a>	Brown, Derek D.	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	No
<a href="#">1290668</a>	Carter, Jane E.	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	No
<a href="#">1290669</a>	Cornell, Alyssa F.	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	No
<a href="#">1290670</a>	Deaner, Amy G.	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	No
<a href="#">1290671</a>	Engdahl, Sheila H.	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	No
<a href="#">1290672</a>	Estevez, Erica I.	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	No
<a href="#">1290673</a>	Faulconer, Jeff J.	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	No
<a href="#">1290674</a>	Kureck, Keith K.	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	No
<a href="#">1290675</a>	Lyons, Carolyn L.	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	No
<a href="#">1190783</a>	Marciano, Alex	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	No
<a href="#">1290676</a>	Marciano, Alex M.	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	No
<a href="#">1290677</a>	O'Brien, Mark N.	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	No
<a href="#">1290678</a>	Radack, Alice O.	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	No
<a href="#">1190782</a>	Russas, Diana	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	No
<a href="#">1290679</a>	Tostado, Ana R.	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	No
<a href="#">1290680</a>	White, Jen S.	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	No
<a href="#">1290681</a>	Yablonski, Matt T.	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	No

### 3. Security

Protecting the privacy of personal and student information and providing reliable access to data is the core of security efforts at MyGradeBook.

Information privacy precautions include:

- Access encrypted via the Secure Socket Layer (SSL) protocol.
- Secure logout for protection of accounts accessed on a shared computer.
- If three consecutive unsuccessful login attempts are made, the account is automatically locked for an hour. A teacher can only unlock the account prior to the end of the hour by entering the "Unlock Password".
- Every time an attempt to access a teacher account is made, we email the teacher showing the date, time, IP address, and the result of the attempt.

Data security:

- Daily server backups and offsite storage of archives.
- Even if a computer is stolen or the hard drive crashes, all data remains safe and private at MyGradeBook.

## 2.4.2 Student Tracking System (DUBAI WOMEN'S COLLEGE)

URL : <http://www.oz.dwc.hct.ac.ae>

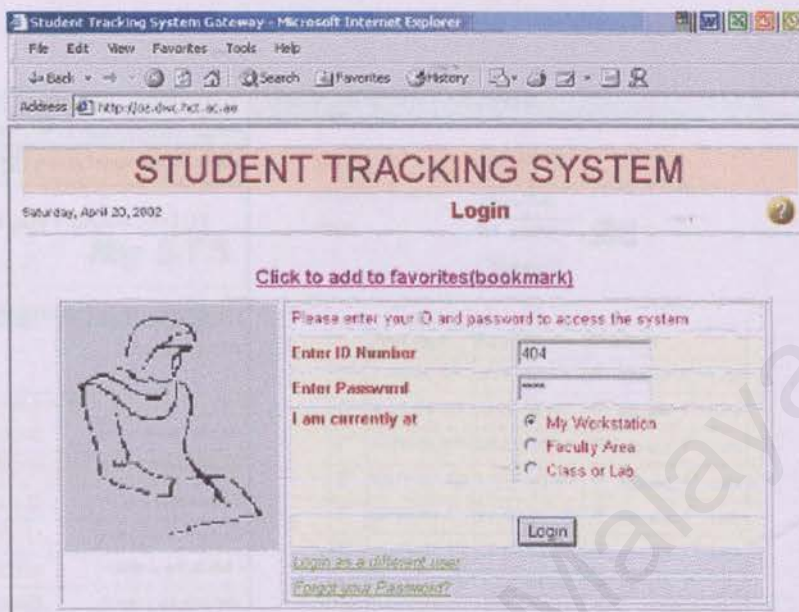


Figure 2-3 Main screen Of Student Tracking System

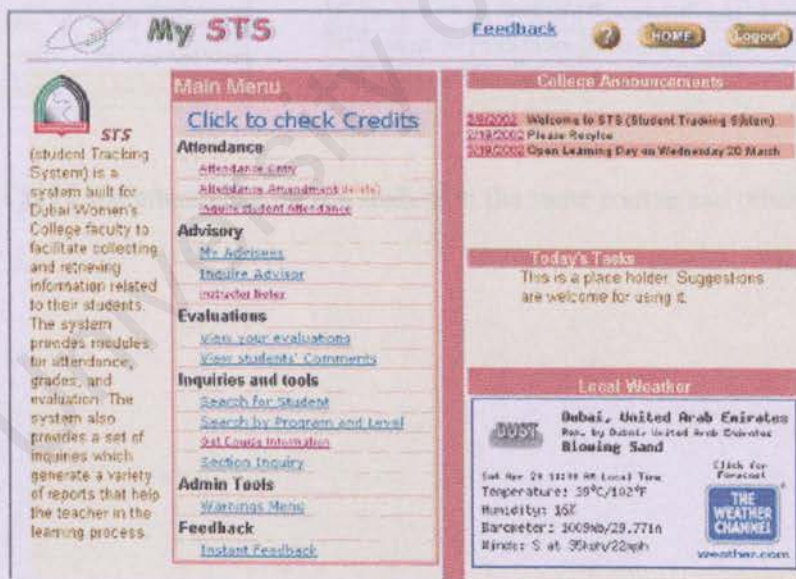


Figure 2-4 Menu Screen Of Student Tracking System

## System Feature/Menu

### 1. Attendance Entry

- to input the student attendance into the system.

Address: http://FACULTY/updates.asp?CRNK

Hello, Valery Plummer >>>1111

My STS

Reload Page My Courses

Attendance En

Student ID	Name
2000221203	Amal Bilal Sabt Ma
2000222508	Asma Abbas Mohamed Hassan Alstoooshi
2000221407	Ayada Mohamed A
2000221119	Fatema Ahmed Ab
2000221404	Fatema Ali Abdulla
2000221426	Fatma Ali Khamis
2000221510	Fatma Mohamed .
2000221423	Fatma Ali Mohamed
2000221532	Fatma Othman Mo
2000211956	Haifa Salahuddin F
2000221601	Hind Humaid Saad
2000221209	Khasia Nubarak B

Student ID: 200022558

Student Name: Asma Abbas Mohamed Hassan Alstoooshi

Date: 21-4-2002

Submit

	Start time	End Time	Abs/Late
<input type="checkbox"/>	7:45:00 AM	8:30:00 AM	<input checked="" type="radio"/> Absent <input type="radio"/> Late
<input type="checkbox"/>	8:35:00 AM	9:20:00 AM	<input checked="" type="radio"/> Absent <input type="radio"/> Late
<input type="checkbox"/>	9:25:00 AM	10:10:00 AM	<input checked="" type="radio"/> Absent <input type="radio"/> Late
<input type="checkbox"/>	10:15:00 AM	11:00:00 AM	<input checked="" type="radio"/> Absent <input type="radio"/> Late
<input type="checkbox"/>	11:05:00 AM	11:50:00 AM	<input checked="" type="radio"/> Absent <input type="radio"/> Late
<input type="checkbox"/>	11:55:00 AM	12:40:00 PM	<input checked="" type="radio"/> Absent <input type="radio"/> Late
<input type="checkbox"/>	12:45:00 PM	1:30:00 PM	<input checked="" type="radio"/> Absent <input type="radio"/> Late
<input type="checkbox"/>	1:35:00 PM	2:20:00 PM	<input checked="" type="radio"/> Absent <input type="radio"/> Late
<input type="checkbox"/>	2:25:00 PM	3:10:00 PM	<input checked="" type="radio"/> Absent <input type="radio"/> Late
<input type="checkbox"/>	3:15:00 PM	4:00:00 PM	<input checked="" type="radio"/> Absent <input type="radio"/> Late
<input type="checkbox"/>	4:05:00 PM	4:50:00 PM	<input checked="" type="radio"/> Absent <input type="radio"/> Late
<input type="checkbox"/>	4:55:00 PM	5:40:00 PM	<input checked="" type="radio"/> Absent <input type="radio"/> Late

Figure 2-5 Check the attendance for the student in the same course and other courses

## 2. Attendance Amendment

- to delete a record of the same day

The screenshot shows the 'Attendance Update for Student' page in the My STS system. At the top, there is a navigation bar with 'My STS' logo, a 'Feedback' link, and buttons for 'HOME' and 'Logout'. The main heading is 'Attendance Update for Student'. Below this, a red warning message states: 'You are required to re-actually enter the student id when you wish to change the attendance. Please notice that you can only amend attendance entered today. Any attendance entries that triggered a warning can not be deleted. At the moment, you can delete any entries for a student. If update is necessary, delete the particular record then re-enter it from the attendance entry screen.' The form includes a 'Select Course' dropdown menu currently set to 'OTEC2202 Office Computer Application II (C4B)', an 'Enter Student Number' text input field, and a 'Request Entries' button. A 'Date' field is visible at the bottom left of the page.

Figure 2.6 Attendance Amendment

## 3. Inquire Student Attendance

- provide a report of student attendance over semester in all courses taken

The screenshot shows the 'Inquire Student Attendance' page in the My STS system. The navigation bar at the top includes the 'My STS' logo, a 'Feedback' link, and buttons for 'HOME' and 'Logout'. The main heading is 'Enter student id or select section to display student's attendance'. Below this, there is a search form with three input fields: 'Student ID' (with a note '(must leave ID blank)'), 'or search by name', and 'Student Name' (with a dropdown menu showing 'Choose one of the above entries'). At the bottom of the form are 'Submit' and 'Reset' buttons.

Figure 2-7 Inquire Student Attendance

## 2.4.2 Faculty Senate Attendance System

URL : <http://facsen.library.musc.edu>



Figure 2-8 Faculty Senate Attendance System

Faculty Senate Attendance System is one kind of web based system to view all information regarding attendance for Medical University Of South Carolina. Users are able to view attendance list by individual, by meeting, by year and by archived individual but they not allow editing any records. Only authorized person can log into this system by click on administrative menu. They must key in user ID and password to enter the system.

**Faculty Senate**

**Faculty Senate Member List**

*Click on a Member to View Record*

Name	Electoral Unit	Type	End Date	Attendance
<a href="#">Susan Ackerman</a>	College of Medicine - Clinical Sciences	Senator	2004	4/7
<a href="#">Richard J. Albenasius</a>	College of Dental Medicine	Senator	2004	6/7
<a href="#">Elinor Amella</a>	College of Nursing	Senator	2004	7/7
<a href="#">Philippe Arnaud</a>	College of Medicine - Basic Sciences	Senator	2003	18/20
<a href="#">Julia Ball</a>	College of Nursing	Senator	2004	5/7
<a href="#">Connie Best</a>	College of Medicine - Clinical Sciences	Senator	2003	12/20
<a href="#">John Botsis</a>	College of Pharmacy	Alternate	2002	20/32
<a href="#">DeAnna Cheek</a>	College of Medicine - Clinical Sciences	Senator	2003	10/20
<a href="#">Katharine Cheesman</a>	College of Pharmacy	Senator	2003	17/20
<a href="#">Jack Cumbley</a>	College of Medicine - Clinical Sciences	Senator	2004	0/7
<a href="#">Libby Dismuke</a>	College of Health Professions	Alternate	2004	5/6
<a href="#">Elizabeth Etkal</a>	College of Dental Medicine	Senator	2004	4/7

Figure 2.9 Faculty Senate Attendance System (Member List)



## 2.5 Technology Consideration

### 2.5.1 Client-Server Computing Technology

Client/server computing refers to the method of distributing computer applications across many platforms. Typically applications are divided between a server providing access to a central data repository and numerous clients containing graphical user interfaces (GUI's) to access and manipulate the data. The client application sends a request across the network for data or the use of a resource. The server application sits idle waiting to service incoming requests from clients. When the server receives a request it performs the necessary action. One server application can support numerous clients, giving each client access to all of the application data. Properly implemented, a client/server application appears to be a stand-alone application.

Client/server systems operate over LANs, WANs, or a combination of the two. Client applications are typically placed on workstations with graphics capability. The primary purpose of the client application is to provide an interface between the application/data and the user. The server application usually resides on a more powerful computer containing a database.

While fulfilling requests the server application prevents inconsistent data conditions caused by more than one client attempting to update the same data. Numerous client applications can run simultaneously on the same workstation. Numerous server applications, or multiple instances of one server application, can run on a server.

### 2.5.1.1 Usage

Client/server computing is typically used in situations where an organization has one or more central data repositories and many user access points. The user access points may or may not be co-located with the central data repository.

Client/server computing is used not only for large distributed systems, it also provides much of the functionality that is normally associated with a LAN. File servers on LANs provide access to directories/files creating the appearance that they reside on the local machine. The local machine is a client sending a request to the server for access to a file or directory. If that file is in use by another client the request will be denied. Print servers are also an example of a client/server implementation. As individual machines issue print requests, they are queued and prioritized by the print server which is responsible for actually sending each job to the printer. The sharing of file and printer resources over a LAN is the most common application of client/server computing.

### 2.5.1.2 State of the Technology

Client/server computing is a well established technology. However because the technology is considered new and complex many organizations have little experience with client/server development and the necessary ongoing support. Client/server application development tools have begun to help address many of the client/server complexities. The state of client/server application development is moving toward the use of very sophisticated CASE tools and application generators. These tools provide mechanisms for analyzing information flow within the organization, designing the application, designing the database for the application, generating the application code, and creating the database. Manual programming of application logic is still required.

Client/server hardware and software technology has reached a mature state, however hardware performance enhancements continue to be realized. The flexible nature of client/server architecture allows old obsolete servers to be replaced by new servers running the same applications. Network and programming standards are at point where applications can quickly be ported from the machine of one vendor to another. Because of the "open" nature of client/server computing, significant competition exists among the major workstation/server manufacturers.

### 2.5.1.3 Strengths

Client/server computing facilitates the development of applications with enhanced access to data. The implementation of client/server systems has enabled organizations to achieve breakthroughs in performance and significantly higher levels of customer satisfaction. The flexibility inherent in client/server architecture allows for more competitive procurements, and easy upgrading of hardware. Client/server implementations force organizations to focus on how information moves within their organization, and how to best use that information to meet the needs of citizens.

### 2.5.1.4 Weaknesses

Client/server computing is a completely different approach from mainframe computing. Using client/server to implement a major system requires a large and very expensive development and conversion effort. Client/server systems are often not implemented until the mainframe system that they are designed to replace is obsolete. Existing information systems (IS) personnel must be retrained on the new technology. Numerous examples of failed or over budget client/server projects exist to give pause to the wary IS executive.

## 2.5.2 Types of Client/Server Computing

### 2.5.2.1 Two Tier (Client/Server)

Two tier software architectures were developed in the 1980s from the file server software architecture design. The two tier architecture is intended to improve usability by supporting a forms-based, user-friendly interface. The two tier architecture improves scalability by accommodating up to 100 users (file server architectures only accommodate a dozen users), and improves flexibility by allowing data to be shared, usually within a homogeneous environment. The two tier architecture requires minimal operator intervention, and is frequently used in non-complex, non-time critical information processing systems.

#### Architecture

Two tier architectures consist of three components distributed in two layers: client (requester of services) and server (provider of services). The three components are

1. User System Interface (such as session, text input, dialog, and display management services)
2. Processing Management (such as process development, process enactment, process monitoring, and process resource services)
3. Database Management (such as data and file services)

The two tier design allocates the user system interface exclusively to the client. It places database management on the server and splits the processing management

between client and server, creating two layers. Figure 38 depicts the two tier software architecture.

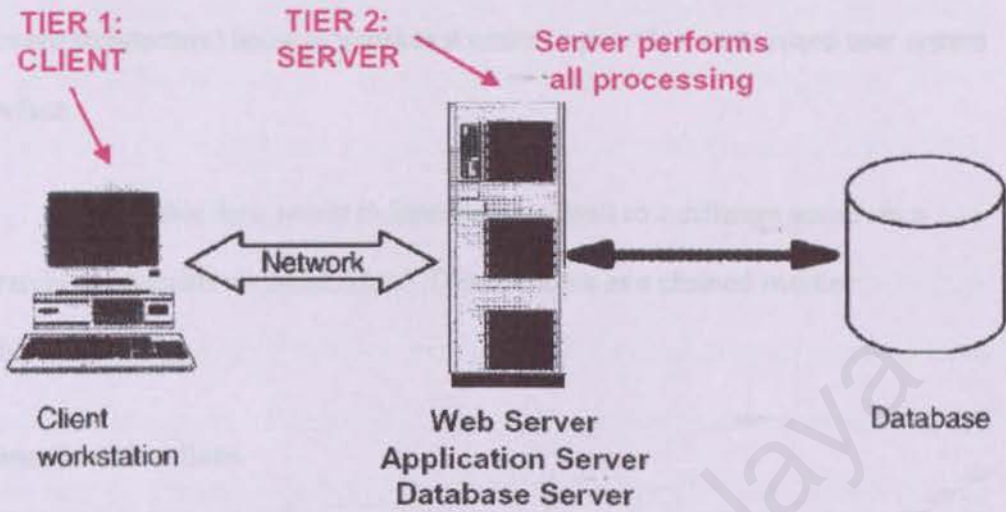


Figure 2-10 : Two Tier Client Server Architecture Design

In general, the user system interface client invokes services from the database management server. In many two tier designs, most of the application portion of processing is in the client environment. The database management server usually provides the portion of the processing related to accessing data (often implemented in store procedures). Clients commonly communicate with the server through SQL statements or a call-level interface. It should be noted that connectivity between tiers can be dynamically changed depending upon the user's request for data and services.

As compared to the file server software architecture), the two tier architecture improves flexibility and scalability by allocating the two tiers over the computer network. The two tier improves usability (compared to the file sever software architecture) because it makes it easier to provide a customized user system interface.

It is possible for a server to function as a client to a different server- in a hierarchical client/server architecture. This is known as a chained two tier architecture design.

### **Usage Considerations**

Two tier software architectures are used extensively in non-time critical information processing where management and operations of the system are not complex. This design is used frequently in decision support systems where the transaction load is light. Two tier software architectures require minimal operator intervention. The two tier architecture works well in relatively homogeneous environments with processing rules (business rules) that do not change very often and when workgroup size is expected to be fewer than 100 users, such as in small businesses.

Two tier client/server architectures have been built and fielded since the middle to late 1980s. The design is well known and used throughout industry. Two tier architecture developments were enhanced by fourth generation languages.

### 2.5.2.2 Three Tier

The three tier software architecture (three layer architectures) emerged in the 1990s to overcome the limitations of the two tier architecture. The third tier (middle tier server) is between the user interface (client) and the data management (server) components. This middle tier provides process management where business logic and rules are executed and can accommodate hundreds of users (as compared to only 100 users with the two tier architecture) by providing functions such as queuing, application execution, and database staging. The three tier architecture is used when an effective distributed client/server design is needed that provides (when compared to the two tier) increased performance, flexibility, maintainability, reusability, and scalability, while hiding the complexity of distributed processing from the user.

The three tier architecture is used when an effective distributed client/server design is needed that provides (when compared to the two tier) increased performance, flexibility, maintainability, reusability, and scalability, while hiding the complexity of distributed processing from the user. These characteristics have made three layer architectures a popular choice for Internet applications and net-centric information systems.



## Architecture

A three tier distributed client/server architecture (as shown in [Figure 28](#)) includes a user system interface top tier where user services (such as session, text input, dialog, and display management) reside.

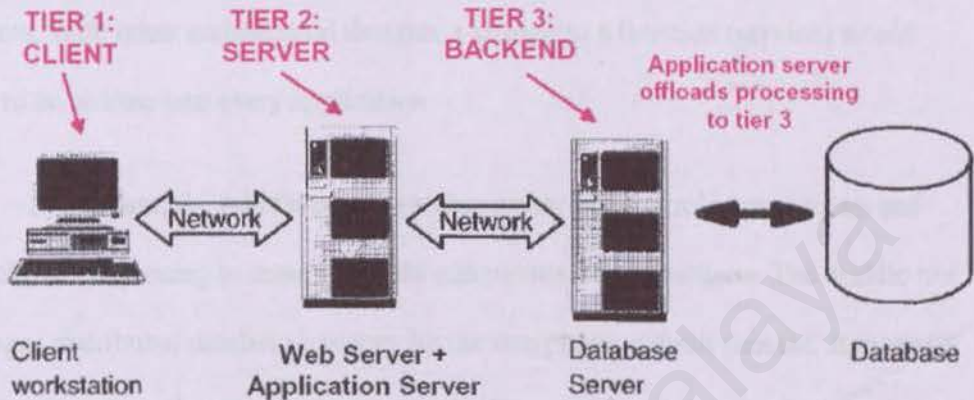


Figure 2-11: Three tier distributed client/server architecture depiction

The third tier provides database management functionality and is dedicated to data and file services that can be optimized without using any proprietary database management system languages. The data management component ensures that the data is consistent throughout the distributed environment through the use of features such as data locking, consistency, and replication. It should be noted that connectivity between tiers can be dynamically changed depending upon the user's request for data and services.

The middle tier provides process management services (such as process development, process enactment, process monitoring, and process resourcing) that are shared by multiple applications.

The middle tier server (also referred to as the application server) improves performance, flexibility, maintainability, reusability, and scalability by centralizing process logic. Centralized process logic makes administration and change management easier by localizing system functionality so that changes must only be written once and placed on the middle tier server to be available throughout the systems. With other architectural designs, a change to a function (service) would need to be written into every application

In addition, the middle process management tier controls transactions and asynchronous queuing to ensure reliable completion of transactions. The middle tier manages distributed database integrity by the two phase commit process. It provides access to resources based on names instead of locations, and thereby improves scalability and flexibility as system components are added or moved.

Sometimes, the middle tier is divided in two or more unit with different functions, in these cases the architecture is often referred as multi layer. This is the case, for example, of some Internet applications. These applications typically have light clients written in HTML and application servers written in C++ or Java, the gap between these two layers is too big to link them together. Instead, there is an intermediate layer (web server) implemented in a scripting language. This layer receives requests from the Internet clients and generates html using the services provided by the business layer. This additional layer provides further isolation between the application layout and the application logic.

## Usage Considerations

Three tier architectures are used in commercial and military distributed client/server environments in which shared resources, such as heterogeneous databases and processing rules, are required. The three tier architecture will support hundreds of users, making it more scalable than the two tier architecture.

Three tier architectures facilitate software development because each tier can be built and executed on a separate platform, thus making it easier to organize the implementation. Also, three tier architectures readily allow different tiers to be developed in different languages, such as a graphical user interface language or light internet clients (HTML, applets) for the top tier; C, C++, SmallTalk, Basic, Ada 83, or Ada 95 for the middle tier; and SQL for much of the database tier.

Migrating a legacy system to a three tier architecture can be done in a manner that is low-risk and cost-effective. This is done by maintaining the old database and process management rules so that the old and new systems will run side by side until each application and data element or object is moved to the new design. This migration might require rebuilding legacy applications with new sets of tools and purchasing additional server platforms and service tools. The benefit is that three tier architectures hide the complexity of deploying and supporting underlying services and network communications.

### 2.5.2.3 N-Tier Architecture

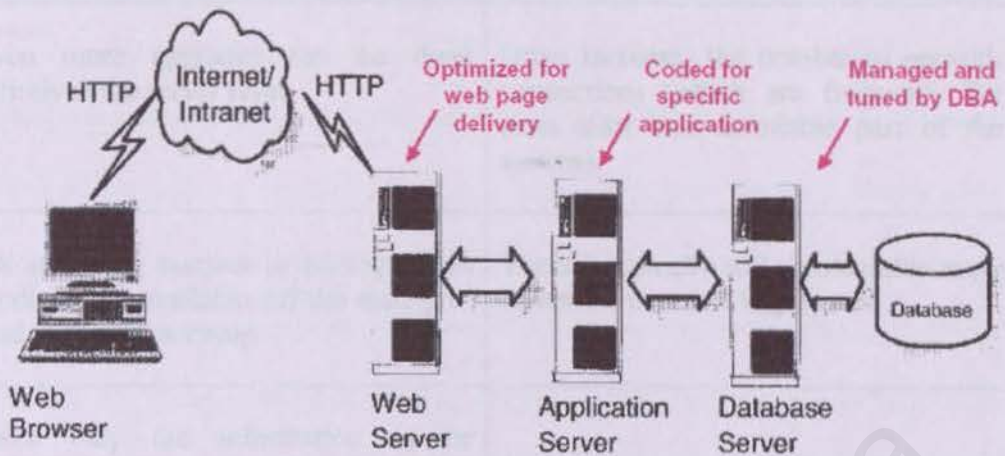


Figure 2-12: N tier distributed client/server architecture depiction

Essentially like 3-tier and theoretically the best of all worlds. This model gets serious about implementing what looks like a three-tier model logically, but instead breaks the components down to their smallest reasonable logical unit of work. If the data services layer is done properly, even the database can be spread across multiple servers and moved around as needed. The only impact is on the data services components that provide information to the moved data. The business services components are obvious to the move, since they only need to know the name of the data services components that supplies the data and what specific method to call.

Advantages	Disadvantages
Even more upgrades can be done entirely at the server level.	Often increases the number of network connections (which are frequently the most slow and unreliable part of the system).
An increasing number of homogeneous products are available off-the-shelf-pre-made software is cheap.	There is typically still considerably more downtime than in a host system.
Since only the information to be displayed is sent on the network, there is little network bandwidth comparative to the client-centric model. The load may, however, be higher between the business-logic and data-services systems if they are on different servers.	
Allows for component-based development which can increase reusability.	
Multiple medium servers are often cheaper than one on large server. The separation of business and data services makes two servers an option.	

Table 2-1 N-Tier

## 2.5.3 Web Server

### 2.5.3.1 Internet Information Server (IIS 5.0)

Internet Information Server (IIS) is a group of Internet servers (including a Web or Hypertext Transfer Protocol server and a File Transfer Protocol server) with additional capabilities for Microsoft's Window NT and Window 2000 Server operating systems. IIS is a Microsoft's entry to compete in the Internet Server market that is also addressed by Apache, Sun Microsystems, O'Reilly, and others. With IIS, Microsoft includes a set of programs for building and administering Web sites, a search engine, and a support for writing Web-based applications that access databases. Microsoft points out that IIS is tightly integrated with the Window NT and 2000 Servers in a number of ways, resulting in faster Web page serving.

IIS includes security features and promises that it is easy to install. It works closely with the Microsoft Transaction Server to access databases and provided control at the transaction level. It also works with Microsoft's Netshow in the delivery of streaming audio and video, delayed or live.

### 2.5.3.2 Personal Web Server (PWS)

Personal Web Server, an abbreviation for Personal Web Server, is a server program for individual PC users who want to share Web pages and other files from their hard drives. PWS is a scaled-down version of Microsoft's more robust Web server, Internet Information Server IIS.

PWS can be used with a full time Internet connection to serve Web pages for a web site with limited traffic. PWS offers:

- **Integration.**

The Personal Web Server turns a Windows 95/98/NT-based personal computer into a low-volume Web server, making it as easy to share HTML and FTP files over intranets and the Internet as it is to share and print document files over a network. The software is fully integrated into the Windows 95/98/98SE/ME/NT taskbar and Control Panel, allowing users to start and stop HTTP and FTP services whenever they want, administer the server or change general options. Microsoft also designed the Personal Web Server to complement its larger and fully compatible Web server products, such as Microsoft Internet Information Server (IIS). The Personal Web Server is also fully complementary to Peer Web Services included with the Windows NT Workstation operating system version 4.0.

- **Easy to install, use and manage.**

The Personal Web Server is designed to install easily in minutes and includes an intuitive HTML-based administration utility that also supports full remote administration. It supports both user-level and local security, ensuring flexible and effective protection of sensitive corporate information. Users can set up the PWS to support Windows NT Challenge/Response encrypted-password transmission.

- **Standards-based technology.**

PWS fully supports existing standards such as CGI and includes the open Internet Server API (ISAPI) extension to the Win32<sup>®</sup> API that is up to five times faster than CGI-based applications. This enables any user to take advantage of ISAPI scripts and CGI scripts

## **2.5.4 Web Application Development**

### **2.5.4.1 ASP (Active Server Pages 3.0)**

ASP is a server based technology from Microsoft Corp. ASP is a famous tool for creating dynamic, interactive web pages.

ASP was first announced to the world by Microsoft on July 16, 1999. The codename for ASP was DENALI. After the launch, Beta version of ASP was released in November the same year. ASP version 1.0 was launched a month later, December 12, 1996. ASP was making headlines again when it was bundled with Microsoft Internet Information Server (IIS 3.0). In 1998, Microsoft launched new web server software, Personal Web Server 4.0 (PWS 4.0) and Internet Information Server 4.0 (IIS 4.0) which supports ASP 2.0. Later Windows 2000 was released and IIS 5.0 was integrated with the operating system. New version of ASP, ASP 3.0 was released.

ASP offers faster and better performance compared to other technologies. ASP can be viewed from any computer and any modern browsers. It is also easier to customize pages to meet individual needs and ASP is very easy to learn.

ASP editors are a lot in the market nowadays. The most commonly used is Microsoft Visual Interdev. Other editors are Microsoft FrontPage, Allaire's Homesite and of course, Notepad.



#### 2.5.4.2 Java Server Pages (JSP)

Java Server Pages or JSP is a technology that is a part of Java family that allows rapid development, easily maintained, information rich dynamic web pages. JSP uses XML-like tags that encapsulate the logic that generates the contents for the page.

JSP is also an extension of Java Servlet technology. Servlets are platform independent, 100% pure Java server-side module that fit into a Web server framework and can be used to extend the capabilities of a Web server with minimal overhead, maintenance, and support. Servlets involve no platform-specific consideration or modifications. Using servlets and JSP technology provides a good alternative to dynamic scripting or programming.

Sun Microsystems has made the JSP specification freely available to the development community.

#### 2.5.4.3 CGI

CGI is a standard for interfacing external programs with Web servers. The server submits client *requests* encoded in URLs to the appropriate registered CGI program, which executes and returns results encoded as MIME messages back to the server. CGI's openness avoids the need to extend HTTP. Most vendors of Web server extension tools continue to support CGI even as more advanced APIs have been added. This is due to the fact that many prewritten scripts are freely available for a variety of platforms and most of the popular Web servers.

CGI programs are executable programs that run on the Web server. They can be written in any scripting language (interpreted) or programming language (must be compiled first) available to be executed on a Web server, including C, C++, Fortran, PERL, TCL, Unix shells, Visual Basic, Applescript, and others. Arguments to CGI programs are transmitted from client to server via environment variables encoded in URLs. The CGI program typically returns HTML pages on the fly (Deep and Holfelder 1996). CGI lets Webmasters add common features, such as counters and date/time displays, on-line order forms, chat pages and search engines.

CGI also has several drawbacks. Each time a CGI script is spawned, it creates an additional process on the server machine, slowing the server's response time. Also, if the CGI script is not set up correctly, security holes can occur on the server, rendering the Web site vulnerable to attacks by hackers. Another problem is that it is difficult to maintain state - that is, to preserve information about the client from one HTTP request to the next (Deep and Holfelder 1996).

CGI is an early Web-to-database integration mechanism that is being replaced by more complex software programs that lie between the Web and database servers.

## 2.5.5 Programming / Scripting Languages

### 2.5.5.1 HTML

HTML is the acronym for the Hypertext Markup Language, which is one of the most widely used computer languages in the world. The popularity and importance of HTML is due to the fact that it is the coding technology used to publish content on the World Wide Web (also referred to as the Internet).

### 2.5.5.2 VBScript

Microsoft Visual Basic Scripting Edition language or know as VBScript, is a simplified version of the Visual Basic and Visual Basic for Applications family of programming languages. VBScript is a scripting language or more precisely a "scripting environment", which can enhance HTML Web pages by making them active, as compared to a simple static display. Specifically, VBScript was created by Microsoft to use either as a client-side scripting language for the Microsoft Internet Explorer (versions 3.0 and later) or as a server-side scripting language with the Microsoft Internet Information Server (versions 3.0 and later).

The advantage for using the server-side approach is that the VBScript is processed by the server before it is transmitted to the client. Therefore, the client only receives an HTML page and we do not have to concern ourselves as to whether the browser can interpret the VBScript. In contrast, by using the client-side approach, you purposely transfer the work load to the browser in order to reduce the work load of the server.

Unfortunately, older or non-Microsoft browsers may not be able to correctly interpret and display the transmitted file. In addition to this, the source code is exposed to the browser user. On the brighter side, a client-side program can produce a more-responsive application, since user input can be processed on the client machine, and not sent back to the server for processing. The true importance of VBScript is that it is the default language of Active Server Pages (ASP).

### 2.5.5.3 JavaScript

The JavaScript language was developed by the Netscape Communications Corporation. Microsoft recognizes JavaScript as JScript. JavaScript is a cross-platform, object-based scripting language that was originally designed for use in Netscape Navigator. Indeed, versions 2.0, and later, of Navigator can interpret JavaScript statements that are embedded within HTML code. When a request is made to see a page, the HTML code that defines the requested page along with the embedded JavaScript statements, are sent by the server to the client. Navigator interprets the HTML document and executes the JavaScript code. The resultant page is displayed for the client. It is important to understand that this interpretation occurs on the client-side rather than the server-side.

## 2.5.6 Database Management System

### 2.5.6.1 SQL Server 7.0

SQL Server was first launched by Microsoft 1988.the project was a joint venture between Microsoft and Sybase. In the early 1990s, Microsoft was developing SQL Server for the NT platform. Realizing the potential of SQL Server, Microsoft made the decision that SQL server should be tightly coupled with the Windows NT operation system in 1992. In 1993, Windows NT 3.1 and SQL Server 4.2 for NT were released. Microsoft's philosophy of combining a higher performance database with an easy to use interface proved to be very successful. Microsoft quickly became the second most popular vendor of high-end relational database software. In 1994, Microsoft and Sybase formally ended their partnership.

In 1995, Microsoft released version 6.0 of SQL Server. This release was a major rewrite of SQL server's for technology. Version 6.0 substantially improved performances, provided built in replication and delivered centralized administration. In 1996 Microsoft released version 6.5 of SQL Server. This version brought significant enhancements to the existing technology and provided several new features. In 1997, Microsoft released version 6.5 Enterprise Edition. This version included for GB RAM support, 8 way processor support and Microsoft Cluster support. In 1998, Microsoft released version 7.0 of SQL Server. This version is a complete rewrite of the core engine and administration components. According to the SQL Server development team, the changes to the database engine are designed to provide an architecture that will last for the next 10 years.

Stated below are the several key features of SQL Server 7.0:

- Architectural Enhancements
- Data Transformation Service (DTS)
- Security Enhancements
- Performance and Scalability Improvements
- Backup and Restoration Improvements
- Simplified Administration.
- Replication Enhancements.
- Internet Enhancements
- Distributed Queries
- Data Warehousing Enhancements.

#### 2.5.6.2 Microsoft Access 2000

Microsoft Access 2000 is the relational Database Management System used to create relational databases. Together with the ODBC driver for Microsoft Access, data can be retrieved from the databases in Client-Server based system. With the new capabilities in the Microsoft Access 2000 database management system, data dropped into HTML pages that can be shared and manipulated on the World Wide Web, making it easy for users to get important information to others.

Microsoft Access 2000 also supports OLE DB, a data access standard that allows Access 2000 to connect directly to SQL Server. Access 2000 can be used as the interface for SQL Server database.

## 2.5.7 Data Access Models

There are five different access models that Microsoft considers being up to date for accessing data. These include:

### 2.5.7.1 RDO: Remote Data Objects

This is a very thin wrapper around ODBC. It was built specifically with VB in mind, but is also usable in J++. RDO is very easy to use in comparison to programming to the ODBC API directly, but doesn't lose all that much in the speed category. There is a ton of legacy code in place using this technology-it is fast and reliable.

### 2.5.7.2 ADO: ActiveX Data Objects

Each version of RDO seems to improve performance and add new features, but ADO still cannot be compared on a one-to-one basis RDO. RDO had a few features that will never be in ADO. (ODBC handles, for example, can't be added because of OLE DB being in between ADO and ODBC), and ADO has some really cool features with persistent record sets, filters, sorts (without going back to the server) and others that RDO never had.

Unlike RDO, which was based on ODBC, ADO is based on OLE DB. This provides a level of flexibility that ODBC alone cannot offer, but it comes with more than just a few headaches.

ADO is now competitive, if not faster, in speed compared to RDO and has a very robust feature set. It is nowhere near RDO in reliability, but Microsoft has made and continues to make a substantial investment in ADO and OLE DB.

### 2.5.7.3 ODBC: Open Database Connectivity

ODBC provides a way of gaining cross-platform access to database information. It is quite fast and allows most of the mainstream standard SQL statements regardless of what the back-end expects for syntax. The major shortcoming for ODBC is that it is very much oriented to tabular data and doesn't deal with non-standard data such as a directory structure or a multi-sheet database.

### 2.5.7.4 OLE DB

OLE DB is an attempt at having an open standard to communicate with both tabular and non-tabular data. OLE DB uses what is called a provider. A provider is a lot like an ODBC driver except that it is relatively self-describing. That is, it is able to tell the application which uses it what kind of functionality it supports.

OLE DB is the foundation under ADO. It is very fast indeed when not used with ADO, but, since it deals with a number of items that aren't compatible with VB, OLE DB being used directly by C++ programmers. It is far more of pain to program in than ADO, but it is much faster by itself than when used in conjunction with ADO.

Java Database Connectivity (JDBC)



#### 2.5.7.5 DB-Lib

Prior to version 7.0, this was the native way in which the SQL Server did all of its talking between the main host and client and utility applications. It is actively supported, but only be enhanced to the extent necessary to maintain backward compatibility as SQL Server moves forward. Microsoft will pull support for this access method at some point in the future, but they also acknowledge that there are too many legacy applications using DB-Lib to figure on dropping support for it.

#### 2.5.7.6 VB-SQL

This was based on an old wrapper that was written for VB to make many DB-Lib functions available for VB programmers.

#### 2.5.7.7 DAO: Data Access Objects

This object model can be considered clunky, slow and just plain outdated.

## 2.6 Summary

This chapter covers the literature review of this project. Existing systems are studied carefully to find out what are the features of each system. This is to create a better system.

This chapter also covered the technologies that can be used to develop the system. Each technology was analyzed and will be chosen based of certain criteria that will be explained in the chapter 4.

University of Malaya

CHAPTER 3

Methodology

### 3.1 Introduction

This chapter will discuss about the methodology used to develop the system. This chapter covers the software development models as well as the analysis of the system.

#### 3.1.1 Software Lifecycle Models

Life cycle models describe the interrelationships between software development phases. The common life cycle models are

- ✓ Spiral Model
- ✓ Waterfall Model
- ✓ Throwaway Prototyping Model
- ✓ Evolutionary Prototyping Model
- ✓ Incremental/iterative Development
- ✓ Reusable Software Model
- ✓ Automated Software Synthesis

## 3.2 Software Development Model For Student Attendance System

### 3.2.1 Waterfall Model

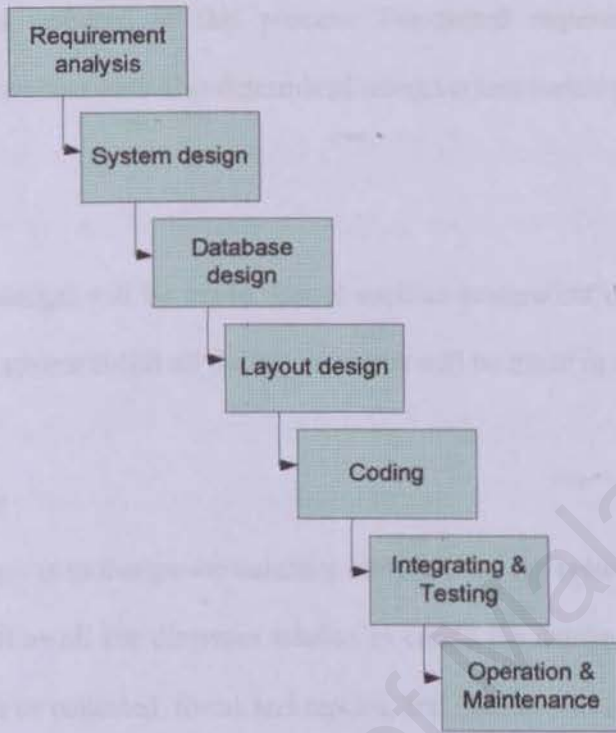


Figure 3.1 The Waterfall Model

The waterfall model is one of the first models to be introduced. The stages of this model are cascading from one another. Each development stage must be completed before the next stage begins. This model gives a high-level view of what goes on during development as well of the sequences of each activity.

## Requirement Analysis

Requirement analysis is the first stage in the development process. All the information was gathered in this process. Functional requirements and non-functional requirements were also determined using various techniques.

## System Design

Overall design will be made. Issues such as integration of the system and making sure the system fulfill all the requirements will be made in this phase.

## Database Design

This phase is to design the database structure for the system. All the tables, relations, as well as all the diagrams needed to create the database. Activities like data that need to be collected, forms and reports, and queries will be designed in this phase.

## Layout Design

This phase will design the interface for this system. The user is designed properly to ensure that it is highly interactive, user friendly and easy to use.

## Coding

Coding is a process that will transform the algorithms defined during the design stages into a computer using programming languages.

## Integration & Testing

Testing is exercising the developed software to check errors and to make sure the system meets the defined requirements and goals. This system used 3 different tests, unit testing, integration testing and system testing.

1. Unit testing is the testing process to check each module to check for bugs. Each module will be checked thoroughly to ensure each module behaves according to the designed specifications without any error.
2. Integration testing is the testing process that will test the interconnection of each tested modules. The main point of doing this integration testing is to make sure that the tested modules behave as well as independently tested during the unit testing previously.

## Operation & Maintenance

Maintenance activities are a bit similar to development process, analyzing requirements, evaluating overall design, writing or reviewing code, testing changes, and updating documentation. There are 4 types of maintenance:

- ✓ Correcting maintenance
- ✓ Adaptive maintenance
- ✓ Perfective maintenance
- ✓ Preventive maintenance

### Advantages of Using Waterfall Model

1. Easier to understand for customers who is not very familiar with software development project
2. This model gives the high level view of the software development
3. This model is the basis of other software development models

### Disadvantages of Using Waterfall Model

1. The waterfall model does not explain how the coding is implemented
2. The model unable to explain changes for product or activities
3. The waterfall model is specifically hardware development model, it is unable to define software development model as a solution

### 3.2.2 The Prototyping Model

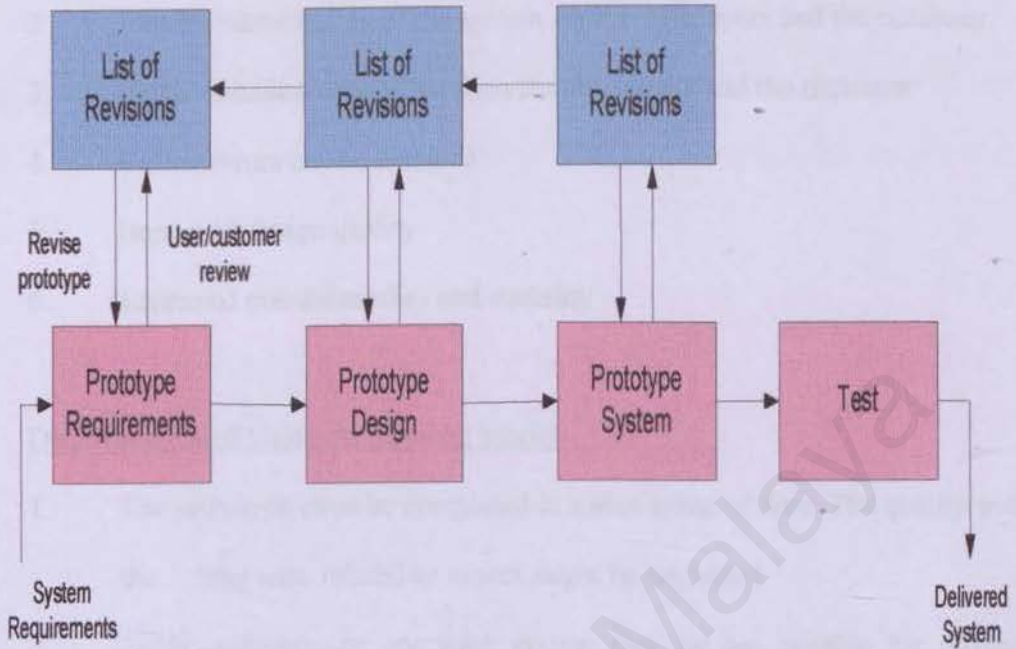


Figure 3.2 The Prototyping Model

Prototyping is a process to create a basic model of the system which demonstrates the concept, try out design options and find out more about the problem and its possible solutions. The prototyping model allows the system to be completed very quickly. The main purpose of doing the prototype is to understand and clarify certain issues. Developing a prototype usually leads to improvements in the specifications of the system.



### Advantages of Using Prototype Model

1. The fastest way to determine the system, the functions, etc.
2. Better understanding of the system for the developers and the customer
3. Better communications between the developers and the customer
4. Earlier errors can be detected
5. Improved design quality
6. Improved maintainability and usability

### Disadvantages of Using Prototyping Model

1. The prototype must be completed in a short space of time. The quality and the long term reliability aspect might be neglected
2. Tools, software, or operating system may be not suitable for further development after the prototype is ready
3. Cost of building a prototype is very high

Figure 3.3 The Iterative Model with Prototyping

### 3.2.3 The Waterfall Model with Prototyping

The development model chosen for this system is a combination of Waterfall Model and Prototyping Model. Various factors such as time, type of system, and development techniques were considered before choosing this waterfall model with prototyping. The need of a prototype was also a factor choosing this development model.

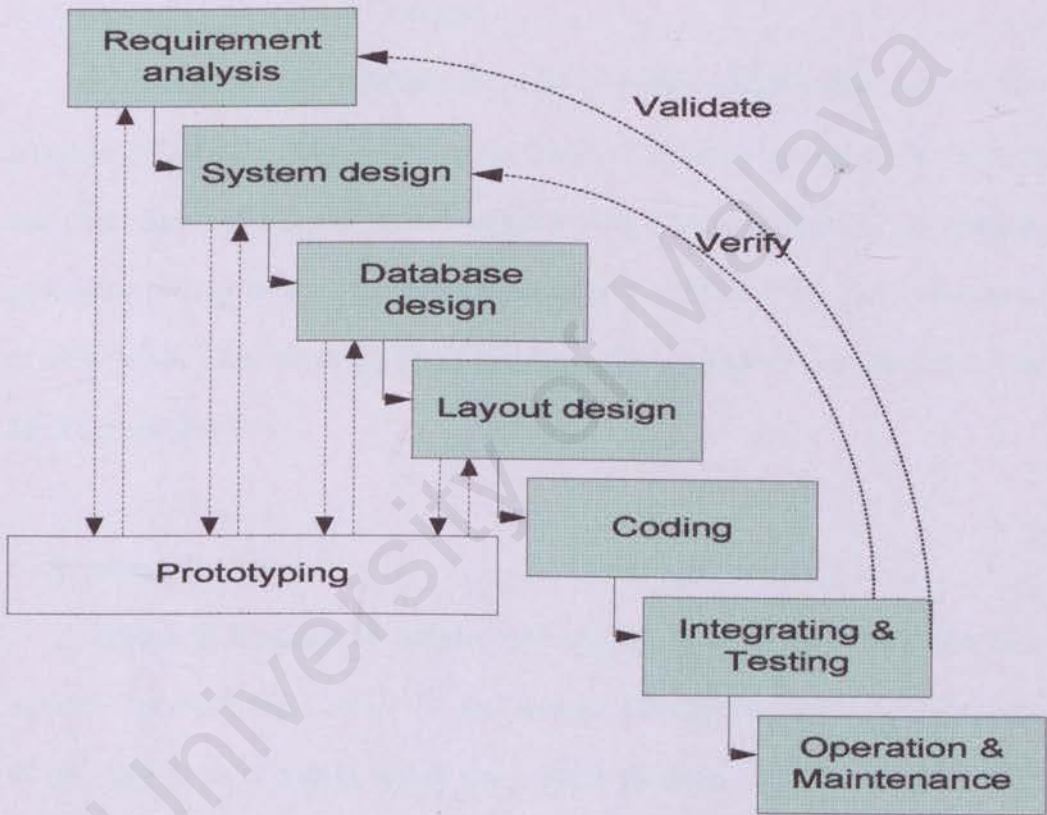


Figure 3.3 The Waterfall Model with Prototyping



## SISTEM KEHADIRAN PELAJAR STUDENT ATTENDANCE SYSTEM



development of Student Attendance System as it provides systematic steps in planning an information system.

### 4.1.1 System Architecture Concepts

A client-server architectural model is simply a relationship between two processes which are made separate to the users. The model has many advantages and could increase the performance of the system. For example, the database processing is done on the back end, the speed of the database is not tied to the speed of workstation. This can save a lot of cost since the workstation only need to be run the front-end software.

### 4.1.2 Operating System

Microsoft Windows XP will be used as server in developing the system as it supports Microsoft SQL Server 7.0 and Internet Information Services 4.0, Windows 95 and XP as an operating system for a single machine. Thus, it can be used for initial development of the Student Attendance System and as a client. Apart from that, Microsoft Windows XP supports a lot of downloadable software. Compared to UNIX, Microsoft Windows XP is easier to install. The UNIX operating system is not as easy as Microsoft Windows XP which has a user-friendly interface that makes the development easier.

## CHAPTER 4

# System Analysis

### 4.1 Development Concepts

The methodology for information system planning is suitable for guiding the development of Student Attendance System as it provides systematic steps in planning an information system.

#### 4.1.1 System Architecture Concepts

A clients server computational model is simply a relationship between two processes which one make request to the others. The model has a lot of advantages and could increase the performance of the system. Since the bulk of the database processing is done on the back end, the speed of the DBMS is not tied to the speed of workstation. This can save a lot of cost since the workstation only need to be run the front-end software.

#### 4.1.2 Operating System

Microsoft Windows XP will be used as server in developing the system as it supports Microsoft SQL Server 7.0 and Internet Information Server 4.0. Windows 98 and 2000 is an operating system for a single machine. Thus, it can be used for initial development of the Student Attendance System and as a client. Apart from that, Microsoft Windows XP supports a lot of software especially the free downloadable software. Compared to UNIX, Microsoft Windows XP is easier to install. The UNIX operating system uses a lot of command to be entered rather than Microsoft Windows XP which has a user friendly interface that makes the development easier.

### 4.1.3 Web Programming Language

#### 4.1.3.1 Active Server Pages 3.0

Active Server Pages or ASP is used to develop this system. Various factors were considered before choosing this ASP technology. Stated below are the advantages of using ASP.

- Supported by most of the web server

ASP is a technology pioneered by Microsoft, so it is supported by most of the web servers including Internet Information Server (IIS) and Personal Web Server (PWS). Chili!Soft has a developer edition called Chili!Soft ASP which enables ASP to run on web servers from Apache, Lotus, Netscape, O'Reilly and Microsoft, running on Microsoft, Sun and IBM systems. Halcyon Software also developed a product called Instant ASP, which runs as Java servlet on a web server to provide ASP support.

- Language-independent

ASP supports with multiple scripting languages such as VBScript and JavaScript. This offers flexibility for the developers in the development process.

- More secure

ASP coding is not available for end-users to view through their browser. In other words, ASP is more secure than writing in pure HTML. Coding in ASP will ensure that sensitive database queries or formulas are kept away from users.

- Browser independent

ASP is supported by most modern web browsers and ASP supports client-side scripting and server side scripting. ASP allows users to run a program in programming languages that are not supported by certain browsers, such as VBScript. ASP enables developers to create dynamic web pages without resource to client-side languages such as Java applets, DHTML, ActiveX controls, and all which is browser specific.

University of Malaya

#### 4.1.4 Database Management System

##### 4.1.4.1 Microsoft SQL Server 7.0

Microsoft SQL Server has been chosen as the database management system for this development project. Main reason of choosing SQL Server is it supports client-server technology magnificently. The other reasons of choosing SQL Server are stated below:

- Compatible with ASP

SQL Server is a Database Management System (DMBS) developed by Microsoft together with ASP and other Microsoft products.

- User Friendly Interface

SQL Server provides an easy menu driven interface that will allow users to issue commands or queries easily.

- Easy To Use

SQL Server includes a set of administrative tools that can manage the database in large volume easily such as SQL Enterprise Manager, SQL Agent, SQL Profiler and so on.

- Scalability

The same database can be used across platforms ranging from laptop computers running Windows 95 or higher or even multiprocessors servers running Windows NT Server.

## 4.1.5 Scripting Language

### 4.1.5.1 VBScript

VBScript is based on Visual Basic programming language developed by Microsoft. VBScript is chosen as the scripting language and the reasons are as following:

- Easy to learn

The project duration is quite short, so time management is crucial to this project. VBScript is easier to learn and master compared to JavaScript of Jscript as recognized by Microsoft.

- Features of VBScript

VBScript has all the abilities to provide scripting, automation and customization capabilities for web browsers. VBScript pages responds intelligently to user actions and VBScript is both client-side scripting as well as server-side scripting.



## 4.1.6 ASP Editor

### 4.1.6.1 Microsoft Visual Interdev 6.0

Numerous text editors and applications can be used to create ASP pages nowadays. Microsoft Visual Interdev 6.0 a part of Microsoft's suite of development tools known as Microsoft Visual Studio. Advantages of Microsoft Visual Interdev:

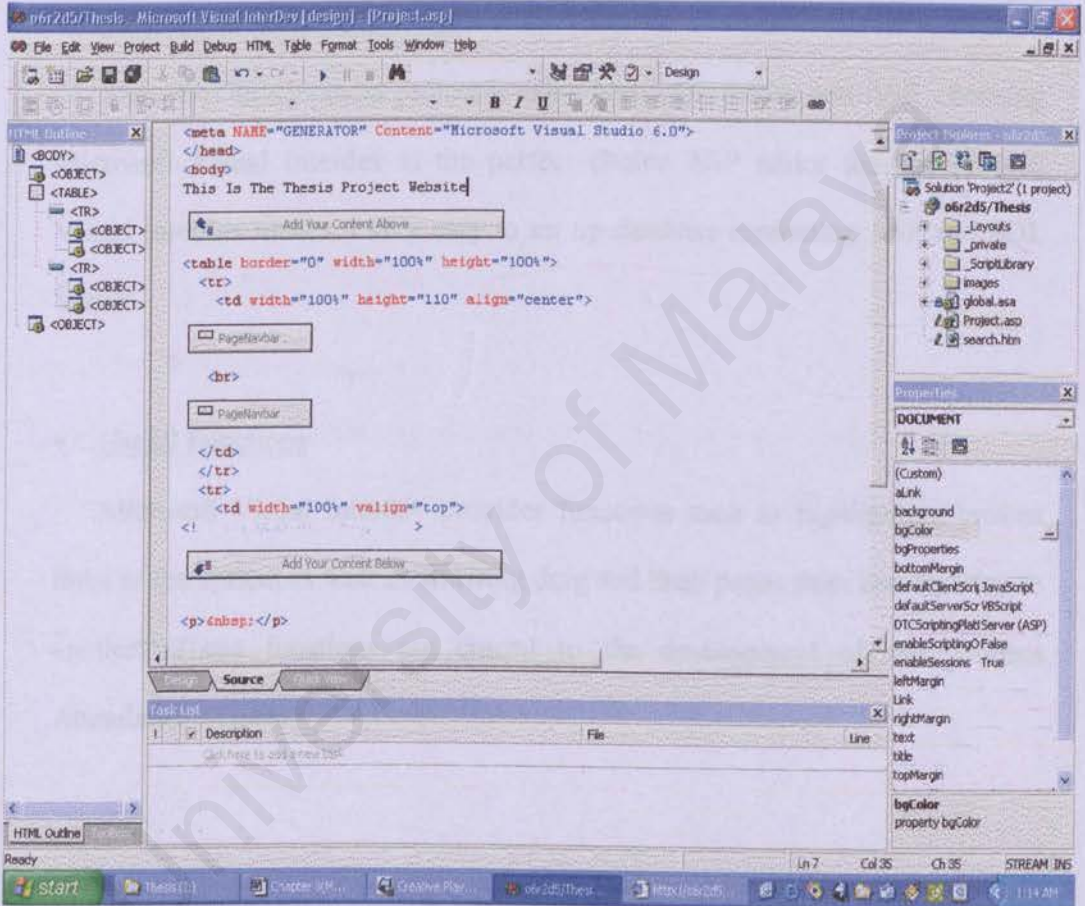


Figure 4-1 Screenshot of Microsoft Visual Interdev 6.0

- Good Graphical User Interface (GUI)

This development tool has a very good and user friendly interface. The approach for the interface is WYSIWYG (What You See Is What You Get). Links, pictures, sounds can be inserted without having to write a single line of HTML code.

- Strong links with SQL Server

Since this development project is using both SQL Server and ASP, choosing Microsoft Visual Interdev is the perfect choice ASP editor for this project. Visual Interdev makes it very easy to set up database combining ASP and SQL Server.

- Useful Functions

Microsoft Visual Interdev provides functions such as highlighting broken links to the system as well as allowing drag and drop pages from one location to another. These functions are crucial to the development of this Student Attendance System.

## 4.2 System Requirements

The requirements of the system can be identified or classified into 2 categories, functional requirements and non-functional requirements.

Functional requirement is the interaction between the system and the environment. It also defines how the system will behave given certain stimuli.

Non-functional requirement is the limits or the constraints for constructing a solution to the problem. Non-functional requirements also define system properties and constraints under which a system must operate.

### 4.2.1 Functional Requirements

The following are some of the functionality of Student Attendance System

- ✓ The system is a web-based system
- ✓ The system is able to support for the growth and future enhancement.

Table 4-1 shows functional requirement in more details.

Module	Requirements
Students Module	<ul style="list-style-type: none"> <li>• Administrator's/lecturer's login is required to make any changes to this module.</li> <li>• Administrators/lecturers are able to make changes or modifications to the student's module, such as ADD,SEARCH,EDIT,DELETE</li> </ul>
Attendance Module	<ul style="list-style-type: none"> <li>• Users (lecturers) key in the daily attendance of students to classes, tutorials while the system automatically store in databases.</li> <li>• This module will provide the function to search the records.</li> <li>• Reports will be generate based on per student, per class, less than 80%</li> </ul>

Table 4-1 Functional requirement

## 4.2.2 Non-Functional Requirements

- User Friendly

The system developed must be user friendly and easy to use. These features will enable new users of the system spend less time to know about the system and how to use it. The user interface must be designed so that it can be used by the non-technical staffs or even new users can use it efficiently.

- Reliable

Reliability is the extents to which a system can be expected to perform its intended function with required precision and accuracy. Therefore, the system should be reliable in performing its daily functions and operations.

- Security

To maintain security of the system, only authorized user with login and password will be allowed to modify the database. Normal users are able to view selected data only. Administrator's Login will be required to make any modifications for this system.

- Systematic

The system must be able to produce reports requested by the users or administrators in a systematic way. Records can also be retrieved easily and effectively.

- Robustness

Robustness refers to the quality that causes a system to be able to handle. This system must be designed to be robust, or in other words the system is able to avoid disaster or behaves accordingly when facing unexpected circumstances such as given improper data by the users.

- Accuracy

This functional refers to the precision of several computations, calculations and controls. This is important as to ensure the system generates a reliable and accurate reports.

### 4.3 Software Requirements

This system will be implemented using a client-server technology, so the software requirements for this system will be categorized into 2 requirements, server side and client side.

#### Server Side

A server will need to run and host this system, so appropriate software should be installed in the server:

- Windows XP platform
- Internet Information Server ( IIS 5.0 )
- Microsoft SQL Server 7.0

#### Client Side

Requirements for the client side are not as complicated as the server. Given below are the software requirements for the client side:

- Windows 9x / Windows Me / Windows 2000 / Windows XP.
- Web browsers such as Internet Explorer, Netscape and Opera.

#### 4.4 Hardware Requirements

The hardware requirements for this system can be categorized into 2, server side and client side as this system is using a client-server technology.

##### Server Side

Stated below are the server side hardware requirements of the system.

Processor	Pentium III or above
Memory	128 MB or above
Hard Disk	5 GB or above
Others	Internet Connection

Stated below are the hardware requirements for the client side.

##### Client Side

Processor	Pentium 166 or higher
Memory	32 MB or higher
Hard Disk	2 GB or higher
Others	Internet connection





discipline to prevent difficulties in developing a physical solution that will be realized. It involves going from requirement analysis phase. In designing a system, various factors need to be put in consideration such as the ability of the system to function and develop.

System design can be divided into six areas.

- ✓ Architecture Design
- ✓ Layout Design
- ✓ Input Design
- ✓ Output Design
- ✓ Database Design
- ✓ System Functional Design

University of Malaya

## CHAPTER 5

# System Design

### 5.1 System Design

System design is process that has been defined to be a technique and various discipline to prevent difficulties in developing a physical condition that will be realized. It develop from the requirement analysis phase. In designing a system, various factor need to be put in consideration such as the ability of the system to functions and develop.

System design can be divided into six areas :

- ✓ Architecture Design
- ✓ Layout Design
- ✓ Input Design
- ✓ Output Design
- ✓ Database Design
- ✓ System Functional Design

## 5.1 Architecture Design

The Student Attendance System is designed to be a client server web application to achieve the requirement specification. It can be divided into three different tiers.

The first tier, which is the user interface layer, is responsible for retrieving input and presenting data, such as the student profile. The main component in this layer is Internet browser such as Internet Explorer.

The second or middle tier is the application layer or service tier. It is responsible for linking the first and third tier together. This is where the web server resides.

The third tier is the data repository. This layer consists of a database server and data store. This is where the application stores all records.

The first layer communicates with the second layer through Hypertext Transfer Protocol (HTTP). This protocol is used to transfer web pages from the web server at the second tier to the browser at first tier. The communication between the second and the third tier is done through selected data access model connection. Figure depicts the system tier of the Student Attendance System.

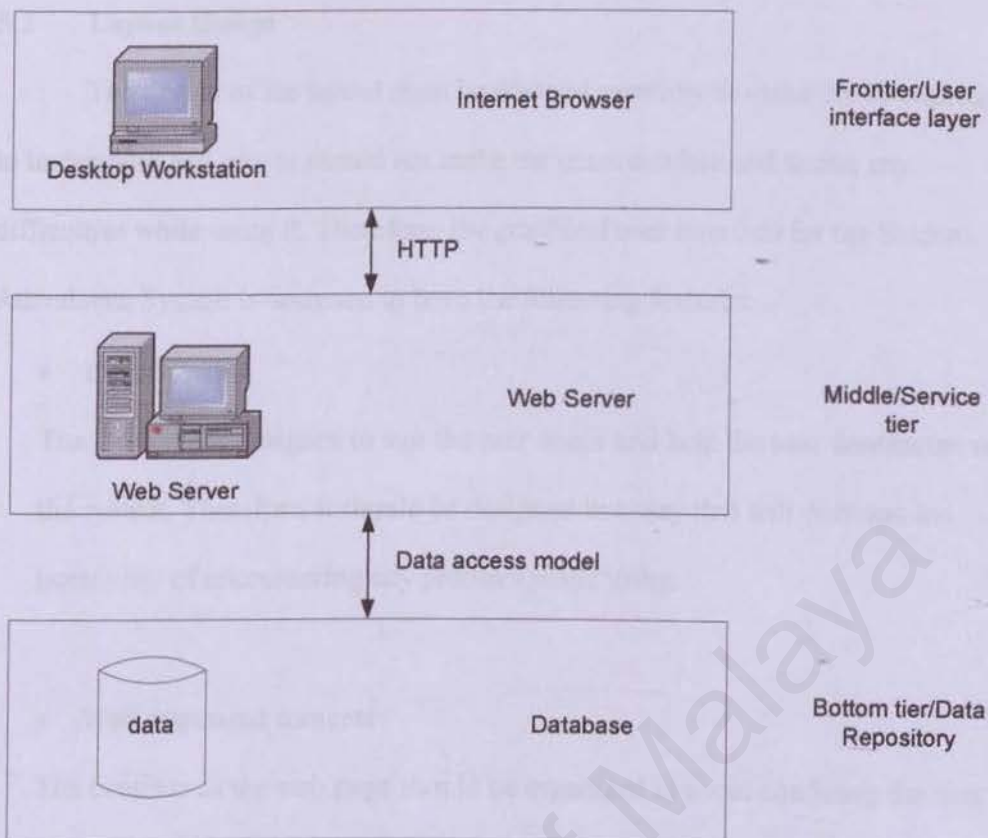


Figure 5-1 System tier of Student Attendance System

## 5.2 Layout Design

The design of the layout must be planned carefully to make the system easy to understand and use. It should not make the users confuse and facing any difficulties while using it. Therefore, the graphical user interface for the Student Attendance System is designed to have the following features:

- User friendly

The interface is designed to suit the user needs and help the user familiarize with the system. Therefore, it should be designed in a way that will decrease the possibility of encountering any problem while using.

- Well organized contents

The contents of the web page should be organized to avoid confusing the user. For this reason, the content should be grouped according to their category and a menu is displayed to help the user in finding a page.

- Easy to navigate

The system should provide enough links for user to navigate from page to page. This will avoid the user in doing more steps to accomplish a single task and hence wasting the time.

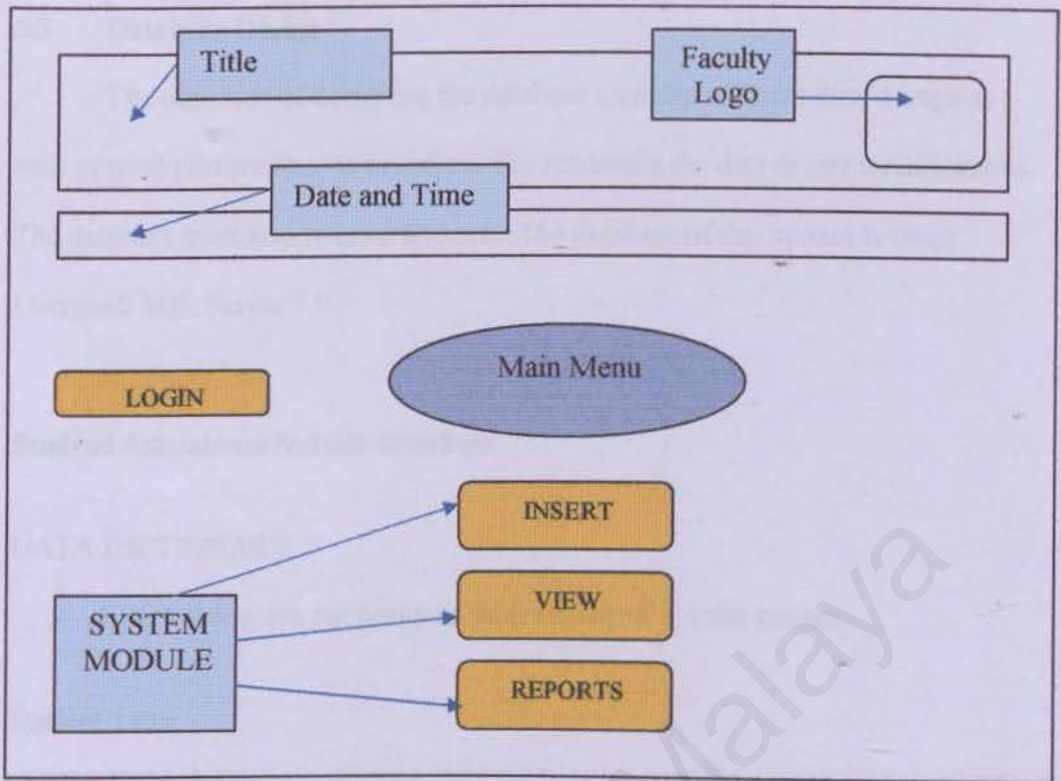


Figure 5.2 Main Menu For Student Attendance System

### 5.3 Database Design

The objective of designing the database includes efficient data storage as well as good performance in processes like retrieving the data or any modifications.

The database must also be reliable and safe. The database of this system is using Microsoft SQL Server 7.0.

#### Student Attendance System Database

##### DATA DICTIONARY

Listed below are the designed tables involved for the system:

##### Student Table

Field Name	Description	Data Types	Size
MtxNo	Matrix Number	Number	10
Name	Name	Text	50

##### Attendance Table

Field Name	Description	Data Types	Size
Attend	Remark	-	-
Date	Date	Date/Time	15

### Login Table

Field Name	Description	Data Types	Size
LoginID	Login Identification	VarChar	20
Pwd	Password	VarChar	20
LevelID	Level Identification	Number	10

### Subject Table

Field Name	Description	Data Types	Size
SbjctCD	Subject Code	Text	10
SbjctName	Subject Name	Text	50




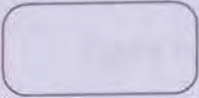
## 5.4 System Functional Design

### Sections Design and Data Flow

The sections design is documented through a variety of design diagrams. A design diagram is a graphics or visual representation of a structure. A design diagram includes data flow diagram (DFD), decomposition diagrams, decision trees and other items (Silver & Silver, 1989). For RRS, DFD was chosen to represent the system.

A DFD is a graphic illustration that shows the flow of data and logic within a system. DFD are composed of four basic symbols. One widely used convention adheres to symbols developed by Edward Yourdan; a similar notation has been developed by Chris Gane and Trish Sarson (Silver & Silver, 1989).

For this project, symbol developed by Gane and Sarson is used to represent the system. Table below describes the symbol using Gane and Sarson method and figure shows the context diagram for Student Attendance System.

Symbol	Name	Description
	Entity	External sources or destination of data, interact with the system but are outside the boundary.
	Process	It represents the transformation or processing of information within a system.



	Data Store	It is used for showing the data storage or referenced by a process.
	Data Flow	It is use to show the movement of data from an origin to a destination with the head of arrow pointing towards the destination.

Table 5-1 Symbols using Gane and Sarson

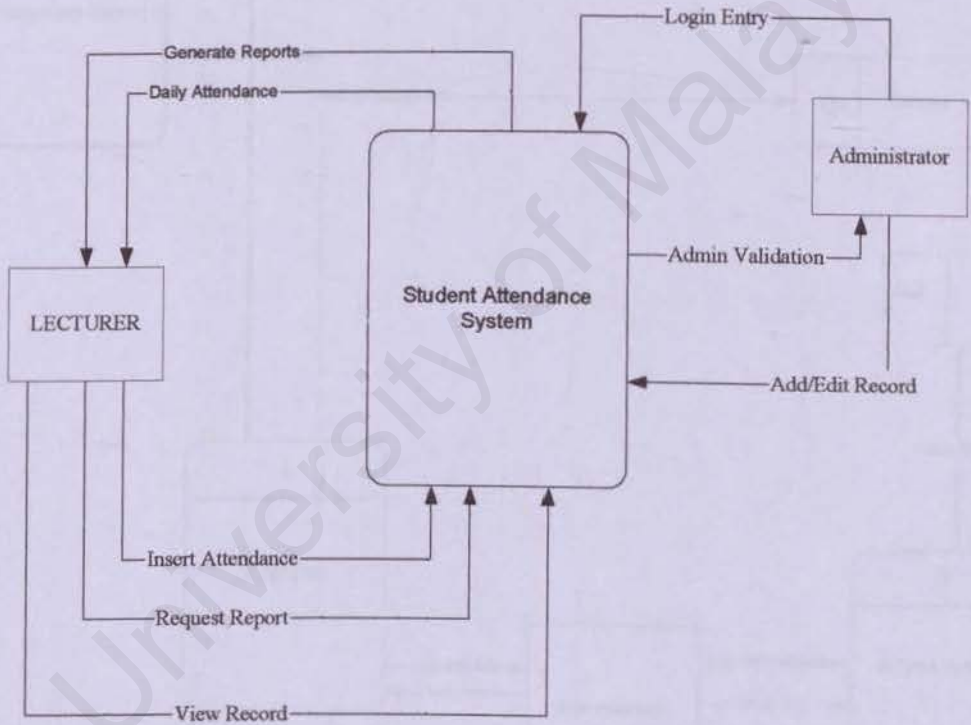


Figure 5-3 Context Diagram For Student Attendance System

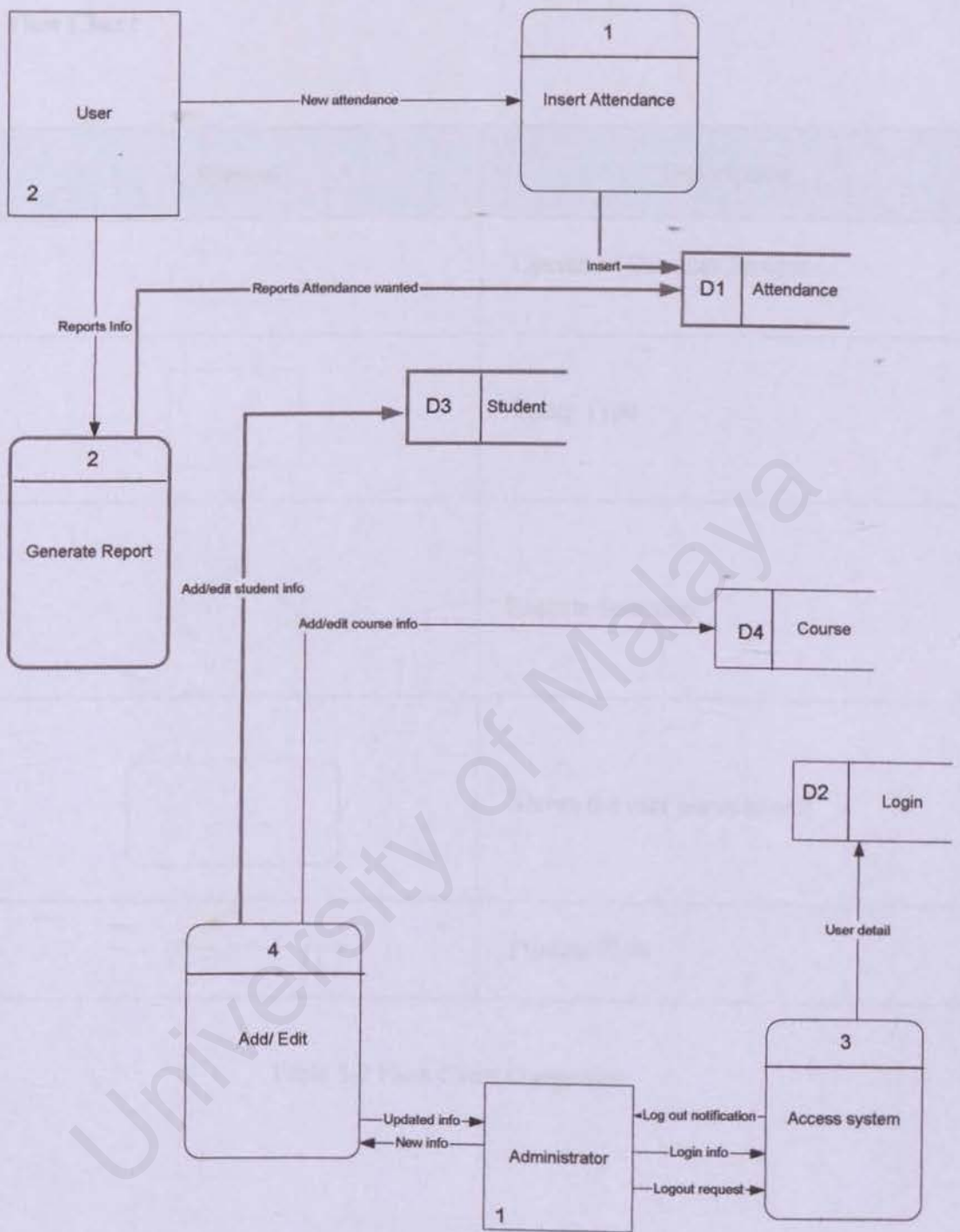


Figure 5-4 Zero-Level Diagram For Student Attendance System

## Flow Chart

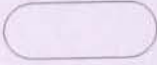
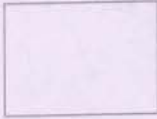

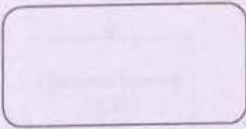

Symbol	Description
	Operation Executes Process
	Entity Type
	Execute Selection
	Shows the user wants to exit
	Process Flow

Table 5-2 Flow Chart Component

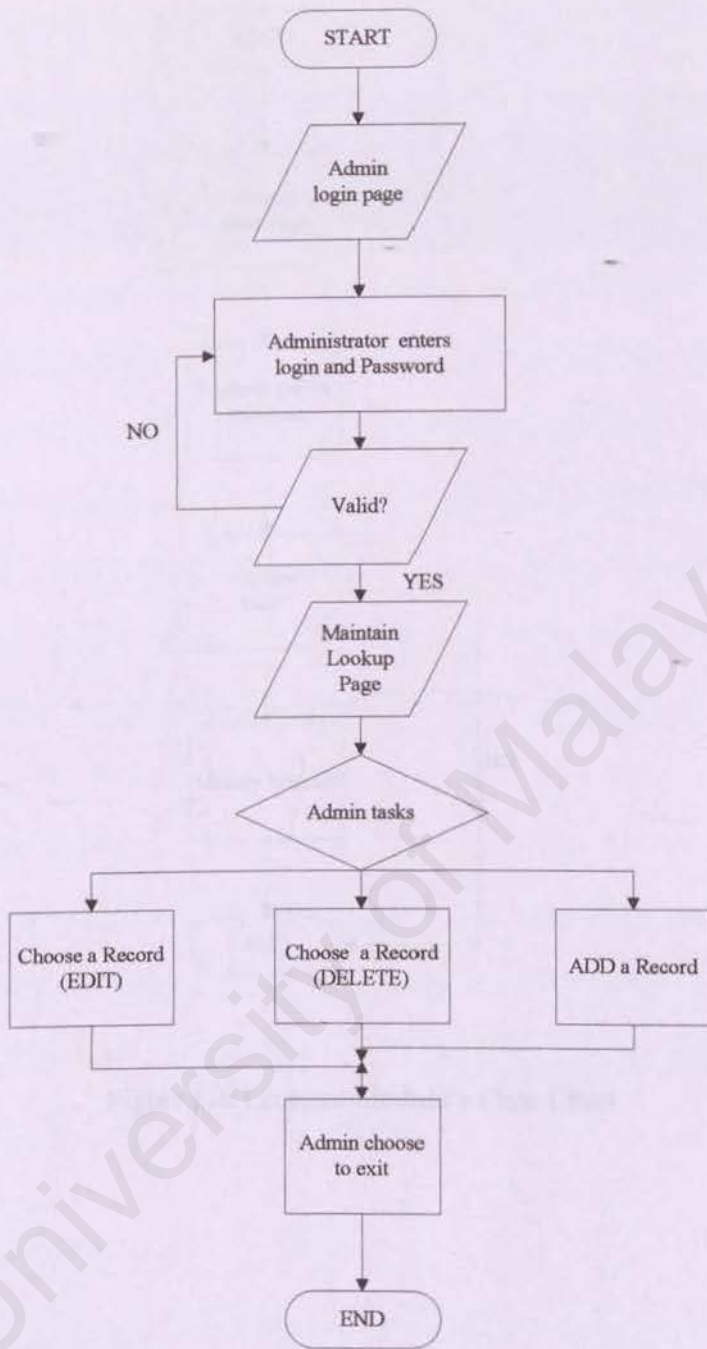


Figure 5-5 Administrator Module's Flow Chart

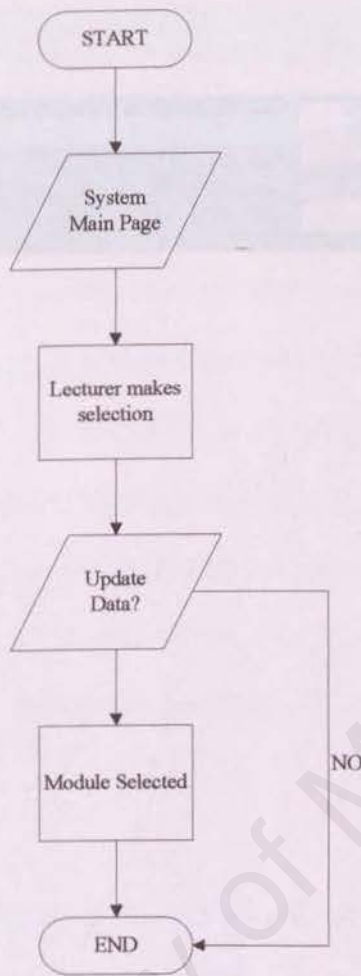


Figure 5-6 Lecturer Module's Flow Chart



## SISTEM KEHADIRAN PELAJAR STUDENT ATTENDANCE SYSTEM



System implementation is a process of converting all the requirements and designs into program code. Before implementation involves the translation of the software representation produced by the design into executable readable form.

A good programming technique is essential for developers to ensure the quality of software. One of the advantages of good programming technique will make the codes easy to write, maintain as well as to modify in the future, which is very important factor for future software development.

### 6.1 System Coding

In this stage, the programs are written and user interfaces are developed and the database is initialized with data. The components built during development are put into operational use. The system is built according to the original design that was done.

#### 6.2.1 Coding Style

Coding style is an important attribute of source code and it indicates the readability of a program. An easy to read source code

## CHAPTER 6

# System Implementation

### 6.1 System Implementation

This chapter covers all the aspects of implementing the system. Basically system implementation is a process of converting all the requirements and designs into program codes. System implementation involved the translation of the software representation produced by the design into computer readable form.

A good programming technique is essential for developers to ensure the quality of software. One of the advantages of good programming technique will make the codes easy to reuse, maintain as well as to modifying purposes in the future, which is very important factor for future software development.

### 6.2 System Coding

In this stage, the programs are written and user interfaces are developed and the database is initialized with data. The components built during development are put into operational use. The system is built according to the original design that was done.

#### 6.2.1 Coding Style

Coding style is an important attribute of source code and it determines the intelligibility of a program. An easy to read source code makes the system easier to maintain and enhance. The element of coding style includes internal (source code level) documentation, method for data declaration and approach to statement construction.



### 6.2.2 Code Documentation

Code documentation begin with the selection of identifier (variable and variable names), continues with the composition of connectivity and end with the organization of the program

**i) Internal documentation.**

Internal comment provides a clear guide during the maintenance phase of the system. Comments provide the development with means of communicating with readers of the source code. Statement of purpose indicating the function of the module and a descriptive comment that is embedded within the body of the source code is needed to describe processing function.

**ii) Naming Convention**

A good and meaningful naming technique for the variables, controls and modules provides easy identification for the programmer. The naming convention is created with coding consistency and standardization in mind.

**iii) Modularity**

Before entering the coding phase, the project has been divided into several modules. The main purpose of modularity is to reduce the complexity of the system. In order to reduce complexity and facilitate changes that result in easier implementation by encouraging parallel development of different parts of a system.

**iv) Readability**

Codes should be easy to understand. Adherence to coding conventions such as naming conventions and indentation contribute to program readability.

**v) Robustness**

The codes should be able to handle cases for user error by responding appropriately. It should be able to avoid any abrupt termination or system failure.

**vi) Maintainability**

Codes should be easily revised or corrected. To facilitate maintenance, code should be readable, modular and as general as possible.

### 6.3 Coding Example

i) coding to insert form data to database

```
<!-- #include file="connect.asp" /-->
<%
studentname= request.form("studentname")
studentid= request.form("studentid")

OpenConnection
cn.execute "Insert into student (studentid, studentname) VALUES ('" & studentid &
",'" & studentname & "'"
CloseConnection

response.redirect "studentadmin.asp"
%>
```

ii) User authentication

User authentication is very important in granting only the person that has privilege can access certain module. Below is coding example in the login page.

```
<!-- #include file="connect.asp" /-->
<%
'on error resume next
dim loginpass,userid,password

userid = request.form("userid")
password = request.form("password")
loginpass=true

if userid<>"" and password<>"" then
    OpenConnection
    qry="SELECT * FROM user WHERE userid='" & userid & "' AND
password='" & password & "'"
    Set rs=server.createobject("ADODB.recordset")
    Set rs=cn.execute(qry)
    if not rs.EOF then
        if userid=rs("userid") and password=rs("password") then

            session("userid") = userid
            session.timeout = 30

        else
            loginpass= false
        end if
    else

```

```

        loginpass= false
    end if
    rs.close
    set rs=Nothing
    CloseConnection
else
    loginpass= false
end if

if loginpass then
    response.redirect "attendance.asp"
else
    response.redirect "default.asp"
end if

'response.write userid & password
%>

```

#### 6.4 Conclusion

In this chapter we have seen how Student Attendance System (SAS) implemented.. In the program coding section I have described how this SAS system is coded and the coding style to make sure the coding more readability when we want to edit for future enhancement. As to ensure the system works well, some of modification of design had been made. Any problem encounter in this phase should be solved before moving to the next phase of system testing.

CHAPTER 7

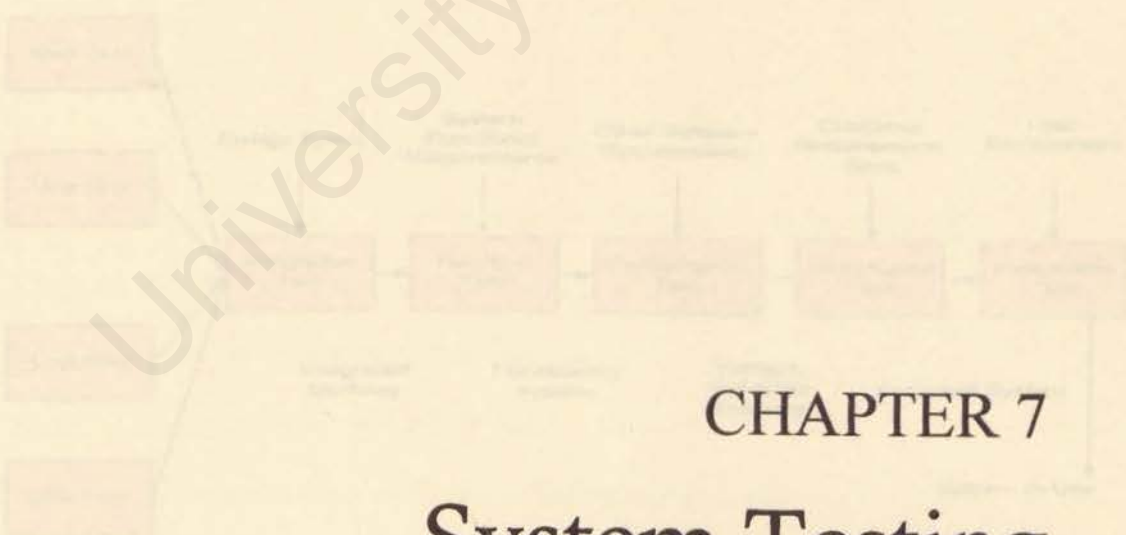
System Testing



plan with the development

## 7.2 Testing

Testing is a process that will determine the errors in a system, and then find the cause of the errors and finally fix the errors. The objective of unit and integration testing was to ensure that the code implemented the design properly, but to the developers write the code whether the desired and intended functionality errors of system failures will occur in systems that we not tested. The testing phase is also crucial as to ensure the quality of software that can make use the system fulfill all the requirements as well as user expectations.



## CHAPTER 7

# System Testing

Figure 7.1: System Testing

# CHAPTER 7: SYSTEM TESTING

## 7.1 Introduction

This chapter covers the testing phase of the development process. Testing methods, analysis of test results as well as problems encountered during the testing phase were also documented.

## 7.2 Testing

Testing is a process that will determine the errors in a system, and then find the cause of the errors and finally fix the errors. The objective of unit and integration testing was to ensure that the code implemented the design properly, that is the developers wrote the code within the desired and intended design. Costly errors of system failures will occur to systems that are not tested properly. The testing phase is also crucial as to ensure the quality of software and to make use the system fulfils all the requirements as well as user expectations.

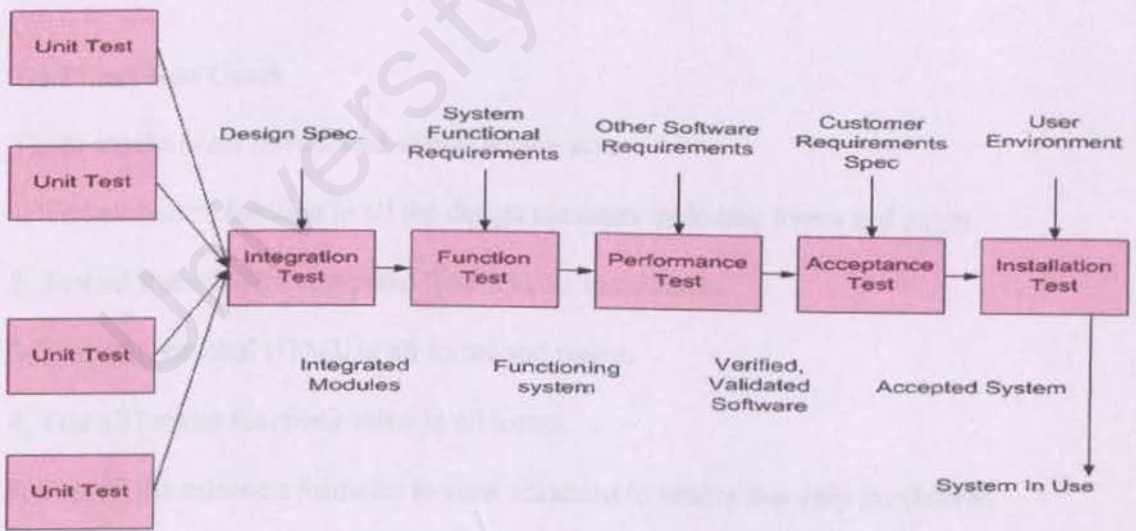


Figure 7-1 Levels of Testing

This system is designed into a few modules and all the modules are integrated to form a system. Three approaches of testing were used in this development process, unit testing, integration testing, and system testing.

### **7.3 Unit Testing**

Unit testing or sometimes known as module testing was performed to detect errors in each and every module in the system. Unit testing begins with the examination of the code by reading through it besides trying to spot algorithm, data and syntax faults. To ensure that all relevant cases have been considered, the code is compared with the specification and design. As a consequence, the remaining syntax faults were eliminated by debugging the codes.

The modules such as Student, Lecturer, Report and Subject were tested separately using this approach. Unit testing involves code walkthroughs, code inspections; both are techniques to examine the codes.

#### **7.3.1 Unit Test Cases**

These are the cases have tested in unit testing areas.

1. Test all button function in all the design elements including forms and pages.
2. Test all formulas for computed field's value in all forms.
3. Test conventional HTML in all forms and pages.
4. Test all lookup functions value in all forms.
5. Test all the selection formulas in view selection to ensure that only the desired types of forms are selected to display in a particulars view.

6. Test all agents to ensure all functioning will be corrected as needed.
7. Test all the association between frames in all framesets.
8. Test all the agents to ensure all functioning correctly.

### 7.3.2 Unit Test Results

After done this testing I found that there are some faults which need corrections. Some links not work properly and some buttons not functioning correctly. At the end of the unit testing, all the functions have been able to execute as desired individually.

University of Malaya



## 7.4 Integration Testing

Integration of the system was done after the all unit testing were conducted. Interaction between modules as well as the calling procedures or functions between each of the modules were the key issues of integration testing.

### 7.4.1 Bottom Up Approach

Bottom-up testing is one of the most popular methods for integration testing. Using this technique, each component or module at the lowest module of the system architecture will be tested first individually. Then, the modules that call all the previous modules will be tested. This procedure will be repeated until all the modules are tested.

One advantage of using the bottom-up testing approach is it allows early testing aimed at proving feasibility of particular modules. Thus, modules can be integrated in various clusters as desired. It has major emphasis on module functionality and performances.

### 7.4.2 Integration Test Cases

The Student Attendance System is tested through form by form, view by view, sub form by sub form, agent by agent and database by database in a sequential manner. The test cases for Student Attendance System consist of the Administrator modules and Lecturer modules as below:

#### Administrator Menu:

1. Test the system login and logout function.
  - Login with authenticated username and password
  - To ensure user will be able to logout the system successfully.

2. Test the space for Student page
  - To create a new entry and view all student which added before
  - To edit or remove the student name and ID number.
3. Test the space for Lecturer page
  - To create a new entry and view all lecturer which added before
  - To edit or remove the lecturer name, ID number and password.
4. Test space for Subject page
  - To create a new entry and view all subject which added before
  - To edit or remove the subject code and name.
5. Test space for Class Assignment page
  - To assign lecturer to particular subject and add student to particular subject.

#### **Lecturer Menu:**

1. Test the system login and logout function.
  - Login with authenticated username and password
  - To ensure user will be able to logout the system successfully.
2. Test space for register attendance page
  - To submit new attendance by subject, date and time.
3. Test space for edit attendance page
  - To edit or delete any previous section attendance
4. Test space for Report page
  - To generate report in various formats (percentage, absentees, multiple graph)

## 7.5 System Testing

The system testing was done after all the modules were integrated and tested thoroughly. The main objectives of system testing are to make sure the system meets customer's expectations and most important, it meets the entire objective of developing the system.

### Function Testing

Function testing is the first step of system testing. It focuses on the functionality of the system, based on the system's functional requirements. These are done to ensure and verify that integrating subsystems hasn't cause error.

Function testing is actually a series of test being carried out to fully exercise parallel in the system. Components and modules of the system were ensured to function properly. Functional requirements such as updating students' attendance, add new subject or new lecturer assignment will be performed and verified.

### Performance Testing

Performance testing is to determine the non-functional requirements of the system as defined in the requirement analysis. Issues such as user-friendliness, security, reliability, and consistency were verified time and time again in the performance testing.

### Sub System Testing

Sub system testing is used to test the sub system. The system testing was done by first browsing the web application without login, then register, login. Records are ensured to have been added or updated into the database.

### Overall System Testing.

After the sub system testing then we do the overall system testing to test overall system functionality. This testing is used to ensure that all the components or modules of the system are functioning properly.

### Acceptance Testing

This is the important part in the system testing phase which is done by user of the system. It commences when the system is ready to use. Users involved in this stage to make sure the system meet their understanding of the requirements, which may be different from the developer. The feedback or comment from the user is important to improve and upgrade the system to fulfill their needs.

From the acceptance testing that involved user, some comments and suggestion have been collected. Every functions of the system either at the back end or front end, is work exactly without any error such as page linking error and so on. The steps of template setup and maintenance is easy to understand and also quite friendly user. Most of the user give the same suggestion that is add in more template design. So that dealer can have more choice for the template design.

## 7.6 Conclusion

In this chapter I have describe the testing methods that used to make sure my system is out of bugs and also fulfill the system requirements. This testing stage has to be done carefully, so that the system output is exactly what we need. Every stage in the system testing is tested by variety of person so that the result more accurate.

University of Malaya

CHAPTER 8

System Evaluation &  
Conclusion



SISTEM KEHADIRAN PELAJAR  
STUDENT ATTENDANCE SYSTEM



Attendance System with its programming logic errors, database errors that was  
incorrect design. There were easily solved while others were solved in due time, but  
some problems are in the logic programming. The system's strengths and limitations  
were evaluated by targeted users. Improvements on the present system and potential  
future enhancements are based on the suggestions and evaluation results from these  
users.

8.2 PROBLEMS ENCOUNTERED AND SOLUTIONS

Throughout the development process, few problems were encountered that  
were eventually resolved. Some of the solutions came easily but there were some that  
required an alternative solution.

Problems in Database and Solution

There was some difficulty in connecting database to web page using  
Microsoft SQL Server especially in Region window. I tried a lot of new things to  
configure the database as well as ASP code but it didn't work. Then to double  
check, I decided to change database to Microsoft Access.

CHAPTER 8

System Evaluation &  
Conclusion

### 8.1 INTRODUCTION

I have experienced many challenges during the development of the Student Attendance System such as programming logic errors, database errors and user interface design. Some were easily solved while others were solved in due time, but most problems are in the logic programming. The system's strengths and limitations were evaluated by targeted users. Improvements on the present system and potential future enhancements are based on the suggestions and evaluation result from these users.

### 8.2 PROBLEMS ENCOUNTERED AND SOLUTIONS

Throughout the development process, a few problems were encountered that were eventually resolved. Some of the solution came easily but there were those that required an alternative solution.

#### Problems in Database and Solution

There was some difficulty in connecting database to web page using Microsoft SQL Server especially in Report module. I spend most of time trying to configure the database as well as ASP coding but it doesn't work. Due to dateline time, I decided to change database to Microsoft Access 2000. With all reference and friends help as well as permission from supervisor, finally the system was connected to database and able to function correctly.

### **Lack of Knowledge in the Language**

Due to the time constraint, the learning and developing process was done in parallel. Without a firm understanding of the programming language, a lot of time was spent in looking for the solutions to problems that occurred while developing this system. These problems arose in cases relating to concepts of the programming language that are new to the author such as parameter passing between pages and response and session objects. The solution is to get reference from books and friends help to understand more about the language.

### **Difficulties in determining the scope of the system**

It is impossible to build full scale complete system with the time frame given. Sometimes I feel to include some extra module while developing this system and some module can be done. I try to follow the modules submitted in proposal earlier and with supervisor assistance I can outline the scope of the project to be built during initial stage.

### **Time Constraint**

This system had to be built in a semester's time. A lot of time was needed to learn new things. The system was built while learning new things. A better system could be produced if a better understanding of the programming language and other things were attained earlier. As such, the system includes the modules which are important to the system and other modules could be added and recommended for future enhancements.



### 8.3 System Strengths

#### User Friendly and Easy to Use

The user interface of this system is very user-friendly and quite consistent from one interface to another in the same component. The flow of the system is very easy to follow and users do not have to go through complex procedure in performing a particular function. All functions can be performed easily by just clicking the button and filling the required information.

#### System transparency

The system is transparent to the user, as they do not need to know where the database resides, how the system is structured, etc. For example, users do not need to know how to retrieve and insert records into the database. All they need to do is submit the data and then view the query results.

#### Security Features

Most of the modules in this system are meant for registered users, thus requiring a user to login, i.e. user identification and password must be keyed in and verified, before any changes can be made. However, any changes have to be approved by the system administrator who also has the authority to edit or delete records in the database. Doing encryption of the data that will be transferred through the network can help enhance security.

#### Consistency

The screen design maintains its consistency throughout the system. Therefore, difficulty in seeking for a particular option that they require will be less experienced by user.

### Comprehensive reports with graphical interface.

Student Attendance System lecturer module also can generate comprehensive reports based on student attendance. There are tables and graph generated dynamically from the store database. This is to ease lecturer (user) to request any reports that make sense for them.

### Maintainability

As some of the functions are stored in the files, the system is easy to maintain. Every object type is written in standard form and the file name of every component also rename in a standard way

## **8.4 System Limitations**

### No change Password function

Changed password function could be provided in the system because of the time constraint. The function in the system in the current system is only available for the administrator.

### No student personal attendance list

User (lecturer) can't view personal attendance list student by student. There is no link to display overall student attendance during semester.

## 8.5 Evaluation By End User

The system also was tested by other users after it was released. Some of the positive feedbacks given are:

1. User friendly interface design is achieved.
2. Error handling and message prompt can easily be understood.
3. Most of the objectives of the system are carried out successfully

But, there are also negative feedback given such as:

1. Lack of 'forgot password' function.
2. User needs to key data manually which takes lot of times.
3. No classification on student department and their details.

## 8.6 Future Enhancement

There are plenty rooms of improvement for future enhancement of the system. Below are some of the features that could be added into the current system:

- ✓ Provide additional reports

The current system only provides three types of reports for attendance analysis. There should be additional function for the user to more comprehensive reports for analysis.

- ✓ Printable Report

Since the system generates various formats of reports, there is no function to print the reports in printable version for record keeping purpose or presentation in meeting. There should be function to print the reports in documentation formats.

- ✓ Support more email services

An email service can be integrated to the system to issues warning or show cause emails to any student who are not fulfill the attendance requirement.

- ✓ More administrative function

Administrator tasks could be enhanced by including more features to ease maintenance process. Added features that could be included are database backup, report generation and multiple students grouping by theirs department.

### **8.7 Knowledge and Experience Gained**

Knowledge gained throughout the development of student attendance system for FSCIT is undoubtedly valuable. Knowledge was gained on the client server development environment, the database system, programming and concepts. The exposure to software tools that include ASP, Ms Access 2000 and Dreamweaver that proves to be beneficial.

Through the development of this system I also gained knowledge how the flow of web based application and also experience how is the job of a programmer. This system really teaches me patient in doing whatever work that I do.

## 8.8 Conclusion

Overall, this Student Attendance System has fulfilled the objectives as in the requirement analysis. Users and university authorities will benefit from using this online system.

This project helped me a lot in learning web-based application development as well as handling a project which requires strict discipline and time management.

As a conclusion, there is still room for improvement for this online system in adding more features to the system as discussed earlier. Hopefully this system will benefit whoever reading my thesis and maybe they can do some enhancement from the weakness of my system.

Reference

## REFERENCE

1. Shari Lawrence Pfleeger, Software Engineering Theory And Practice 2<sup>nd</sup> Edition, Prentice Hall International, INC 2001
2. Chriss Ullman, David Buser, Jon Duckett, Brian Francis, John Kauffman, Juan T.Libre, David Sussman, Beginning ASP 3.0, Wrox Press Ltd, 1999.
3. Robert Vieira, SQL Server 7.0 Programming Professional, Wrox Press Ltd, 1999
4. Gerald V Post, Database Management Systems, McGraw Hill, 1999
5. Ian Sommerville, Software Engineering 6<sup>th</sup> Edition, Addison Wesley, 2001
6. Whitten, Jeffrey L., Systems Analysis and Design Methods, 4<sup>th</sup> Edition, McGraw Hill, 1998
7. Anderson, Richard. Beginning components for ASP. Birmingham: Wrox Press, 1999
8. Visual Basic Forum  
<http://vbworld.net>
9. ASP  
<http://activeserverpages.com>  
<http://learnasp.com>  
<http://www.powerasp.com>  
<http://www.aspfree.com>  
<http://www.webwizguide.info/asp/default.asp>
10. Microsoft  
<http://www.microsoft.com/iis/default.asp>

# USER MANUAL



## SISTEM KEHADIRAN PELAJAR STUDENT ATTENDANCE SYSTEM



The Student Attendance System (SAP) Faculty Computer System and Information Technology is a system to provide facilities to student management. It has functions of producing and using information to meet proper planning in the future.

### CHAPTER 2 GETTING STARTED

#### 2.1 About This Manual

The user manual will guide you through all the functions available in the system including Lecturer Session and Administrator Session.

#### 2.2 System Requirements

Users can access this system by using the address of this site using browsers available. Please make sure you have installed all the minimum hardware and software requirements as stated.

	Requirement
Operating system	Windows 95/98/NT/2000
Web browser	Internet Explorer 3.0 or above
Hardware	Internet Connection

## APPENDIX A

# User Manual

225 MB RAM

Internet Connection

# CHAPTER 1 USER MANUAL

## CHAPTER 1 INTRODUCTION

The Student Attendance System for Faculty Computer Science and Information Technology is a system to provide facilities in student management. It has function of producing and utilizing information to meet proper planning in the future.

## CHAPTER 2 GETTING STARTED

### 2.1 About This Manual

The user manual will guide you through all the function available in the system including Lecturer Session and Administrator Session.

### 2.2 System Requirements

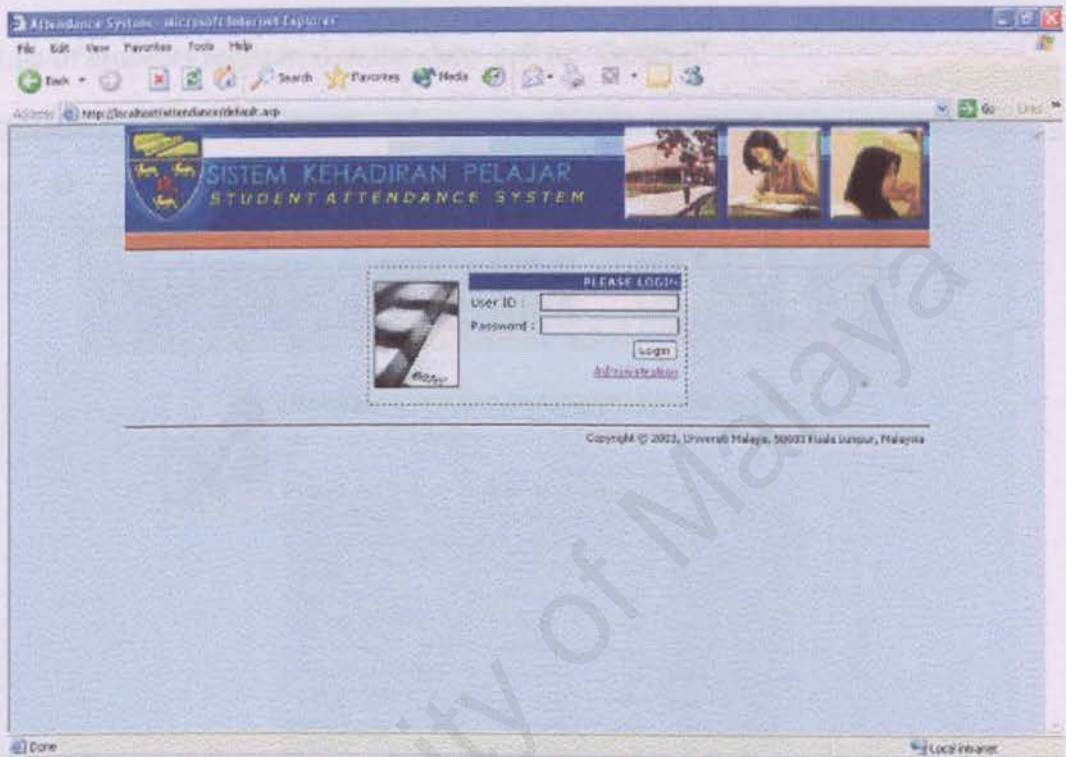
Users can access this system by typing the address of this site using browsers available. Please make sure your system meets all the minimum hardware and software requirements as stated below:

Category	Requirement
Operating system	Windows 98/NT/2000/ME/XP
Web browser	Internet Explorer 5.0 or above
Hardware	Pentium II or above 128 MB RAM Internet Connection



## CHAPTER 3 LECTURER SESSION

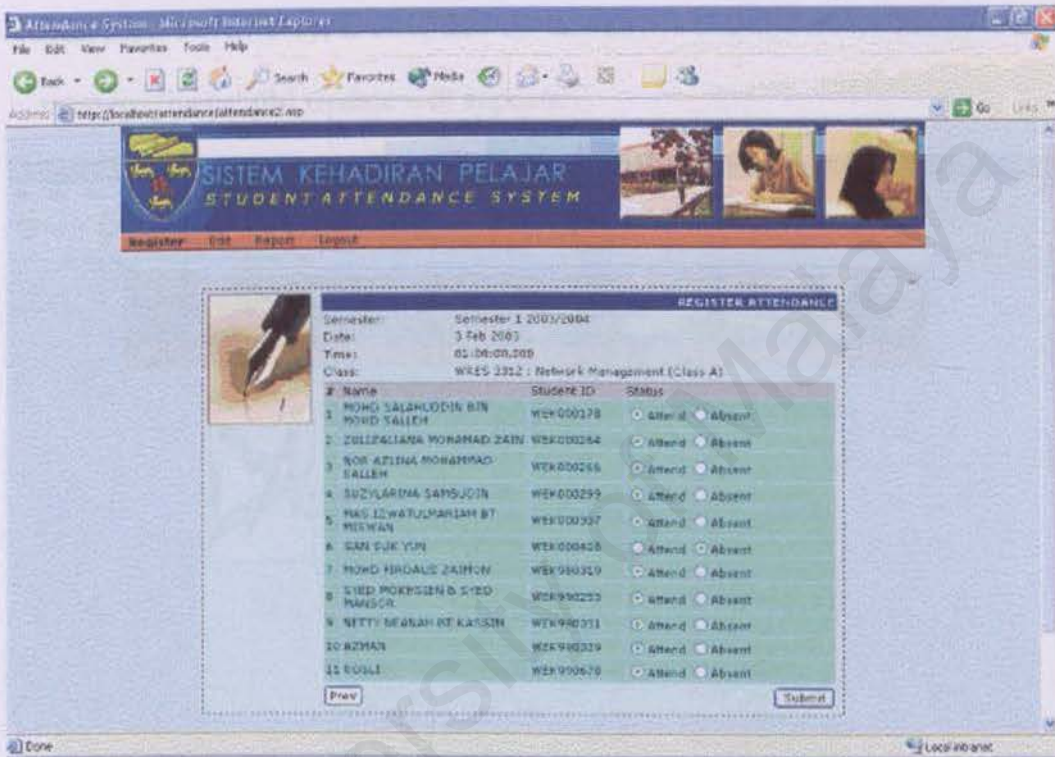
1. Type in the URL for STUDENT ATTENDANCE SYSTEM and the browser will then display the main page of the system



2. Login with the correct username and password
3. User will be brought to the main lecturer's menu of the system.

### 3.1 Register Session

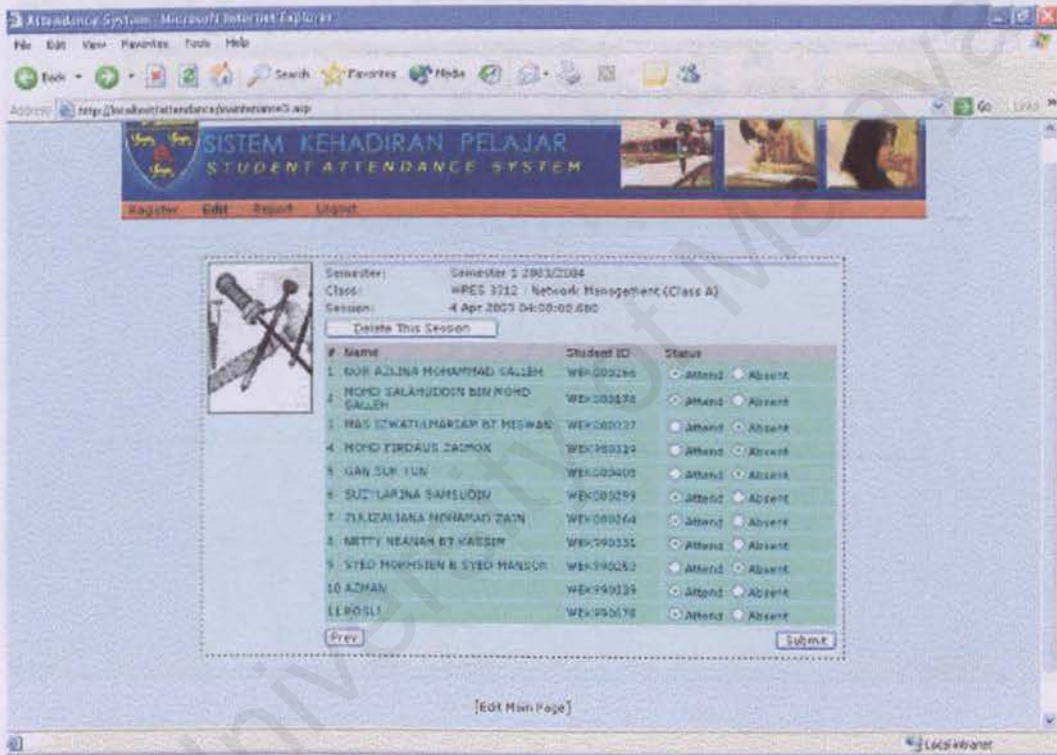
- i) On the main page of register session, the information for particular user will display, user needs select from the list data for date, time and class, then click NEXT button
- ii) List of students registered in the class will be displayed



- iii) All the students are default to Attend button, so user only need to click on *absent* button for particular student, then click SUBMIT button
- iv) Message Successful Update will display.

### 3.2 Edit Session

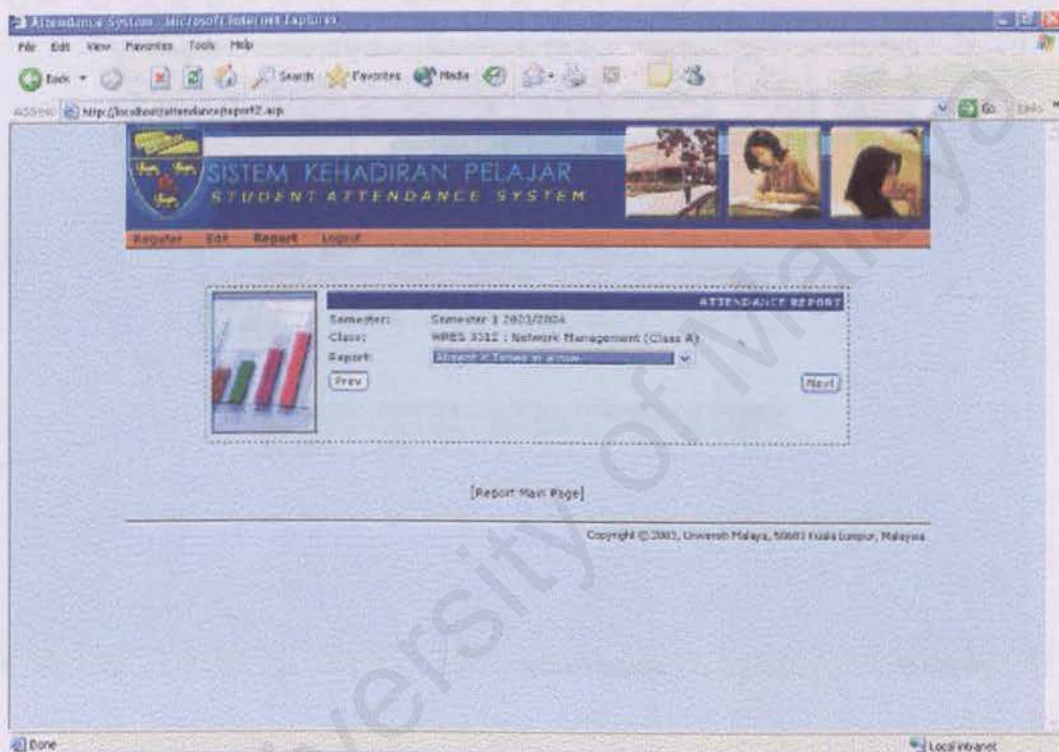
- i) Click on *edit* link on the menu list
- ii) Select semester and click NEXT
- iii) Select subject and click NEXT
- iv) Select session and click NEXT
- v) Edit page will display, after update the attendance click SUBMIT button and message Successful Update will appears.



- iv) User also can delete the section if the want to by click on DELETE THIS SECTION button.

### 3.3 Report Section

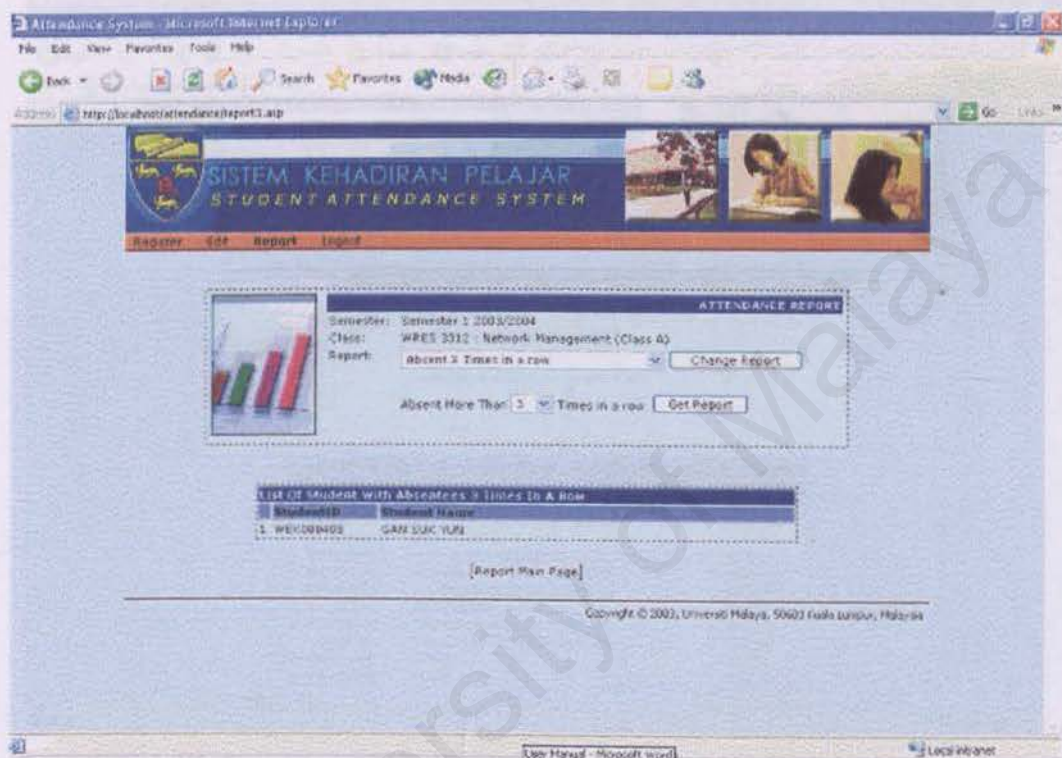
- i) Click on *report* link on the menu list
- ii) select semester and click NEXT
- iii) select subject and click NEXT
- iv) select reports type and click NEXT
- v) page for reports format is display



### 3.3.1 Attendance Percentage

#### 3.3.1 Absent in A Row

- i) on the absent in a row page, select from the list for how many times student absent at the current time.
- ii) click GET REPORT button and list of student with absentees will appear.



### 3.3.2 Attendance Percentage

- i) on the attendance percentage page, select from the list percentage of attendance for all student
- ii) click GET REPORT button and student list will appear

The screenshot shows a web browser window displaying the 'SISTEM KEHADIRAN PELAJAR' (Student Attendance System). The page is titled 'ATTENDANCE REPORT' and shows the following details:

- Semester: Semester 1 2003/2004
- Class: WRES 3011 - Nebozi Management (Class A)
- Report: Attendance Percentage Below X% (with a 'Change Report' button)
- Attendance Percentage: All Attendance % (with a 'Get Report' button)

Below the report controls is a 3D bar chart showing attendance percentages for 11 students. Below the chart is a table titled 'Student Attendance' with the following data:

Student ID	Student Name	Attendance
1. WEK000178	MOHD SALAHUDDIN BIN MOHD SALLEH	80.00%
2. WEK000397	HAS ZIKRATULHARIS BT HESWAN	50.00%
3. WEK900314	MOHD FIRDAUS ZAIMON	37.50%
4. WEK000299	SUZYLARINA SAMSUDIN	50.00%
5. WEK900253	SYED MOHSIN B SYED MANSOR	50.00%
6. WEK000264	SULIZALIANA HOSAMAD ZAIN	66.67%
7. WEK990331	NETTY NEANAH BT KASSIM	37.50%
8. WEK000408	GAN SUK YUN	31.50%
9. WEK000266	NOR AZLINA HOSAMAD SALLEH	50.00%
10. WEK990339	AZMAN	25.00%
11. WEK990478	ROSLI	31.50%

- iii) to view graph percentage of class attendance, click on report list and select [Graph] Percentage of Class Attendance, then click Change Report
- iv) the graph will display and user can switch either in normal type of graph or 3D graph

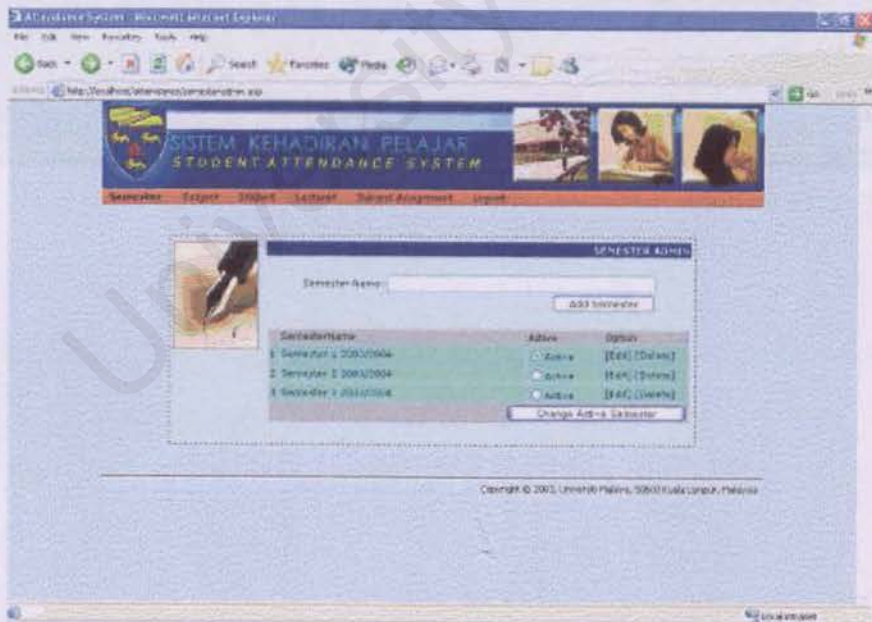
## CHAPTER 4 ADMINISTRATION SESSION

### 4.1 Login

- i) On the main page of Student Attendance System, click *administrator* link to login as administrator.
- ii) Login with the correct username and password
- iii) User will be brought to the administrator's menu of the system

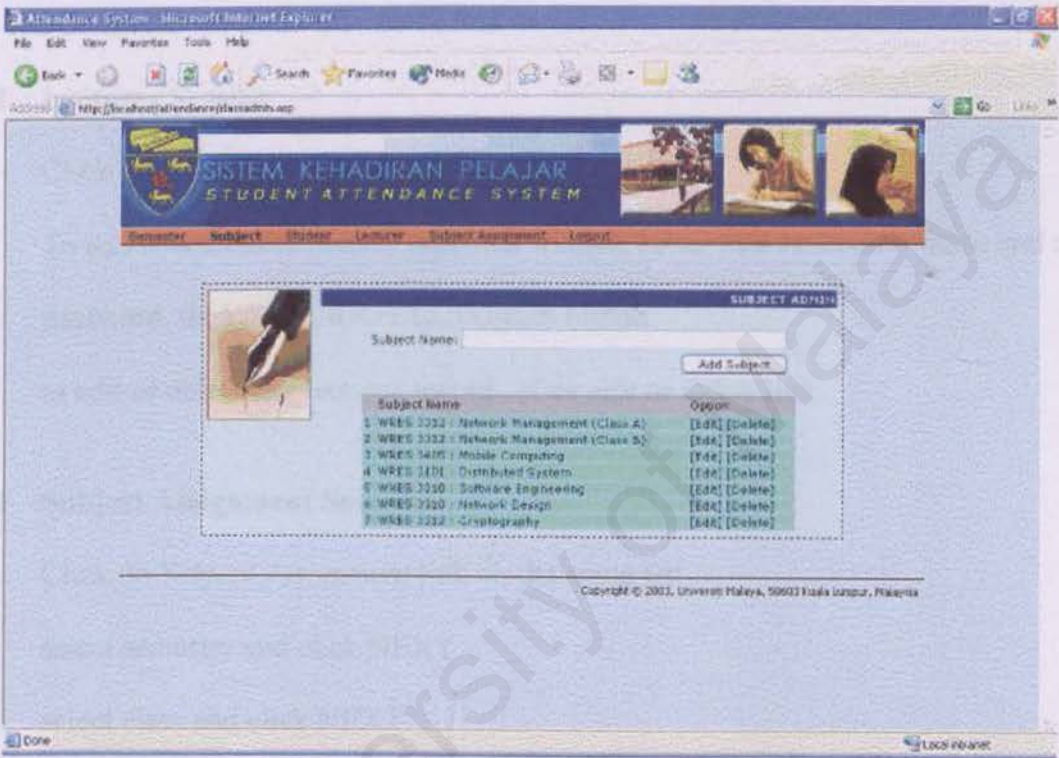
### 4.2 Semester Session

- i) To add new attendance record, type new semester name and click ADD NEW SEMESTER button.
- ii) To change active semester, user needs to click *active* button for semester desired and click CHANGE ACTIVE SEMESTER button.
- iii) to edit or delete any semester desired, click *edit* or *delete* link



### 4.3 Subject Session

- i) Click on *subject* link on the menu list
- ii) To add new subject record, type new subject name and click ADD SUBJECT button.
- iii) to edit or delete any subject desired, click *edit* or *delete* link





#### 4.5.1 Student Session

- i) Click on *student link* on the menu list
- ii) To add new student record, type new student ID as well as student name and click ADD STUDENT button.
- iii) to edit or delete any student desired, click *edit* or *delete* link

#### 4.5.2 Lecturer Session

- i) Click on *lecturer link* on the menu list
- ii) To add new lecturer record, type new lecturer ID as well as lecturer name and the password, then click ADD LECTURER button.
- iii) to edit or delete any lecturer record , click *edit* or *delete* link

#### 4.5.3 Subject Assignment Session

- i) Click on *Subject Assignment* link on the menu list
- ii) select semester and click NEXT
- iii) select class and click NEXT
- iv) to assign any lecturer to selected class, choose lecturer from the list and click UPDATE LECTURER button.
- v) to add student into selected class, choose student name from the list and click ADD STUDENT button.

vi) list for student registered in selected class will appear

SISTEM KEHADIRAN PELAJAR  
STUDENT ATTENDANCE SYSTEM

Semester Class Student Lecturer Subject Assignment Logout

SUBJECT ASSIGNMENT

Semester: Semester 1 2003/2004  
Class: WPES 3312 Network Management (Class 4)  
Lecturer: Fadziah Osman

Student: MOHD GAFFULBZAM

Student ID	Student Name	Option
1 WEP00170	MOHD SALAHUDDIN BIN MOHD SALLEH	[Remove]
2 WEP00171	MAS LIZWATI FARHAM BT NISWAN	[Remove]
3 WEP00172	MOHD FIRDAUS FAIRUDS	[Remove]
4 WEP00173	SITI ALERBA CALEICH	[Remove]
5 WEP00174	SHEE MOVRISTEN B SHEE HANSON	[Remove]
6 WEP00175	DULIZALINA MOHAMMAD ZAIN	[Remove]
7 WEP00176	NETTY BEANAH BT KASSIM	[Remove]
8 WEP00177	SAN LIA YUN	[Remove]
9 WEP00178	NOR AZLINA MOHAMMAD SALLEH	[Remove]
10 WEP00179	ATHAN	[Remove]
11 WEP00180	ROSLI	[Remove]

[Edit Main Page]

Copyright © 2003, Universiti Malaysia, 50003 Kuala Lumpur, Malaysia

Logout

vii) to remove student from the class, click *remove* link for any student desired.