Faculty of Computer Science and Information Technology
University of Malaya

WXES 3182 : Thesis Project

Electronic Record Book Management System
(E-Record Book)

By
WONG KOK CHUNG (WEK 000032)

Under the Supervision of
Assoc Prof Dr Lee Sai Peck

Moderator
Assoc Prof Dr Ow Siew Hock

A graduation exercise submitted to the
Faculty of Computer Science and Information Technology
University of Malaya

As partial fulfillment of the requirement for the
Degree of Bachelor of Computer Science
Session 2002/2003

Abstract

Electronic Record Book Management System (E-Record Book) is a client-server application that provides the headmaster and teachers with a more efficient and secure environment in managing and handling the electronic version of record books maintained by teachers. It is developed to support all operations on the manual record books used in schools nowadays.

The objectives of E-Record Book are:

- To provide teachers with ease of recording students' particulars, lesson plans and other related matters via an easy-to-use Web interface.
- To provide the headmaster with a better monitoring and validation system on teachers' record books.
- To create a better management environment by providing the headmaster and teachers with an online medium for effective communication and discussion.
- To eliminate the unnecessary procedures in updating manual record books, and thus, reducing teachers' workload.
- To enable teachers to update their own record book after working hours.
- To provide a more secure way for the school administrator or school clerk to handle information on teachers and students.

To provide such a good teaching and learning practices in schools, E-Record Book has been developed to provide 3 main sections: Administrator Section, Headmaster Section and Teachers Section.

The Administrator Section allows the school administrator or clerks to create user accounts for teachers and the headmaster, informing any updates of events in the school and creating students' profiles and examination records.

The Headmaster Section allows the headmaster to monitor the teachers' progress in teaching and to validate their lesson plans. Besides, the headmaster will also be able to post any comments on any written syllabus plan and/or lesson plans. In addition, the headmaster has the privilege to browse through any information on the teachers' profiles, timetable, syllabus plans, lesson plans and students' records.

The Teacher Section allows teachers to manage their own E-Record Book as like the manual record book. They will be able to prepare their own timetable, syllabus plans, lesson plans, students' details and grades. Moreover, they will have the privilege to add or update the existing records and to check whether the headmaster has verified their record book.

E-Record Book has been developed to help the headmaster and teachers to interact and deal with record books management more efficiently and effectively. This is also a way to promote a paperless environment, thus revolutionizing the way the record books are currently managed.

Acknowledgement

First and foremost, I would like to take this opportunity to express my gratitude and heartfelt appreciation to my supervisor, Assoc. Prof. Dr. Lee Sai Peck. Without her support, guidance and advice, this project will not be a reality.

Next, I would like to thank my moderator, Assoc. Prof. Dr. Ow Siew Hock for being most considerate and kind.

I would also like to extend my appreciation to Mrs. Wong Sook Fun who is a teacher in Sekolah Menengah Kebangsaan Sri Rahmat, Johor Baharu for her cooperation throughout the interview and information given regarding the record book in the school. Many thanks to the administrative staffs in the school for their cooperation in providing the necessary information on the school's record book management system.

Finally, I would like to thank to my parents and friends who have been very kind, helpful and have given me with valuable advice and moral support during the development of the E-Record Book project.

List of Contents

List of Tables

List of Figures

Chapter 1 - Introduction

1.1 Overviews

Teaching and learning process become more challenging nowadays to achieve the targets in Malaysian Smart School application. The Malaysian Smart School application is the government's initiative to enhance learning institutions through a 'Teaching-Learning' process of curriculum, pedagogy, assessment and teaching-learning materials. It will provide better teaching-learning practices and school management in order to prepare children for the Information Age and to achieve the goals of the National Educational Philosophy. **(www.mdc.com.my)**

In order to achieve that target, the school has to play a major role especially the teachers who have to prepare themselves in the era of new technology. This also includes the manual record management system which is currently used by teachers in schools.

Thus, technology should be adopted to help school headmaster and teachers to provide a more efficient record management system in school. That's why Electronic Record Book Management System (E-Record Book) is needed. E-Record Book is a web-based system that helps both school headmaster and teachers to interact and deal with the teaching and learning process more effectively.

E-Record Book enables teachers to create lesson plans for various classroom based on the teaching subjects, update the students' records and their grades, and manage personal time-table and classroom time-table. Thus, teachers can prepare the record book for daily lesson's planning faster and effective. They only need to access

the Web to update the necessary information for electronic record book.

The main purpose of the manual record book is to provide the ease for the school headmaster to perform checking and monitoring on the teachers' activities and the experience or progress in every classroom within a short period. With this new E-Record Book, the headmaster does not have to collect all the record books from teachers by the end the weekdays to check and validate them. Due to the instance access through the Web, he or she can directly monitor the teacher's progress of the day with certain clicks and gives comments.

With E-Record Book, it can increase effectiveness and easiness in the teaching and learning process. It would certainly give space for the schools in Malaysia to increase the education quality to produce teachers with better quality.

1.2 Project Motivation

Producing high quality teachers are important for the school institutions growth in terms of management. E-Record Book will make teachers to conduct the teaching and learning process more effective and comfortable to achieve the objective of this system.

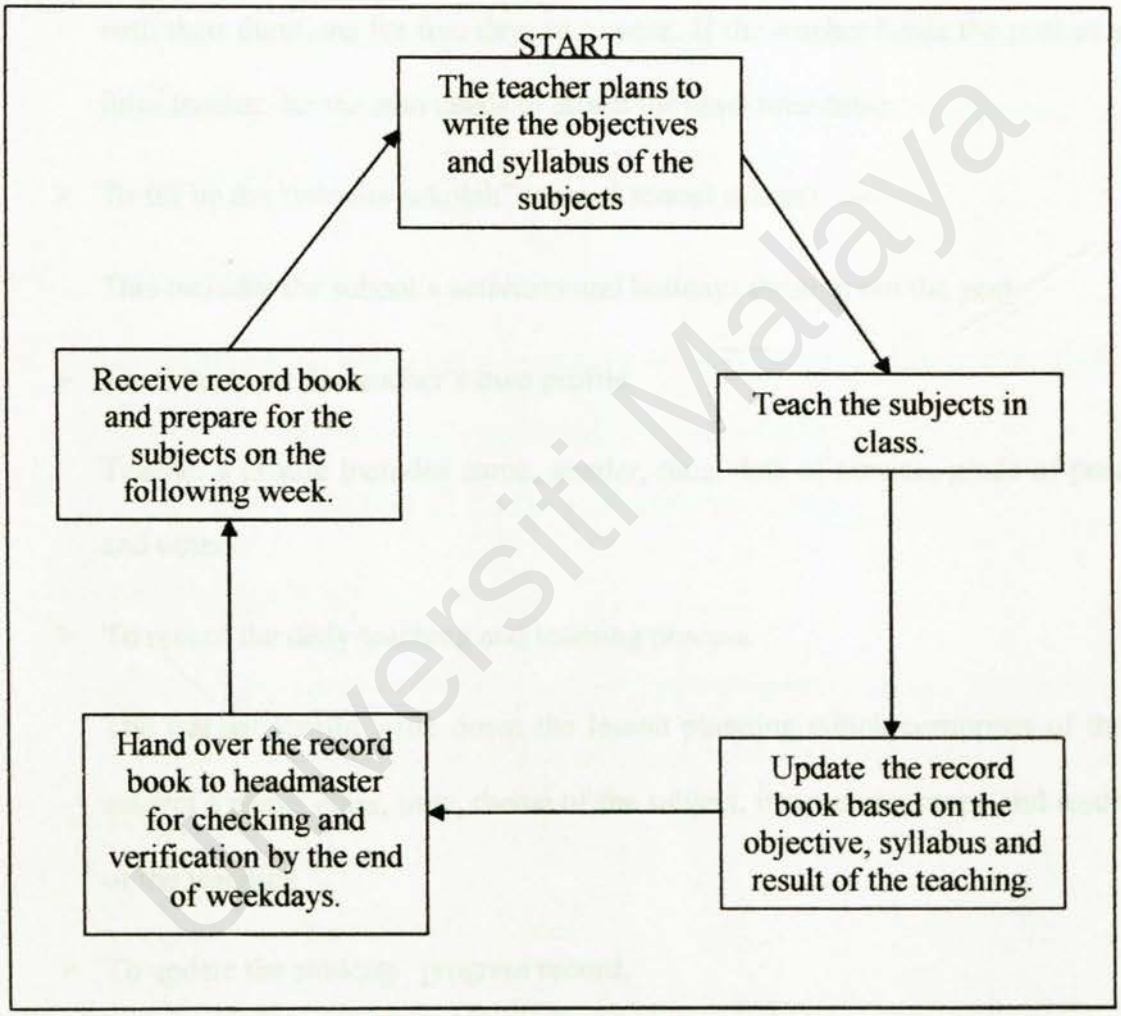


Figure 1.1: Usual stages for teacher to update a record book

1.2.1 Current Practices Performed In Manual Record Book

Below are the practices and their descriptions to be performed by the teacher in the record book :

- To fill up the personal time-table and classroom time-table.

  The teacher should write down the name of subjects taught in certain classes with their durations for five days in a week. If the teacher holds the post as a form teacher, he/she also needs to attach the class time-table.

- To fill up the "takwim sekolah" (annual school events)

  This includes the school's activities and holidays through out the year.

- To write down the teacher's own profile.

  Teacher's profile includes name, gender, race, date of service, grade of post and others.

- To record the daily teaching and learning process.

  The teacher should write down the lesson planning which comprises of the subject's name, class, time, theme of the subject, its teaching scope and result of the teaching.

- To update the students' progress record.

  The teacher needs to record the grades for the students for the subjects taught based on their monthly tests, mid-semester and final semester examinations.

1.2.2 Problems with Current Manual Record Book

As we can see through figure 1-1, the current tasks performed by teachers in the record books consist of weaknesses. Some major weaknesses are as below:

- Time-consuming and inefficient

  Updating the record book manually needs a lot of time and preparations. Teacher cannot use the record book once it has been handed out to the headmaster for validation.

- Wastage of resources

  Printing the record books which is bulky and hard covered would involve higher cost then if it were prepared electronically. It involves also wastage of spaces and human resource to collect all the record books from teachers.

- Ineffective communication

  Any inquiries regarding the record book from headmaster can only be done in school days. It is because normally teachers are hard to be reached when they are after work.

1.2.3 Solution for the Current Situation

An effective online system is needed to solve the current problems faced in managing the record book in school. The system should be able to bring the headmaster and teachers closer to each other, make it easier to monitor the teacher's progress in teaching, students' achievement in their academic studies and eliminate the usual stages

to update the record book and hand out it for validation.

Here is where the Electronic Record Book Management System (E-Record Book) plays these roles. It can shorten the distance between headmaster and the school teachers. E-Record Book can make the teachers' jobs easier and increase the effectiveness in the teaching and learning process.

1.3 Project Objectives

Core objectives of the project are as below:

- To enable the teachers a better record management system with the ease of recording books in the Web.
- To provide the headmaster a better monitoring and validation system on the teachers' record books.
- To enable the headmaster and teachers to communicate well to create a better management environment.
- To eliminate some unnecessary procedures in updating the record book and reduce teachers workload.
- To enable the teachers to update the record book after working hours.

1.4 Project Scope

Generally, E-Record Book can be divided into three major parts, which are the **Administrator Section**, **Headmaster Section** and **Teachers Section**. The School Administrator Section is to manage the system including give approval to headmaster and teachers to use the system. Headmaster Section allows school headmaster to monitor, check the teachers' progress in teaching and validate their record books. The Teachers Section allows teachers to manage their own record book such as prepare their lesson plan and update the students' grades in the electronic record book in Web environment.

The project scope of E-Record Book for **Administrator Section** includes:

- Develop a web-based management system to manage the headmaster and teachers who are using the system.
- Develop a web site to view the relevant information in the electronic record book such as Piagam Sekolah, School Objectives and School Annual Events.

The project scope of E-Record Book for **Headmaster Section** includes:

- Develop a web-based monitoring system to browse through the teachers' record books.
- Develop a web-based checking system to validate the electronic record books and make comments on the book.

The project scope of E-Record Book for **Teachers Section** includes:

- Develop a web site to enable teachers to login differently into system.
- Develop a web-based application for teachers to update their time-table, own profile, syllabus of subjects, daily lesson plan, students records and grades in their electronic record books.
- Develop a web-based to enable teachers to submit their record book to headmaster for validation by the end of the weekday.

1.5 Expected Outcome

Expected outcomes of E-Record Book for **Administrator Section** include:

- A control that can create user login and password for headmaster and teachers and remove user account.

Expected outcomes of E-Record Book for **Headmaster Section** include:

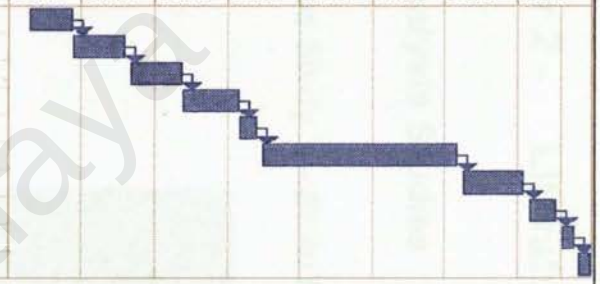
- Monitoring management panel – to view every sections in electronic book for every teacher in school.
- Validation panel – to validate the electronic record book after checking.
- Comment and inquiries panel – to post comments or inquiries to teachers regarding their record books.

Expected outcomes of E-Record Book for **Teachers Section** include:

- Web information about the School Objectives, School Annual Events, Organization Chart and Teacher's Roles in school.
- Teacher's profile – to input own profile.
- Time-table management panel – to update personal time-table or classroom time-table.
- Subject's syllabus – to update the syllabus for every subject taught by teachers.
- Daily subject's planning – to input time, date, name of subject, class, objective, learning scope and result of learning for each subject.
- Student's record and grade – to input student's personal information, grades and marks for the subjects taken in class.

1.6 Project Schedule for E-Record Book

ID	Task Name	Duration	Start	Finish	Predecessors
1	Preliminary Study and Planning	14 days	Mon 06/10/02	Thu 06/27/02	
2	Literature Review	16 days	Fri 06/28/02	Fri 07/19/02	1
3	System Analysis	16 days	Mon 07/22/02	Mon 08/12/02	2
4	System Design	18 days	Tue 08/13/02	Thu 09/05/02	3
5	Prototype	6 days	Fri 09/06/02	Fri 09/13/02	4
6	Development and Coding	60 days	Mon 09/16/02	Fri 12/06/02	5
7	Unit Testing	20 days	Mon 12/09/02	Fri 01/03/03	6
8	System Testing	10 days	Mon 01/06/03	Fri 01/17/03	7
9	Documentation	6 days	Mon 01/20/03	Fri 01/24/03	8
10	Implementation and Maintenance	6 days	Mon 01/27/03	Fri 01/31/03	9



Project: E-Record Book

Task
Split
Progress
Milestone
Summary
Project Summary
External Tasks
External Milestone
Deadline

Figure 1.2 : Project Schedule

Chapter 2 - Literature Review

2.1 Analysis Studies

2.1.1 Case Study 1 – *Buku Rekod Mengajar Guru Sekolah Menengah*

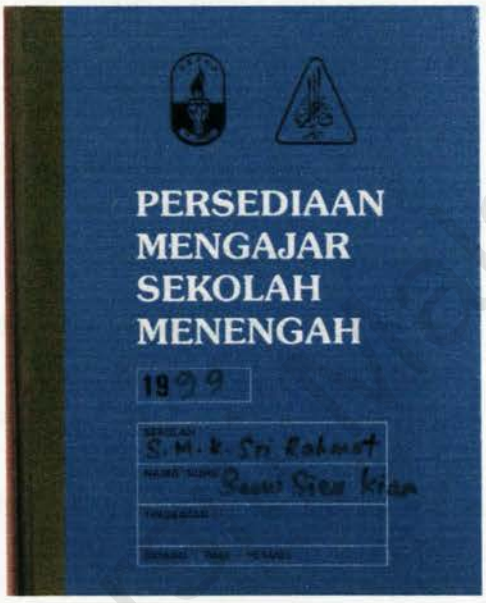


Figure 2.1: The main cover of Buku Rekod Mengajar Guru Sekolah Menengah

This is the official record book for secondary teachers of Sekolah Menengah Kebangsaan Sri Rahmat in Johor Bahru. It is very important for the teachers who need to make comments on the events during their daily teaching operations. Teachers can make a record of their students' progresses in every subject. The book can be referred once the other teacher has replaced the current teacher on a certain subject. Headmaster of the school can used it as a guide to evaluate the experience and progress of their teachers in a short time. **(Pekeliling Kementerian Pendidikan 256 Buku Rekod Untuk Guru, 1956)**

Some of the contents of the book are as below:

- The **Takwim Sekolah** section
  - The start date and finish date for school days, semester holidays for every Semester 1 and Semester 2 are listed.
  - The date for the public holidays and school events are also listed in the book.
- The **Teacher Profile** section
  - The teacher has to fill in their profile which include name, gender, religion, IC Number, qualifications, etc.
- The **Time-table** section
  - Class time table can be used by the form teacher as a reference and analyze of every period of the subjects is viewed.
  - Personal time table also available for teacher who teaches different subjects for certain classes in a week.
- The **Lesson Plan** section
  - Teacher can record the time and name of the teaching subject for certain class and write down his or her lesson plan based on the subject.
  - Lesson plan is divided into three parts which are the title of the chapter, its objectives and contents.

- The **Student Examination Record** section
    - Marks and grades for every students who have taken the subject will be recorded for monthly tests, mid-term and final examination.

2.1.1.1 Result of Study

Strength:

- Easy to retrieve information manually from the book.
- Able to monitor teachers' progress and performance by the headmaster.
- Able to track down students' performance by the teachers and headmaster.

Weaknesses:

- Low protection information – normally the record books are kept in the staff room and they are easily accessed by public.
- Need to hand out to headmaster for checking and validation – teachers have to hand out their record books to headmaster by the end of the weekdays.
- Waste of resources – lots of paper is used annually to manufacture the books and the cost of manufacturing the books are money consumption.
- Bulky and heavy – the books are quite heavy and inconvenient for the teachers to bring along with them.

2.1.2 Case Study 2 – *ThinkWave Educator*

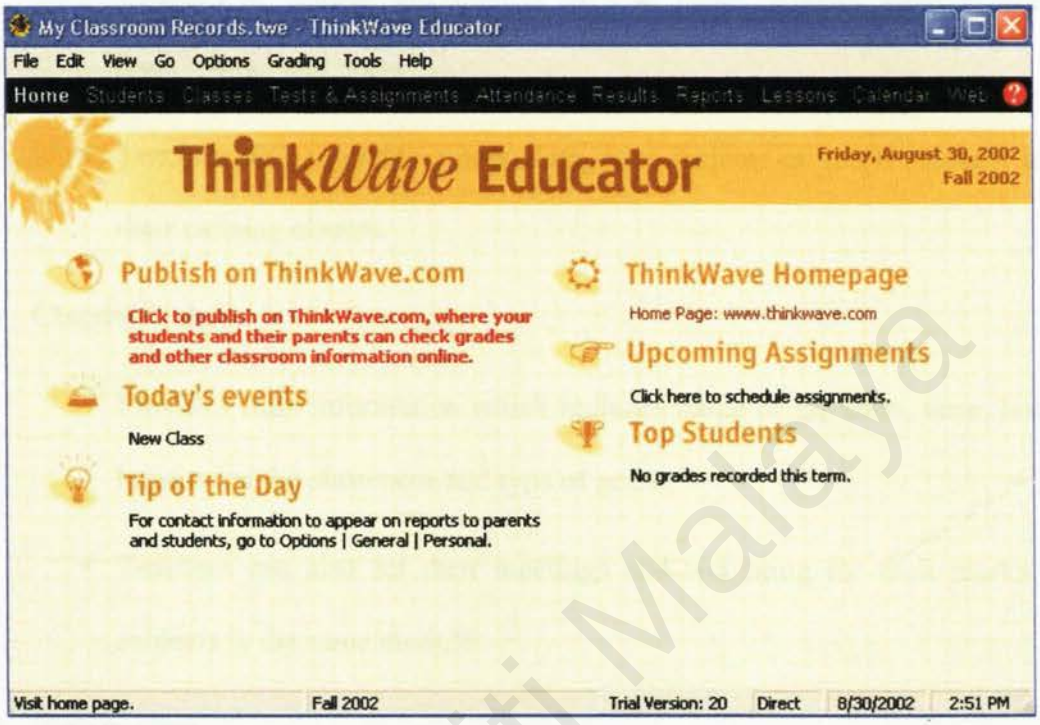


Figure 2.2: The main page of ThinkWave Educator application

This (as shown in figure 2-2) is a stand-alone Windows application designed by ThinkWave company. The application can help teachers to record all the necessary items in the schools includes records of students, classes, test and assignments, attendance, results, reports, lessons, calendar and web. It also has the ability to publish the students' grades and other classroom information online on ThinkWave.com so that their students and parents can check their grades. **(www.thinkwave.com)**

Some of the modules in the application are:

- **Main page** module
  - Consists of links to access the modules in the application.

- **Students** module
  - Consists of student information such as name, student number, grade level, e-mail and notes.
  - Teachers able to create a new class, new students or enroll students into their existing classes.
- **Classes** module
  - Provides class information which includes name of the class, term, level, location of the classroom and type of grade.
  - Teachers can also set their meetings and weighting for their marks of subjects in the same module.
- **Web** module
  - By publishing grade book to ThinkWave.com, the students and their parents can check their own individual grades, attendance and other classroom information online.
  - The teacher will need to enter his or her publishing key from ThinkWave in order to publish successfully the information online.

2.1.2.1 Result of Study

Strength:

- Manageability – the application provides most of the record management tools.
- Publishing online – able to publish on ThinkWave.com for students and parents

to check their grades.

- Universal accessibility – people who have access key in the Internet can able view the data in ThinkWave.com.

Weaknesses:

- Limited space – the application provides limited workspace for the teachers to perform their daily tasks and have to scroll to the right of the page to update their records.
- High dependency – users of this application have to depend on the ThinkWave server to publish their grade book and class information.
- Time-consuming – the process of learning up how to use this application may take time for the end-users.

2.2 Software Architecture

A system's architecture describes the system in terms of a set of architectural units, and a map of how the units relate to one another. The more independent the units, the more modular the architecture and the more easily user can design and develop the pieces separately. **(Wasserman, 1996)** There are a few software architectures available now: mainframe architecture, client-server architecture, two-tier architecture and three-tier architecture.

2.2.1 Mainframe Architecture

In mainframe system architecture, all operation is within the central host computer. User interacts with the host through a terminal that captures keystroke and sends that information to the host. Mainframe architecture is not tied to a hardware platform. User interaction can be cloned using PCs and UNIX workstations. A limitation of mainframe architecture is that it does not easily supports graphical user interface or accesses to multiple databases from graphically dispersed sites.

2.2.2 Client-Server Architecture

2.2.2.1 Client-server

Client-server is network architecture in which each computer or process on the network is either a client or a server. Client-server architecture implies a cooperative processing of requests submitted by a client, or requester, to the server, which processes the requests and returns the results to the client. The client manipulates the data and

presents the result to the user. **(Dixon, 1996)**

2.2.2.2 Client

Client is a networked information requester, usually a PC or workstation, that can query database and/or other information from a server. Clients rely on servers for resources, such as files, devices, and even processing power.

2.2.2.3 Server

Server is a computer, usually a high-powered workstation, a minicomputer, or a mainframe, that houses information for manipulation by networked clients. Server is dedicated to managing disk drives (file servers), database (database servers), printers (print servers), or network traffic (network servers).

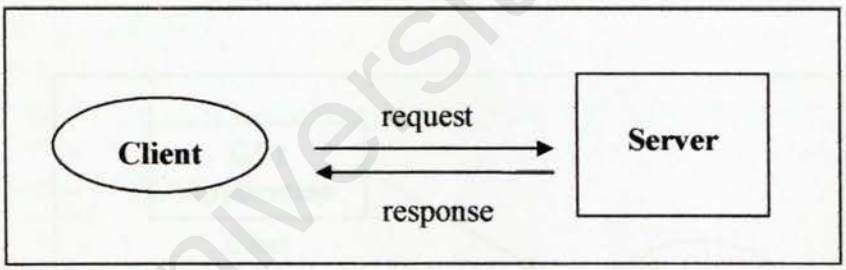


Figure 2.3: One-to-One Client Server

Client-server solutions can be in a many-to-one design that is more than one client typically makes requests of the server.

2.2.3 Two-Tier Architecture

Two-tier architecture refers to client/server architectures in which the user interface runs on the client and the database is stored on the server. The actual

application logic can run on either the client or the server. There are only the architecturally tiered data server and client.

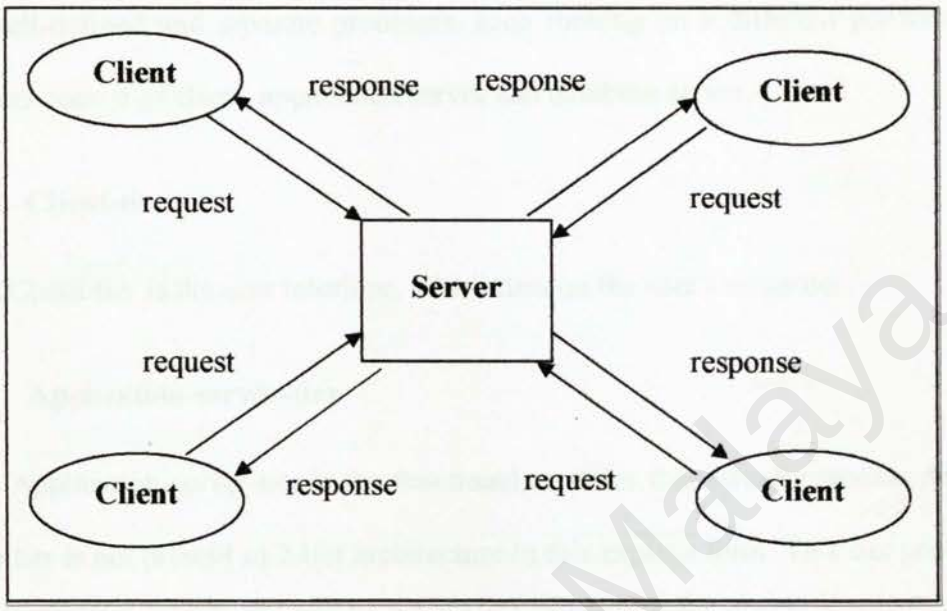


Figure 2.4: Many-to-One Client Server

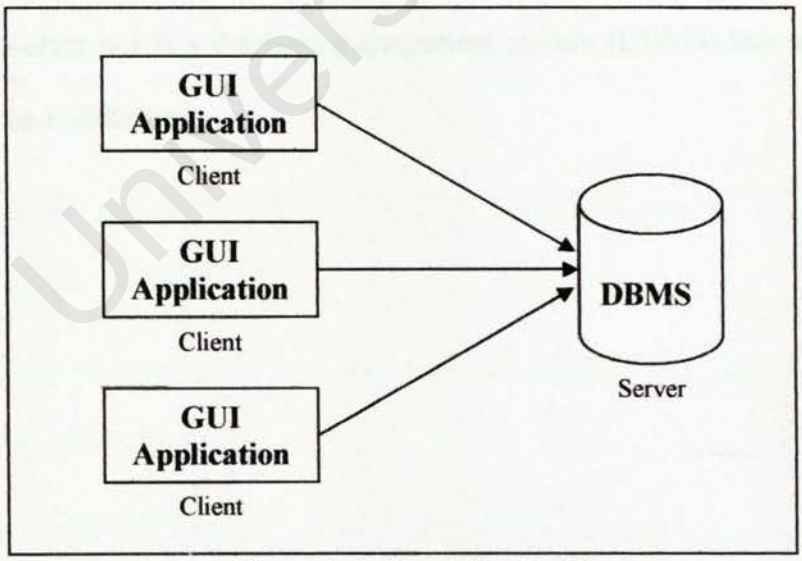


Figure 2.5: Two-Tier Architecture

2.2.4 Three-Tier Architecture

Three-tier architecture is a special type of client/server architecture consisting of three well-defined and separate processes, each running on a different platform: The three tiers consist of client, application server and database server.

2.2.4.1 Client-tier

Client-tier is the user interface, which runs on the user's computer.

2.2.4.2 Application-server-tier

Application-server-tier is the functional modules that actually process data. This middle tier is not present in 2-tier architecture in this explicit form. This tier protects the data from direct access by the clients.

2.2.4.3 Data-server-tier

Data-server-tier is a database management system (DBMS) that stores the data required by the middle tier.

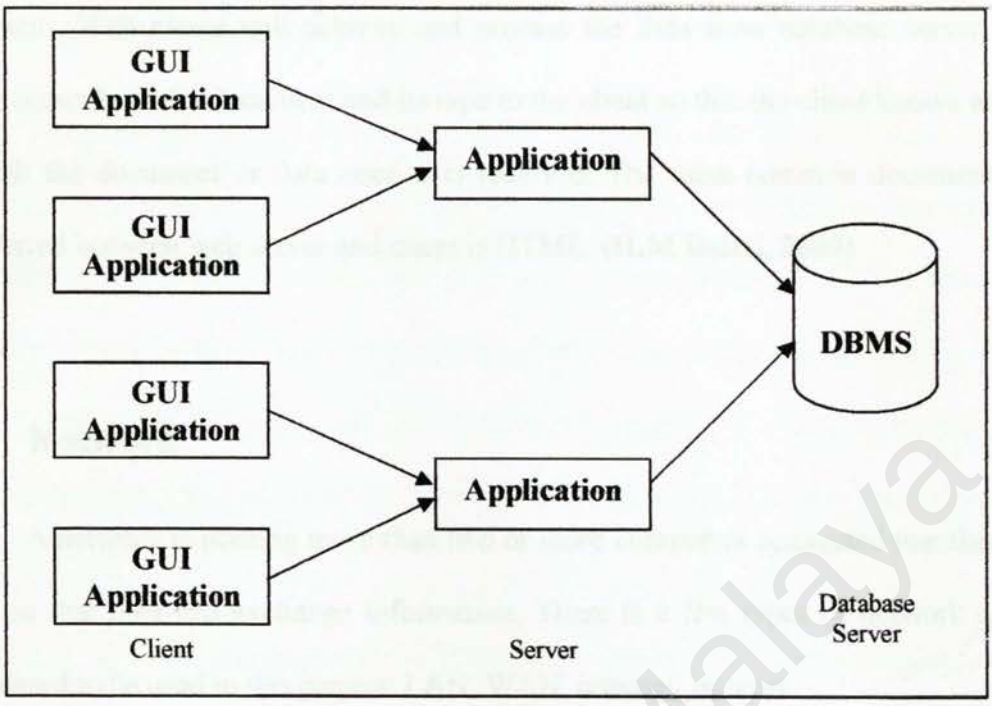


Figure 2.6: Three-Tier Architecture

2.2.5 Conclusion for Software Architecture

The three-tier architecture is chosen for this project because it is easier to implement and design. The three-tier design has many advantages over traditional two-tier or single-tier designs, because:

- The added modularity makes it easier to modify or replace one tier without affecting the other tiers.
- Separating the application functions from the database functions makes it easier to implement load balancing.

In this project, the three tiers consists web browser as client-tier, web server as application-server-tier and database server as the data-server-tier. A web server can be defined as a computer program that receives HTTP requests from web browser for

document. Web server will achieve and process the data from database server. Web server return both the document and its type to the client so that the client knows what to do with the document or data once it is received. The most common document type transferred between web server and client is HTML. **(H.M Deitel, 2000)**

2.3 Network

A network is nothing more than two or more computers connected together by a cable so that they can exchange information. There is a few types of network can be considered to be used in this project: LAN, WAN, internet, intranet

2.3.1 Local-Area Network (LAN)

A LAN is a connection between two or more computers, which allows users to share files, programs, or data with a minimum of effort. A LAN is usually local; this means that the machines are located in one physical location -- like a building or just one floor of a building. A LAN tends to use just one set of networking options. For example, a LAN generally uses one network operating system, one type of cable, and one logical topology. A LAN is usually set up for a small group of people such as a department or a division. A LAN is not limited to any particular computer operating system. DOS, Macintosh, and UNIX can all run across a LAN. Actually, they can all run across the same LAN at the same time, if the right software is used.

2.3.2 Wide-Area Network (WAN)

While the geographic distinctions of "local" and "wide" area networks imply a difference in the distance between network nodes that is not always the case. By definition, a Wide Area Network (WAN) is a government-regulated public network or privately owned network that crosses into the public network environment. It doesn't matter whether the area being bridged is across the country or across the street. If the geographical separation crosses over a public thoroughfare, a WAN is required to make the connection.

The WAN is typically used to connect two or more local area networks (LANs). As you know, a LAN is a privately owned communications system that is designed to allow users to access and share resources (computers, printers, servers) with other users. LANs that are interconnected by a WAN may be located in the same geographical area, such as an industrial park or campus setting, or in geographically separate areas, such as different cities or even different regions.

2.3.3 Internet

Internet is a collection of communication networks interconnected across 2 or more LANs or sub-networks. It is a global network connecting millions of computers. More than 100 countries are linked into exchanges of data, news and opinions.

Each Internet computer, called a host, is independent. Its operators can choose which Internet services to use and which local services to make available to the global Internet community.

There are a variety of ways to access the Internet. Most online services, such as America Online, offer access to some Internet services. It is also possible to gain access through a commercial Internet Service Provider (ISP).

2.3.4 Intranet

Intranet is a term used to refer to the implementation of internet technologies within a corporate organization rather than for external connection to the global Internet. It is a network based on TCP/IP protocols (an internet) belonging to an organization, usually a corporation, accessible only by the organization's members, employees, or others with authorization. An intranet's Web sites look and act just like any other Web sites, but the firewall surrounding an intranet fends off unauthorized access.

Like the Internet itself, intranets are used to share information. Secure intranets are now the fastest-growing segment of the Internet because they are much less expensive to build and manage than private networks based on proprietary protocols.

2.3.5 Conclusion for Network

Since E-Record Book is an web based application that may access by teachers and headmaster in certain schools, Internet is the most suitable network to be used in this project. Users from different states can access the system if they have Internet access. This means that users can manage their record books at anytime and anywhere despite of the limitation of geographical barrier.

2.4 Security Technology

Security is an important part in developing a web site. Without a good security system, a web site can be hacked and make the user to loose confidence of web site. SSL is considered for securing the transport of information in DECP.

2.4.1 Secure Sockets Layer (SSL)

SSL is a security protocol designed to ensure data moving between a browser and a server remains private. In theory, someone could intercept information, such as a credit card number while it is in transit between the browser and the server. One solution to prevent information from being usable if it is intercepted is to encrypt it. The most widely implemented encryption system for the web at present is SSL.

SSL is an open, non-proprietary protocol developed by Netscape Communication. It uses industry, accepted RSA public key cryptography for authentication and encryption. The SSL protocol was designed to provide a data security layer between TCP/IP and application protocols such as HTTP, Telnet, NNTP or FTP. SSL provides data encryption, server authentication, message integrity and optional client authentication for TCP/IP connection.

The advantage of the SSL Protocol is that it is application protocol independent. A "higher level" application protocol (e.g. HTTP, FTP, TELNET, etc.) can layer on top of the SSL Protocol transparently. The SSL Protocol can negotiate an encryption algorithm and session key as well as authenticate a server before the application protocol transmits or receives its first byte of data.

2.5 Web Server

A Web server is a program that serves Web pages upon request. Every Web server has an IP address and possibly a domain name. For example, if an user enter the URL http://www.thinkwave.com/index.html in your browser, this sends a request to the server whose domain name is thinkwave.com. The server then fetches the page named index.html and sends it to the user's browser. Web servers and browsers communicate using HTTP (Hypertext Transfer Protocol), a simple but effective language for requesting and transmitting data over a network.

Web servers come in various shapes and sizes. They run under a variety of operating systems, have varying levels of power and complexity, and range in price from rather expensive to free. Studies on several web servers will be carried out: Apache, Microsoft Internet Information Server (IIS) and Personal Web Server (PWS).

2.5.1 Internet Information Server (IIS) 5.0

Microsoft IIS 5.0 is an enterprise-level Web server that is included with Windows 2000. This version, which comes exclusively as part of the Windows 2000 Server operating system, contains many new features along with performance and reliability enhancements. **(H. M. Deitel, 2000)**

IIS 5.0 is good as both a first-time Web server for those familiar and comfortable with Windows operating systems, and a high-end server for hosting providers and large corporate installations. It handles the basics well and is better integrated in Windows than previous versions. IIS 5.0 also comes with performance and feature enhancements

that will be attractive for mission-critical tasks.

The ideal computer to run IIS on is at least a 200 MHz Pentium with 128 MB of RAM. Organizations should plan on doubling the RAM and CPU speed if they intend to run Advanced Server's clustering, SQL or Transaction services on the same machine as the Web server.

2.5.2 Personal Web Server (PWS)

PWS is entry-level/mid-range server for Windows 9x/NT platforms. It is a scaled-down version of the commercial Information Internet Server (IIS) included with the Server edition of Microsoft Windows NT. PWS is a great entry-level Web server that makes it easy to publish personal home pages, serve small Web sites, and share documents via a local intranet.

PWS is one of the best servers available for helping to get users up and running quickly. Wizards are included to guide users through the process of setting up home pages and sharing files, and the PWS administrator reduces the complexity of actually running the Web server itself. Users can also use the familiar Explorer interface or PWS's Personal Web Manager to share directories, start and stop the server, and view Web site statistics.

One of the best uses for PWS is as a platform for testing out Web sites on Windows 95/Windows NT Workstation computers before hosting them on the Internet. This allows users to check the validity of links, scripts, and applications as well as to ensure that the overall organization of the site is functioning correctly.

PWS presents the ability to develop transactional Web applications using the Microsoft Transaction Server. Overall, while most large enterprises will likely bypass Microsoft's Personal Web Server for the high-end Internet Information Server, PWS will remain one of best available options for individuals wanting to serve their own personal home pages and for small organizations needing to host their own Web sites.

2.6 Operating System

Operating system (OS) is a platform that performs basic tasks, such as recognizing input from the keyboard, sending output to the display screen, keeping track of files and directories on the disk, and controlling peripheral devices such as disk drives and printers.

Besides that, the OS makes sure that different programs and users running at the same time do not interfere with each other. For security, OS ensures that unauthorized users do not access the system. OS provides a software platform to allow application programs run on it.

The most popular operating systems currently are Windows 98, Linux, Windows 2000 and Windows XP.

2.6.1 Windows 98

Windows 98 is based on the popular Microsoft Windows 95 Operating System, and is designed for the consumer market. Windows 95/98 were designed for backward compatibility with older DOS and 16bit programs, as well as providing a platform for

the newer (back in 1995) 32 bit programs.

Windows 98 works better by making it simple to access the Internet and by providing better system performance along with easier system diagnostics and maintenance. With Windows 98, users' system plays better as well with support for the latest graphics, sound, and multimedia technologies, the ability to easily add and remove peripheral devices with support for Universal Serial Bus (USB), and it also enables users to watch TV on PC. Besides that, Windows 98 is compatible with more software (including games) and hardware.

2.6.2 Linux

Linux has gradually become a popular operating system for Internet/ intranet serving purposes. With a host of performance enhancements that will benefit Web sites and Internet sites of all sizes, Linux is a stable and high-performance operating system for Internet usage.

Linux has made progress, primarily in functionality important to Internet infrastructure and Web server capabilities, including a greater selection of drivers, easier installation, and GUI-based front ends for Web administration and window management.

2.6.3 Windows 2000

Windows 2000 is Microsoft's latest version of popular Windows NT Operating System. Windows 2000 Server has big improvement over Windows NT 4.0. The changes, both fundamental and cosmetic, have made Windows 2000 faster, more

reliable, heavier-duty, and easier to use.

2.6.4 Windows XP

Windows XP Professional has great capability with its overall security which has been improved, making it even safer for users to shop and browse on the Internet. Users can also communicate with other people on other networks without worrying about compromising their privacy or personal data files. Performance in Windows XP is at an all-time high, allowing users to use more programs and have them run faster than ever.

Windows XP Professional is dependable and stable, so users can always rely on the performance and effectiveness of their computer. Best of all, compatibility with other programs is better than ever.

Windows XP Professional has many features and tools that will make using computer easy, effective, and entertaining. For example, a user can use Remote Desktop to access his work computer and its resources from home, and to view files and documents on his computer's desktop from a co-worker's computer. With NetMeeting he can have virtual meetings with anyone, anywhere, or even participate in discussions using audio, video, or chat.

2.7 Database Server

A database is a structured collection of data. To add, access, and process data stored in a computer database, a database server is needed. There are several database

server available currently: Oracle, SQL Server and MySQL.

2.7.1 Oracle

Oracle is a multi-user database. It provides unprecedented ease-of-user and is pre-tuned and pre-configured for today's dynamic workgroup and line-of-bus environment.

Oracle includes a fully integrated set of easy-to-use management tools, full distribution, replication and web features. Oracle also provides the highest levels of availability through fast fail over, easier management, and zero data loss disaster protection, with Data Guard, the only complete data protection solution available on the market.

Oracle can runs on UNIX, Linux and Windows platform. However, it is expensive and separate licenses are required for each of its database engine.

2.7.2 Microsoft SQL Server

Microsoft SQL Server 2000 is a single process, multithreaded relational database server primarily intent for transactional processing. It is based on the client/server architecture, which divides processing into two components: a front-end, or client component, that run on a local workstation and a back-end, or server component, which runs on a remote computer.

SQL Server 2000 able to support databases of almost any size. In fact, SQL Server database and the applications the user use with the database typically takes one of two forms which are Online Transaction Processing (OLTP) system, in which users

continually make changes to the data in the database. For example, the database system for recording customers' orders at Amazon.com is an OLTP system. An Online Analytical Processing (OLAP) system, in which user primarily focus on analyzing the data in the database. He typically don't make many changes to such databases. For example, if he has four different retail stores, each with its own inventory and order database. In this environment, he would use an OLAP system to combine the data from each of the four databases for performing analysis such as sales trends, customer demographics, and so on.

2.7.3 MySQL

MySQL is a relational database management system. MySQL stores data in separate tables rather than putting all the data in one big storeroom. This adds speed and flexibility. The tables are linked by defined relations making it possible to combine data from several tables on request.

MySQL is a small, compact, easy to use database server, ideal for small and medium sized applications. It is client/server implementation that consists of a server and many different client programs. It is available on a variety of UNIX platforms, Linux, Windows NT, Windows 95/98 and Windows 2000.

MySQL is Open Source Software. Open Source means that it is possible for anyone to use and modify. Anybody can download MySQL from the Internet and use it without paying anything. Anybody can study the source code and change it to fit their needs.

2.8 Data Access Technology

E-Record Book will require data access technology to enable communication and access to its various databases. A few of the Microsoft Data access strategy and technology is reviewed and considered.

2.8.1 Universal Data Access (UDA)

UDA is a high-level specification developed by Microsoft for accessing data objects regardless of their structure. The strategy of Universal Data Access is to assure open, integrated, standards-based access to all types of data, which is from SQL to non-SQL to even unstructured data across a wide variety of applications, from traditional client/server to the web. The main components of UDA are ADO, OLE DB and ODBC.

2.8.2 ADO (ActiveX Data Object)

ActiveX Data Object (ADO) is the Microsoft's newest high-level interface for data objects that most applications developers will use.

ADO is designed to eventually replace *Data Access Objects (DAO)* and *Remote Data Objects (RDO)*. Unlike RDO and DAO, which are designed only for accessing relational databases, ADO is more general and can be used to access all sorts of different types of data, including web pages, spreadsheets, and other types of documents.

ADO provides consistent access to data for creating a front-end database client or middle-tier business object using an application, tool, language, or even an Internet browser. ADO is the single data interface for developers creating 1 to n-tier client/server

and Web-based data-driven applications.

2.8.3 OLE DB

OLE DB Providers are the data access engines or services, as well as the business logic components that these applications can use in a highly interoperable, component-based environment.

OLE DB is a set of interfaces that are designed to provide data access to *all* data, regardless of type, format or location. It effectively make database in several components and related data processing functionality, breaking it up into interoperable components that can run as middleware on the client or server across a wide variety of applications. The OLE DB architecture provides for components such as direct data access interfaces, query engines, cursor engines, optimizers, business rules and transaction managers.

The concept of OLE DB is to explode the database into its basic parts. OLE DB delivers components, external to the database, that provide this typical database functionality in reusable component architecture. These components, because they are not directly linked to the database itself, can be shared across multiple applications, systems and data stores to provide a higher level, universal interface.

2.8.4 ODBC (Open Database Connectivity)

ODBC is a standard database access method developed by Microsoft Corporation. The goal of ODBC is to make it possible to access any data from any application, regardless of which database management system (DBMS) is handling the

data. ODBC manages this by inserting a middle layer, called a database *driver* , between an application and the DBMS. The purpose of this layer is to translate the application's data queries into commands that the DBMS understands. For this to work, both the application and the DBMS must be *ODBC-compliant* -- that is, the application must be capable of issuing ODBC commands and the DBMS must be capable of responding to them. Since version 2.0, the standard supports SAG SQL.

2.9 Language

2.9.1 ASP

Active Server Pages is an open, compile-free application environment in which programmers can combine HTML, scripts, and reusable ActiveX server components to create dynamic and powerful Web-based business solutions. Active Server Pages enables server-side scripting for Internet Information Services (IIS) with native support for both VBScript and Jscript.(Microsoft site, 1998) When a browser requests an ASP page, the Web server generates a page with HTML code and sends it back to the browser.

One of the most important features about ASP is that it allows user to easily access data and put it on a Web page. User can simply display data from an ODBC-compliant database, or use ASP to make decisions about what to display on a Web page. User can then format the results in any way that they please.

Another important ASP feature is the ability to use cookies to store and retrieve

information. The Request object has a Cookie collection, and user can use this in data processing.

2.9.2 PHP

PHP (Hypertext Preprocessor) is a open-source server-side, HTML embedded scripting language used to create dynamic Web pages for e-commerce and other Web applications. In an HTML document, PHP script (similar syntax to that of Perl or C) is enclosed within special PHP tags. Because PHP is embedded within tags, the author can jump between HTML and PHP (similar to ASP and Cold Fusion) instead of having to rely on heavy amounts of code to output HTML. And, because PHP is executed on the server, the client cannot view the PHP code.

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

PHP is the natural choice for developers on Linux machines running Apache server software, but runs equally well on any other UNIX or Windows platform, with Netscape or Microsoft Web server software. PHP also supports HTTP sessions, Java connectivity, regular expressions, LDAP, SNMP, IMAP, COM (under windows) protocols. It also supports WDDX complex data exchange between virtually all Web programming languages.

2.9.3 Cold Fusion

Cold Fusion is a product created by Allaire Corporation of Cambridge, Mass. that includes a server and a development toolset designed to integrate databases and Web pages. Cold Fusion web pages include tags written in Cold Fusion Markup Language (CFML) that simplify integration with databases.

Coding for Cold Fusion pages is much more straightforward and intelligible than JavaScript, VBScript, C++ or Java, even while providing high levels of functionality. The tags themselves conform to the basic HTML syntax of tag name followed by tag attributes, and are enclosed in the familiar HTML brackets (<>). Most tags are two-sided, and can be combined with each other and with HTML elements to create custom tags for use in Cold Fusion applications.

2.9.4 JSP (Java Server Pages)

Java Server Pages™ (JSP) is a web-scripting technology that can mix static HTML content with server-side scripting to produce dynamic output. By default, JSP uses Java as its scripting language; however, the specification allows other languages to be used, just as ASP can use other languages (such as JavaScript and VBScript). While JSP with Java will be more flexible and robust than scripting platforms based on simpler languages like JavaScript and VBScript.

JSP provides a number of server-side tags that allow developers to perform most dynamic content operations. So developers who are only familiar with scripting, or even those who are simply HTML designers, can use JSP tags for generating simple output. Advanced scripters or Java developers can also use the tags, or they can use the full Java

language if they want to perform advanced operations in JSP pages.

2.9.5 JavaScript

JavaScript is a scripting language developed by Netscape to enable web authors to design interactive sites. JavaScript is different from Java. Although it shares many of the features and structures of the full Java language, it was developed independently. JavaScript can interact with HTML source code to enable web authors to spice up their sites with dynamic content. JavaScript is endorsed by a number of software companies and is an open language that anyone can use without purchasing a license. It is supported by recent browsers from Netscape and Microsoft, though Internet Explorer supports only a subset, which Microsoft calls Jscript.

2.10 Authoring Tools

2.10.1 Microsoft Visual InterDev 6.0

Microsoft Visual InterDev is a Web development tool designed for programmers to create an interactive Web page with data is as simple as dragging and dropping, setting some properties, and saving the page. No coding is required in using Visual InterDev.

Visual InterDev includes site design tools that help user easily plan pages, organize their links, and apply a consistent theme to your Web site. Visual InterDev includes three ways to view your HTML and ASP pages.

These three views are the cornerstone of Visual InterDev. They replace the simple source code editor included with Visual InterDev 1.0 and supports design-time controls (DTCs), debugging, statement completion, and object browsing.

The new data environment provides easy commands for making Web application data-driven. Instead of burying complex SQL statements deep within an .asp file, the statements are now exposed, maintained, and reused at the application level through the data environment under the Global asp file. Instead of modifying the query within each page, developers can modify the data command and changes are incorporated into files that reference that data command. Developers also can drag fields from the command directly onto HTML or ASP page.

However, for those so inclined, Visual InterDev exposes a full object model that allows developers to fine-tune their application, perform client validation, and have full control of Web application. Visual InterDev supports not only full-reach applications, using the ASP engine to produce simple HTML pages for the client, but also DHTML and Microsoft Internet Explorer 4.0 data binding for a richer client experience.

2.10.2 Notepad

Notepad is the world's most versatile HTML editing tool absolutely free when purchase this software: Windows version 2.0 and above.

Notepad has one of the simplest user interfaces of any Internet Web authoring tools. The menus are logically laid out, conforming to all standards in design, so users can understand them before use Notepad.

Notepad has the same interface for all versions of Windows, so moving over to the latest version of Windows should not hamper HTML code creation. The Notepad web-authoring tool is compatible with every single standard of Internet presentation medium yet devised. Notepad was designed to have a very small application footprint, taking up as little space as possible in computer's memory, and a minimum of disk space.

Notepad gives clear, easy to read and full HTML. There is no code hidden, and users have control over all parts of the HTML code. JavaScript is also fully supported by Notepad. All parts of the JavaScript are fully available through Notepad, without the need of complex tools.

2.10.3 Macromedia Dreamweaver

Macromedia Dreamweaver is professional visual editor for creating and managing web sites and pages. It gives developers the productivity of a visual web page layout tool, the control of an HTML text, editor and support for new web technologies, all in one software packing.

Developers can use it to create web sites visually, with confidences that HTML being generated is concise and always editable. It includes advanced features that take advantage of the latest innovations on the web, such as dynamic HTML and CSS, while still ensuring that web pages work well in a variety of web browsers. All of the code generated by it is carefully created to work on as many platforms and browsers as possible.

Others features include easy integration of ActiveX components, Java applets,

Plug-ins for improved web page interactivity. It also integrates seamlessly with other components of Macromedia, such as Flash Movies, Shockwave, and Fireworks, which are essential for the development of interactive web pages.

2.10.4 Adobe Photoshop 7.0

Adobe Photoshop 7.0 is the most popular image-editing available for Macintosh and Windows-based computers. It is used as drawing, painting and designing purposes. Users can retouch an image, apply special effects, swap details between photos, introduce text and logos, adjust color balance, and even add color to a grayscale scan. All these functions are included under a set of user-friendly editing tools in Adobe Photoshop. It contains graphical icons to represent every functions of each button. Besides that, it also provides many shortcut keys that is easier and save time for users and for those who do not like to use mouse.

Chapter 3 - Methodology

3.1 Object-Oriented System Development

The object-oriented software development life cycle (SDLC) consists of three macro processes: object-oriented analysis, object-oriented design and object-oriented implementation (see Figure 3-1).

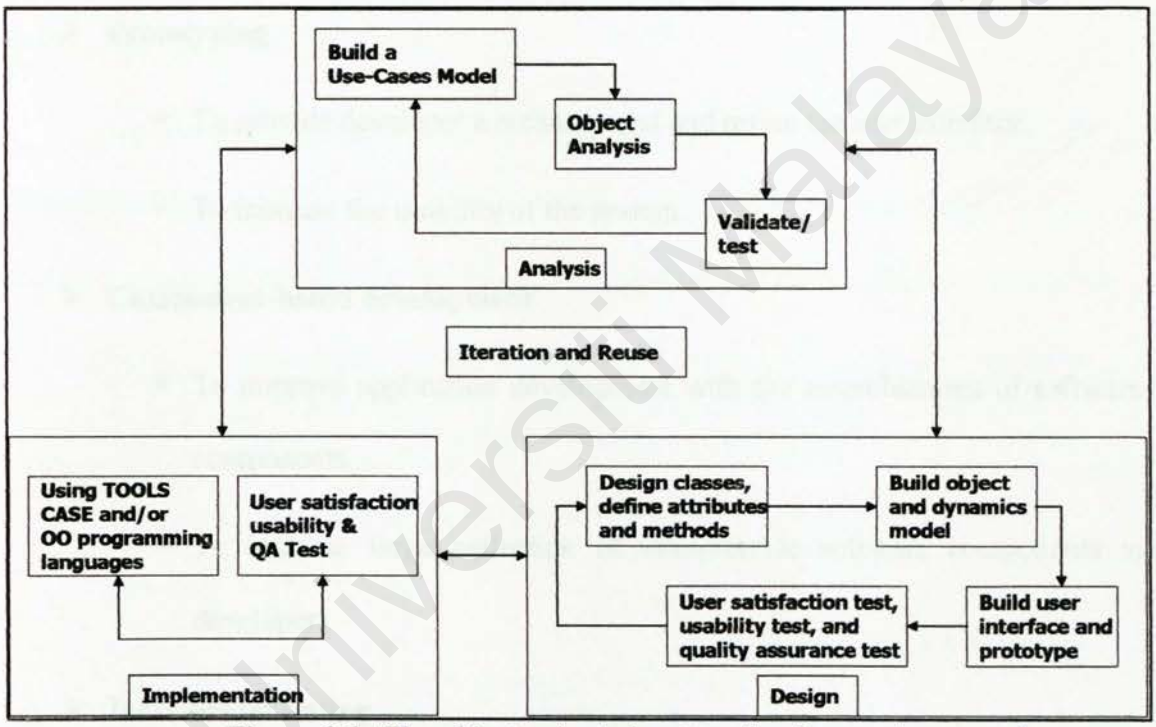


Figure 3.1: The object-oriented system development approach

Object-oriented system development includes these activities:

- **Object-oriented analysis-use case driven**
  - To determine the system requirements and identifying classes and their relationship to other classes in the problem domain.

- To identify the users or the actors.

➢ **Object-oriented design**

- To design the classes identified during the analysis phase and the user interface.
- To design and redefine classes, attributes, methods, structures and associations.

➢ **Prototyping**

- To provide developer a means to test and refine the user interface.
- To increase the usability of the system.

➢ **Component-based development**

- To improve application development with the assemblement of software components.
- To increase large collection of interpretable software components to developers.

➢ **Incremental testing**

- To perform testing regularly for finding bugs and performance.
- To have a clear picture of the system characteristics.

3.2 Object Oriented Methodology

Object-oriented methodology is a set of methods, models, and rules for developing systems. Modelling is the process of describing an existing or proposed system and it can be used during any phase of the software life cycle. Modelling also provides a means for communicating ideas in an easy to understand and unambiguous form while also accommodating a system's complexity. **(Ali Bahrami, 1999)**

3.2.1 The Unified Approach

The Unified Approach (UA) (see Figure 3-2) establishes a unifying and unitary framework around their works by utilizing the unified modelling language (UML) to describe, model and document the software development process. **(Ali Bahrami, 1999)**

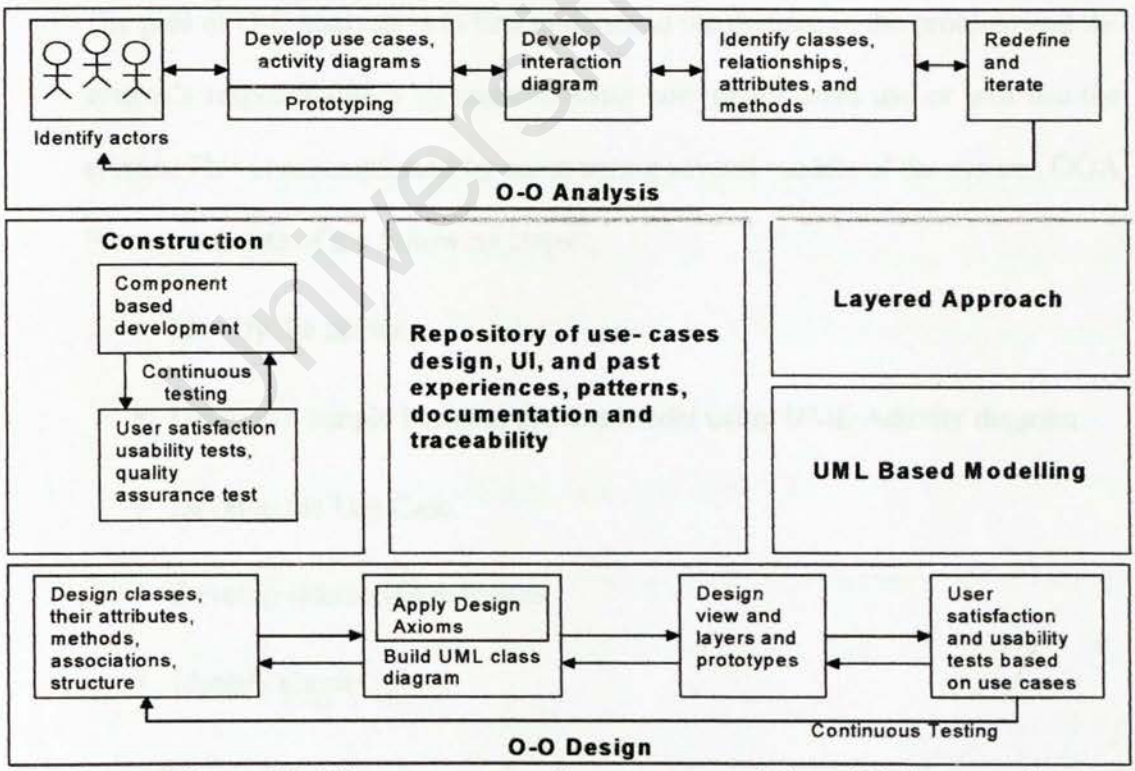


Figure 3.2: The processes and components of the unified approach

The Unified Approach is chosen for E-Record Book because:

- To combine best practices, processes, methodologies and guidelines along with UML notations and diagrams.
- To understand better E-Record Book based on object-oriented concepts and system development.
- To allow iterative development by allowing to go back and forth between the design and the modelling or analysis phases.

The unified approach to software development revolves around the following processes and concepts (see Figure 3-2). The processes are:

- **Object-Oriented Analysis (OOA)**

  The goal of O-O analysis is to first understand the domain of the problem and the system's responsibilities by understanding how many users use or will use the system. This is accomplished by constructing several models of the system. OOA Process consists of the following steps:

  - Identify the actors.
  - Develop a simple business process model using UML Activity diagram.
  - Develop the Use Case.
  - Develop interaction diagrams.
  - Identify classes.

- **Object-Oriented Design (OOD)**

  Unified approach combines Jacobson et al.'s analysis and interaction diagrams, Booch's object diagrams, and Rumbaugh et al.'s domain models to produce designs that are traceable across requirements, analysis, design, coding and testing. OOD Process consists of:

  - Designing classes, their attributes, methods, associations, structures ad protocols, apply design axioms.
  - Design the Access Layer.
  - Design and prototype User interface
  - User Satisfaction and Usability Tests based on the usage/use cases.
  - Iterate and refine the design.

- **Iterative Development and Continuous Testing**

  Iteration and reiteration can uncover design weaknesses or at least provides additional information for user to use, repeat the entire process or moving on to reprototyping and retesting. Prototypes made will be incrementally transformed into actual application.

- **Modelling Based on the Unified Modeling Language**

  The unified modelling language was developed by the joint efforts of the leading object technologists Grady Booch, Ivar Jacobson and James Rumbaugh. The UML merges the best of the notations used by the three most popular analysis and design methodologies: Booch's methodology, Jacobson et. Al.'s use case

and Rumbaugh et al.'s object modelling technique. The UA uses UML to describe and model the analysis and design phases of system development.

- **The UA Proposed Repository**

  Repository enables maximum reuse of previous experience and previously defined objects, patterns, frameworks and user interfaces in an easily accessible manner. The repository should be accessible to many people and relatively easy to search the repository for classes based on their attributes, methods or other characteristics

- **The Layered Approach to Software Development**

  Approach to system architecture isolates the functions of the interface from the functions of the business and the details of the data access. This approach uses three layered approach which are user interface layer, a business layer and an access layer.

3.3 Technique Used To Define Requirements

Effective and appropriate techniques must be used to define and elicit users requirements. Research methods that usually used are library research, interview, internet research.

3.3.1 Library Research

I went to library to search books related to record book management system to

have a deeper understanding on the system design and how can the electronic record book management system be implemented effectively.

3.3.2 Interviewing Teachers

A few interview sessions had been conduct with some teachers in Sekolah Menengah Kebangsaan Sri Rahmat, Johor Baharu to have a better understanding of their need toward the management of the record book so that E-Record Book can help them better. From those sessions, I got a clearer view of what E-Record Book should provide to produce better teaching effect. I can see that teachers need better record management tools and need a proper system to handle their teaching material well. These helpful teachers also gave precious advices and suggestions for me to produce a better system.

3.3.3 Internet Research

I have surf around the Internet for sometime to gain deeper understanding about record management and web technology available. For the technology aspect, ASP was chosen for its scalability and portability and most important of all is its easiness to integrate with current and future system.

Summary of Techniques Used to Define Requirements

From the research above, I found that both teachers and headmaster need better communication between each other. They are also looking forward for a more user-friendly system to upgrade the teaching and learning process. The system that they are expecting should achieve basically the goal below:

- Bring the teachers and headmaster together (better communication).

- Well control on the teaching and learning material. (easier to manage)
- Instant result.

3.4 Chosen Platform, Web Server, Database and Tools

3.4.1 Chosen Development

For the E-Record Book, Windows is chosen as the development platform. Microsoft's Windows 2000 Server is built to work with a series of microprocessors from the Intel Corporation that share the same or similar sets of instructions.

The main reason for choosing Microsoft's Windows 2000 Server as the development operating system is because it supports the MS SQL Server 2000dtabase platform. Therefore, the implementation of the new system can be done easily and effectively.

3.4.2 Chosen Database Management System

I have chosen Microsoft SQL Server 2000 as the database platform for Electronic Record Book Management System (E-Record Book) as it is an RDMS (relational database management system) for the workstations. SQL Server enforces database integrity, it handles all queries on the server itself, and just returns a result set. SQL Server also supports stored procedures, triggers, defaults, rules and other mechanisms for defining a database, modifying and retrieving data.

SQL Server contains not only a very powerful database server, but a group of

products that will help you administer your server. SQL Server comes with a suite of graphical tools that allow database administrators and developers to install, configure, and administer the database with little effort.

Below are some of the reason why I prefer SQL Server than any other database platform:

- The need to have a fast database for on-line record processing.

  Security enforcement at the database level and support for entity, domain, referential and school policy integrity.

- Ease of installation, deployment and use.

  SQL Server is by far the easiest database system to install, deploy to users on a Windows 2000 system. It is extremely easy to use with the wide variety of wizards available and the great graphical tools.

- Scalability

  SQL Server can scale from a single user system on a Windows 2000 platform all the way to an enterprise wide usage with thousands of users across a wide area network.

- Data Warehousing

  With new features built in for on-line analytical processing (OLAP), SQL Server is ready to house a larger database and be able to retrieve that data in a timely and easy format. An English query processor will help users ask questions of the database and get the correct answers back.

- Integration with other OS Services

SQL Server easily integrates with other Windows NT services such as NT security, email, the internet and other services.

- XML Support

SQL Server has great XML support. It can retrieve data in an XML format, and, with a little help from some system procedures, it can use XML to modify data in a database.

The following table is a list of the capacities for SQL Server 2000.

Object	Limits
Databases per server	32,767 databases.
Size of database	Over 1 million terabytes.
Tables per database	2 billion tables per database.
Columns per table	1024
Rows per table	Limited only by available storage
Bytes per row	8060. This does not include text and image data types.
File size of data	32 TB
File size of log data	32 TB
Column names & variable name length	128 characters
Indexes per table	1 clustered index per table. 249 nonclustered indexes per table. A composite index may have up to 16 columns.
Triggers per table	Limited only by the number of objects in a database
Stored procedures	Can have up to 255 parameters, and can nest up to 16 levels deep.
User connections	Depends on licensing

Table 3.1: Capacities of SQL Server 2000

3.4.3 Chosen Development Data Access Technology

ActiveX Data Objects (ADO) is chosen as the data access technology for E-Record Book due to the reasons below:

- ADO is a standard database access which is compatible with SQL Server 2000.
- ADO is a cross language technology for data access that exposes an object model incorporating data connection objects, data command objects, Recordset objects and collections within these objects.
- ADO object model provides an easy-to-use set of objects, properties and methods for creating script that accesses data in databases.

3.4.4 Chosen Development Web Server

Microsoft Internet Information Services (IIS 5.0) has been chosen for this project as it allows publication and distribution of information on the Internet. IIS consists of three different components which are World Wide Web (WWW) server, File Transfer Protocol (FTP) server and Gopher server. These components support virtual servers, virtual directories, logging to ADO databases, Common Gateway Interface (CGI), Internet Server Application Programming Interface (ISAPI) and Secure Socket Layer (SSL).

3.4.5 Chosen Web Development Tool

ASP (Active Server Pages) has been selected as the web development tool for the proposed system. The reasons of choosing ASP are as follows:

Strength of ASP

- VBScript is usually used for scripting ASP pages.

  The strength of VBScript is that it is derived from Visual Basic, and literally millions of people have at least passing familiarity with Visual Basic. The support for VBScript in Microsoft Visual InterDev (part of Microsoft Visual Studio) includes the IntelliSense code completion ability that many programmers have become accustomed to. For example, while creating VBScript code, I only needed to type Response followed by a period and I was presented with a list of the methods and properties of the Response object. Highlight the correct method or property (either by using the arrow keys or by typing the first letter of the method or property) and press the Tab key, and the method or property appears in the code, with further help on the arguments required for any methods.

- ASP provides fast resulting application.

  That is, a programmer could certainly find ways to improve performance, but the application might already be fast enough as written. This is especially true for intranet applications, for which he can get some idea of the likely traffic on the site when it is available.

- ASP pages also are fairly well integrated into IIS, which it fairly well integrated into Windows 2000.

  We can presume that Windows 2000 server machines will have IIS and thus will be able to run any ASP application that has been created. Documentation and support for ASP is extensive, including Microsoft's MSDN as well as third-party

articles and books.

- Debugging ASP applications has become much simpler as well.

  With IIS 5.0 and later, debugging has reached a new height. Developer can debug applications on the server or client machines, and the debugger will inform him at the line where the error has occurred. Variable values can be displayed as hints when the mouse cursor is placed over the variable in the source, or they can be displayed in a watch window. IIS 5, which is included with Windows 2000, introduces a new *Error* object that allows much greater control when an error occurs.

- ASP can take advantage of COM and DCOM (Component Object Model and Distributed Component Object Model) objects with minimum effort. The scoop on the COM objects is **request**, **response**, **server**, **session** and **application**.

Chapter 4 - System Analysis

System analysis enables us to break the system into pieces or sub system that we can understand and try to describe their interrelationships (see Figure 4-1). The relationships are essential as the sub system because sometimes they hold the clue to how to solve the larger system rather than simply the nature of the sub systems. **(Shari Lawrence Pfleeger, 2001)**

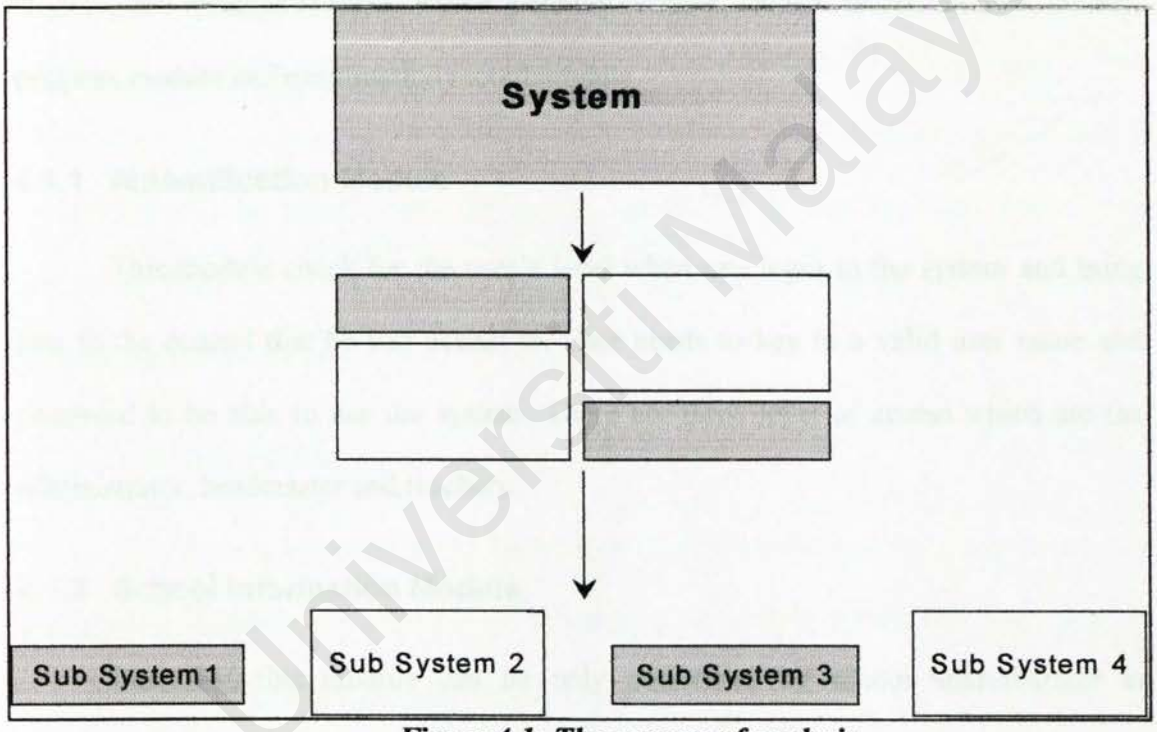


Figure 4.1: The process of analysis

4.1 Functional Requirements

Functional requirement is a statement of the service or functions that a system should provide how the system reacts to particular inputs, and how the system should behave in particular situations. **(Sommerville, 1998)**

The functional requirement for E-Record Book consists of four main parts: General Section, Administrator Section, Headmaster Section and Teachers Section. For General Section, functional requirement consists of authentication module for teachers, headmaster, school staff who acts system administrator and school information module. For Administrator Section, functional requirement includes user maintenance module while for Headmaster Section, there is the monitor and validate record book module. Finally there is the Teachers Section which includes teacher profile module, time-table module, and syllabus module, lesson planning module, student record module, student progress module and examination record module.

4.1.1 Authentication Module

This module check for the user's level when one login to the system and bring him to the control that he had access to. User needs to key in a valid user name and password to be able to use the system. There are three level of access which are the administrator, headmaster and teachers.

4.1.2 School Information Module

Basically, this module can be only performed by school administrator in providing the latest information about the school's annual events which lists out the school semester, holidays and school activities including the venues and dates. The module also provides the facts on School's vows, objectives and organization chart.

4.1.3 User Profiles Module

This module enables the administrator to create a new user account for the teachers and headmaster. Due to the security of the school's data, every user have to get a login name from the administrator before starting to use the system. He also can delete the user's account when he or she is not working in the school anymore.

4.1.4 Monitoring and Validate Module

Headmaster or the school's assistant headmaster is the only user who can perform this module as to monitor every teacher's record book in the school. The headmaster can select any teacher in the available list, monitor the teacher's performance in the record book and finally validate it. If there are any inquiries or comments from the headmaster, he can pose them to the teacher in the comments column which will be provided in the module.

4.1.5 Teacher Profile Module

This module basically stores the teacher's profile such as name, gender, race, IC, age, grade of position and others. Teachers able to fill in their profile edit them or even make changes once they have login into the system.

4.1.6 Time-table Module

There are two parts in this module which are the classroom time-table for the form teacher and the teacher's personal time-table. The classroom time-table is optional as there may be teachers who are not form teachers for certain classes. The personal time-table is compulsory for every teacher which includes the period of the teaching

subject, class involved and the time for every subject's period.

4.1.7 Syllabus Module

This module includes the syllabus of the teaching subject where teachers able to record the name of the subject, its objectives or contents according to chapters and the references used.

4.1.8 Lesson Plan Module

Teachers able to perform their daily lesson planning which includes the theme of the subject, the scope of learning and its results for certain subject, class and time. They can also input the current day and date when updating the lesson planning. This module can be viewed by headmaster as a way to validate the record book.

4.1.9 Student Record Module

As usual, this module will keep the profile about the student's name, class, age, parents' names, addresses and others. This module is only available for the teacher who holds a certain class as a form teacher.

4.1.10 Examination Record Module

This module is just for the form teacher to record the marks and grades for every subject of his or her students in the class for the mid-term and final term examination.

4.2 Non Functional Requirements

Non-functional specifications are the constraints under which a system must operate and the standards which must be met by the delivered system. **(Sommerwille, 1995)** The new Electronic Record Book Management System must ensures certain web application qualities like user-friendliness, correctness, functionality, reliability, flexibility, efficiency as well as maintainability.

4.2.1 Functionality

The functionalities stressed here are the searching and retrieving capability, which is very important in any web application that deals with data retrieval from existing database. Besides, navigation and browsing features as well as application domain-related features will be taken into account.

4.2.2 Reliability

Reliability is the extent to which a program can be expected to perform its intended function with required precision (Pressman, 2001). It is closely related to correct link processing, error recovery and user input validation and recovery. This quality is essential as it indicates how far users will be confident in the implementation of the new computerized system in getting daily minutes processing done.

4.2.3 Maintainability

System maintenance accounts would require more effort if the system is not designed according to good programming practices. Maintainability is the ease with

which a program can be corrected if an error is encountered, adapted if its environment changes, or enhanced if the customer desires a change in requirements. **(Pressman, 2001)**

4.2.4 Security

The proposed system has also security measures to minimize the risk of data exposure to unauthorized people.

4.3 Use Case Diagram for E-Record Book

In order to achieve the objective of the object-oriented analysis for E-Record Book, the use case diagram has been built. Use case diagram consists of actors and use cases. An actor plays the role with regard to the system or an entity such as another system or a database that resides outside the system. A use case is a sequence of actions that an actor performs within a system to achieve a particular goal. **(Kendall Scott, 2001)**

Below is the use case diagram:

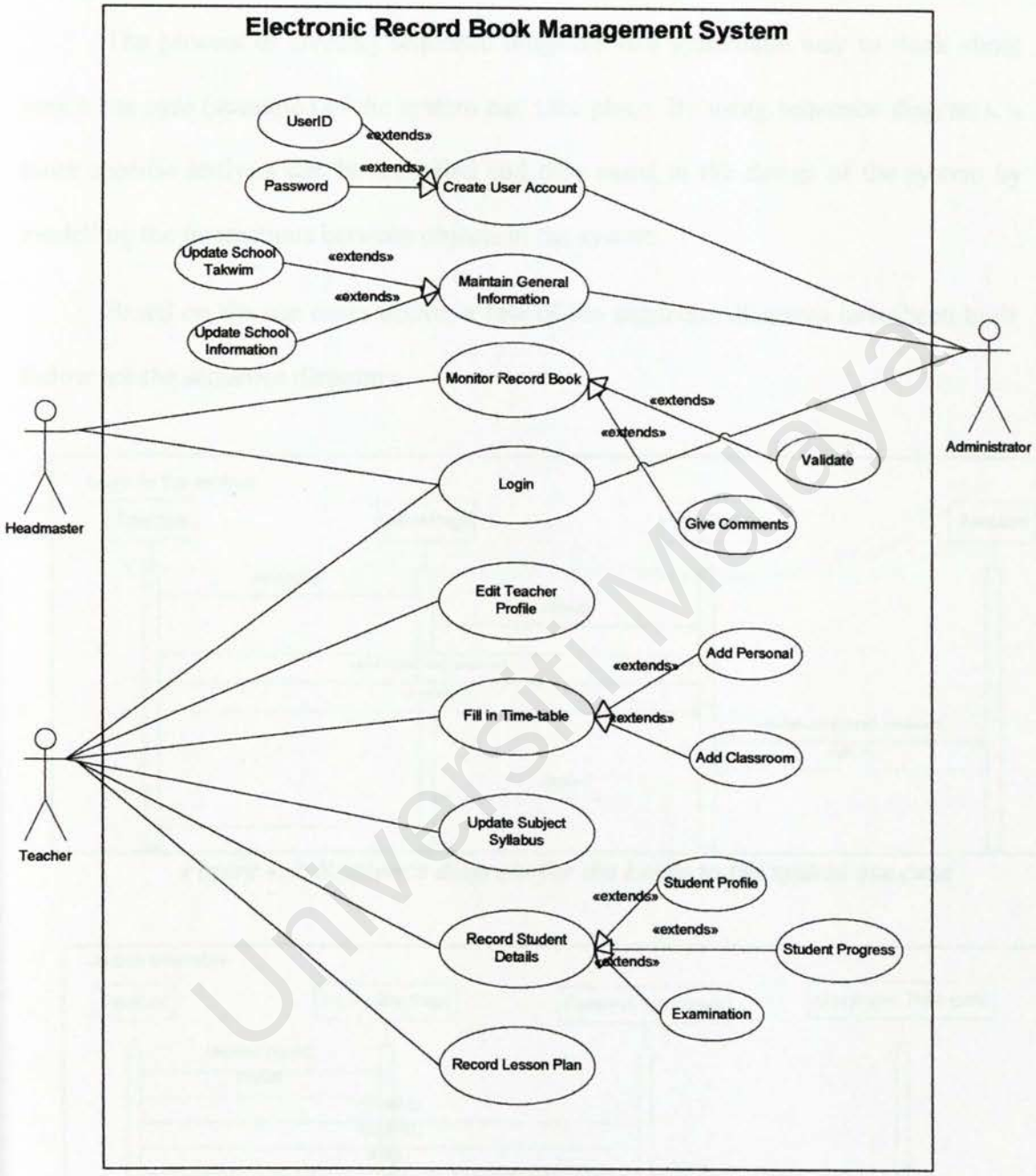


Figure 4.2: Use Case Diagram for E-Record Book

4.4 Sequence Diagrams for E-Record Book

The process of creating sequence diagrams is a systematic way to think about how a use case (scenario) of the system can take place. By using sequence diagrams, a more specific analysis can be modelled and they assist in the design of the system by modelling the interactions between objects in the system.

Based on the use cases above, a few of the sequence diagrams have been built. Below are the sequence diagrams:

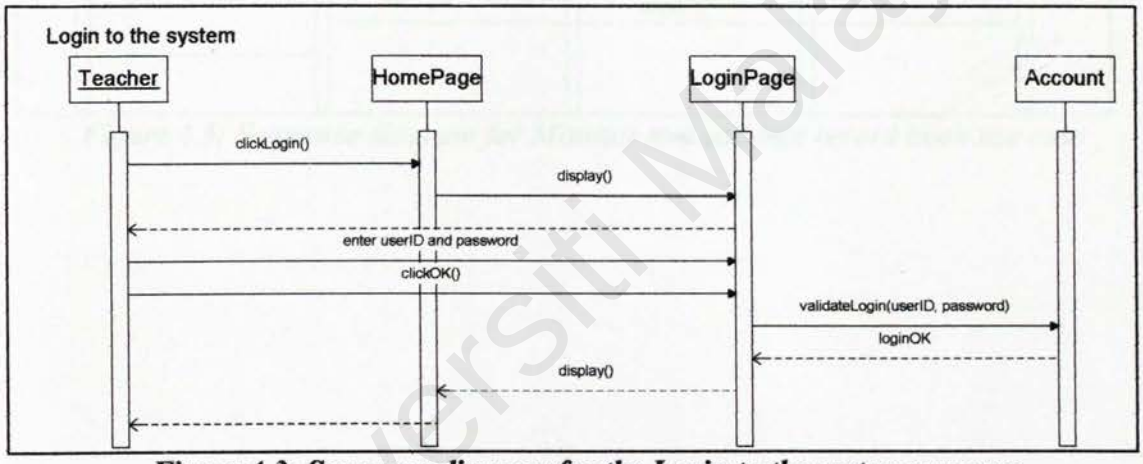


Figure 4.3: Sequence diagram for the Login to the system use case

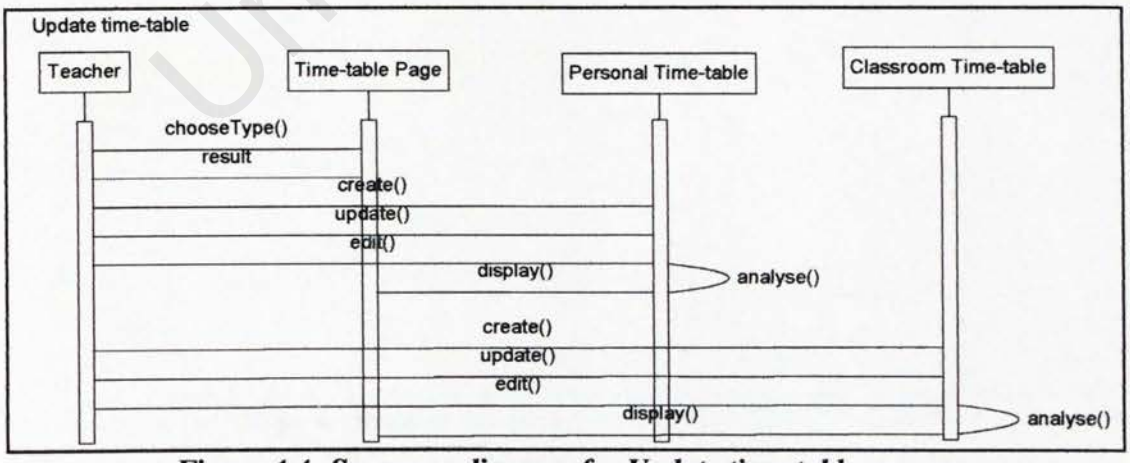


Figure 4.4: Sequence diagram for Update time-table use case

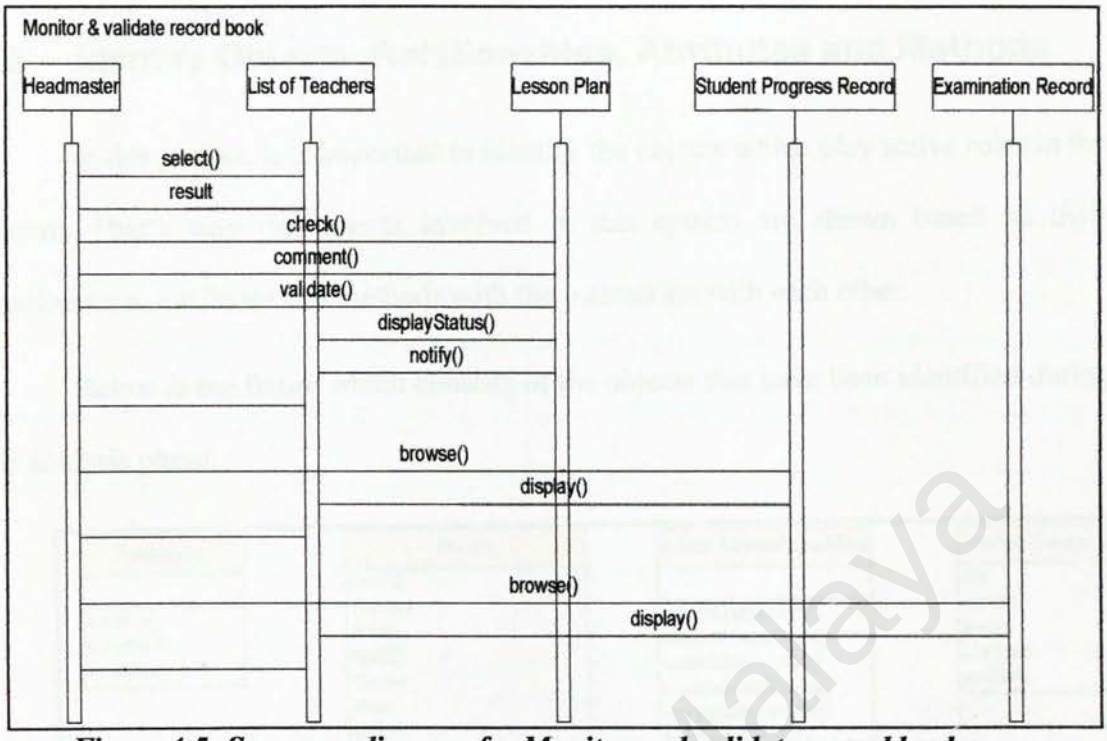


Figure 4.5: Sequence diagram for Monitor and validate record book use case

4.5 Identify Objects, Relationships, Attributes and Methods

In this project, it is important to identify the objects which play active roles in the system. That's why the objects involved in this system are shown based on their relationships, attributes and methods with the interaction with each other.

Below is the figure which consists of the objects that have been identified during the analysis phase:

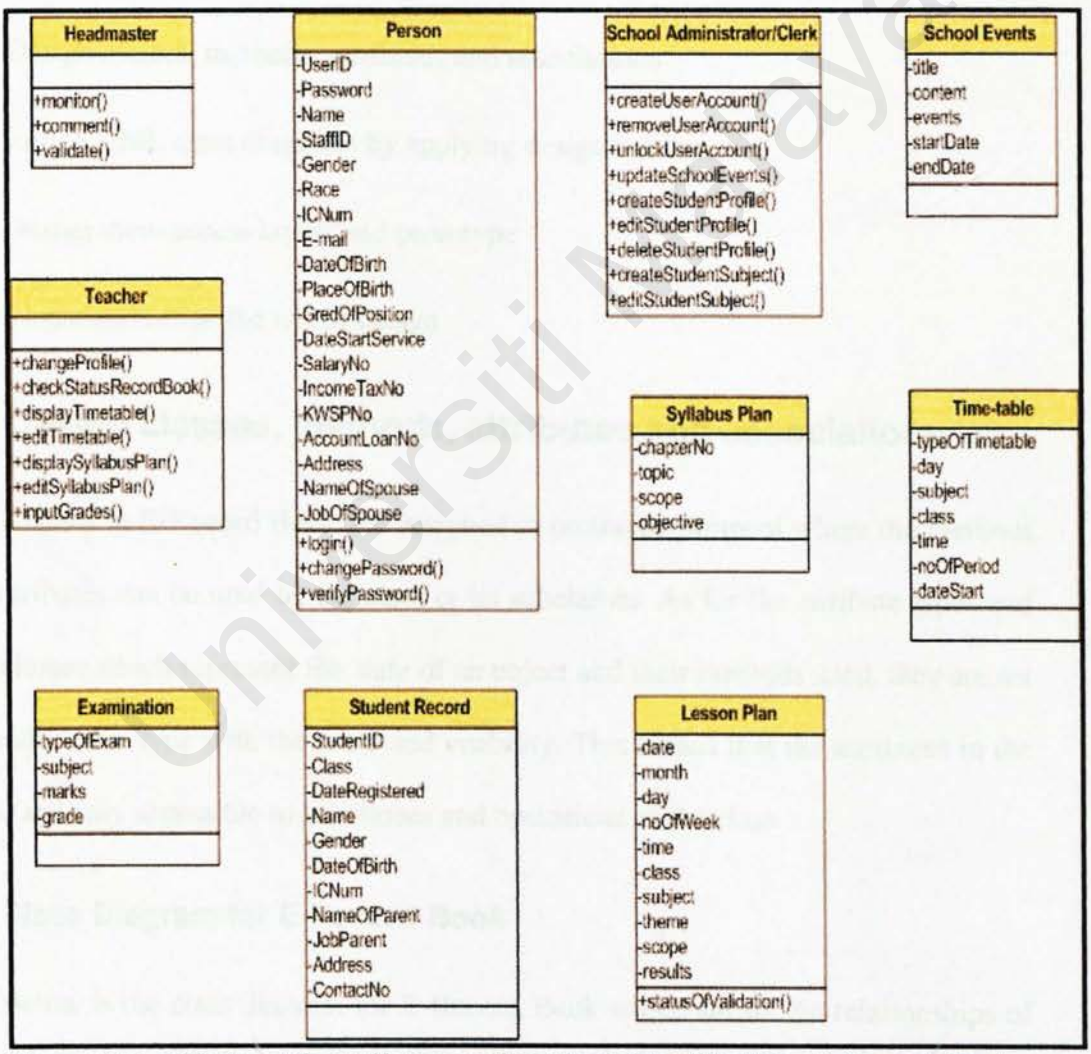


Figure 4.6: Objects with attributes and methods identified for E-Record Book

Chapter 5 - System Design

5.1 Introduction

System design requires the design developers to elevate the model into actual objects that can perform the task required by the application. The objects or classes identified during analysis phase provide them a framework for the design phase. The object-oriented design process consists of the following the activities

- Design classes, methods, attributes and associations
- Refine UML class diagrams by applying design axioms
- Design view/access layers and prototype
- Iterate and refine the whole design

5.2 Design classes, methods, attributes and associations

Classes in E-Record Book are designed in protected protocol where the methods or attributes can be used by the class or its subclasses. As for the attribute types and for classes which represent the state of an object and their methods used, they are set to multi value type with the protected visibility. This means that the attributes in the class are only accessible to subclasses and operations of the class.

5.2.1 Class Diagram for E-Record Book

Below is the class diagram for E-Record Book which shows the relationships of the classes with their attributes and methods.

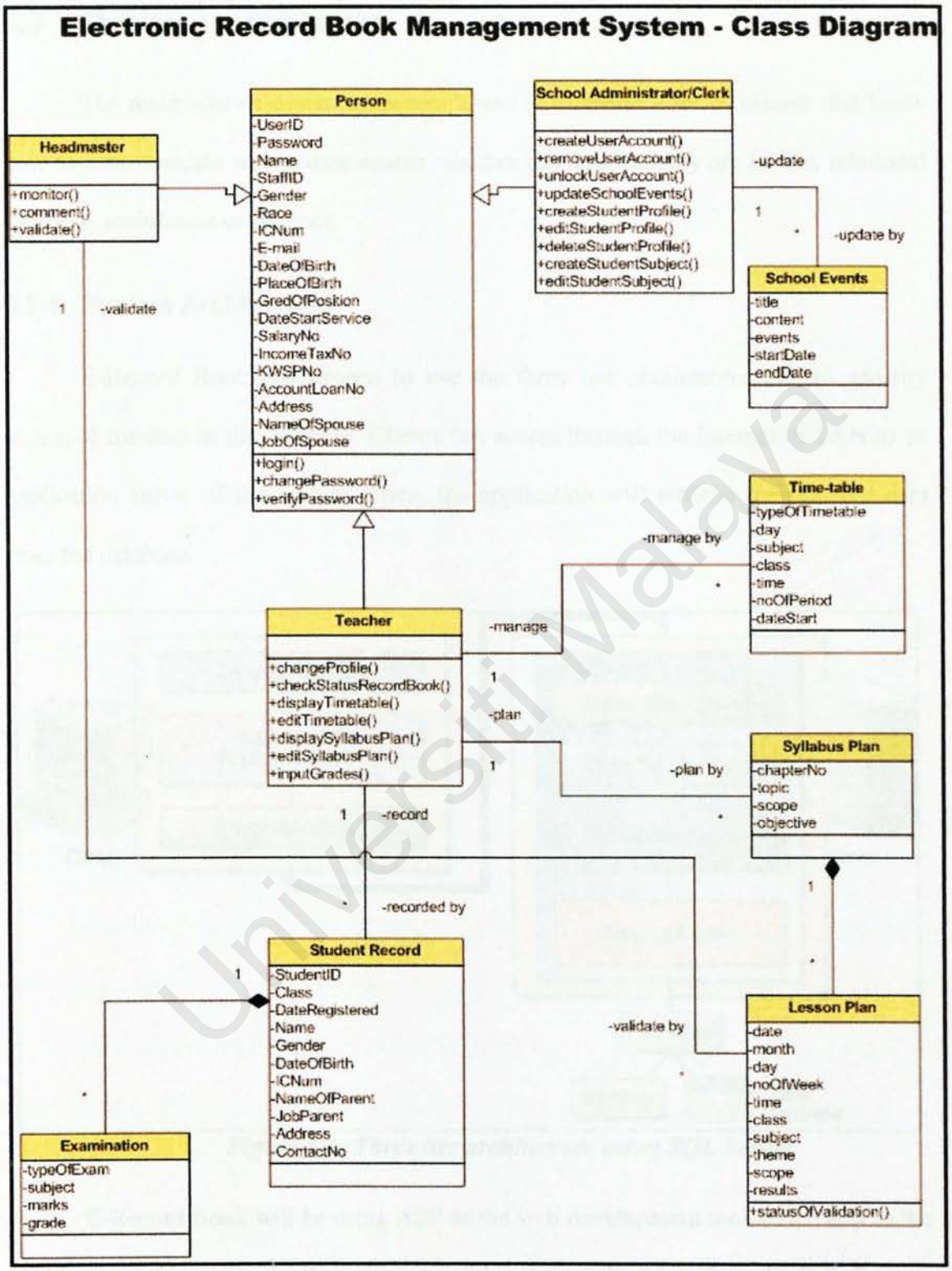


Figure 5.1: A complete UML class diagram for E-Record Book

5.3 Design access layers

The main idea of designing access layers is to create a set of classes that know how to communicate with a data source whether the data actually are in file, relational database, mainframe or Internet.

5.3.1 System Architecture

E-Record Book has chosen to use the three tier architecture due to security access of the data in the database. Clients can access through the Internet or LAN to an application server of the system. Then, the application will retrieve the required data from the database.

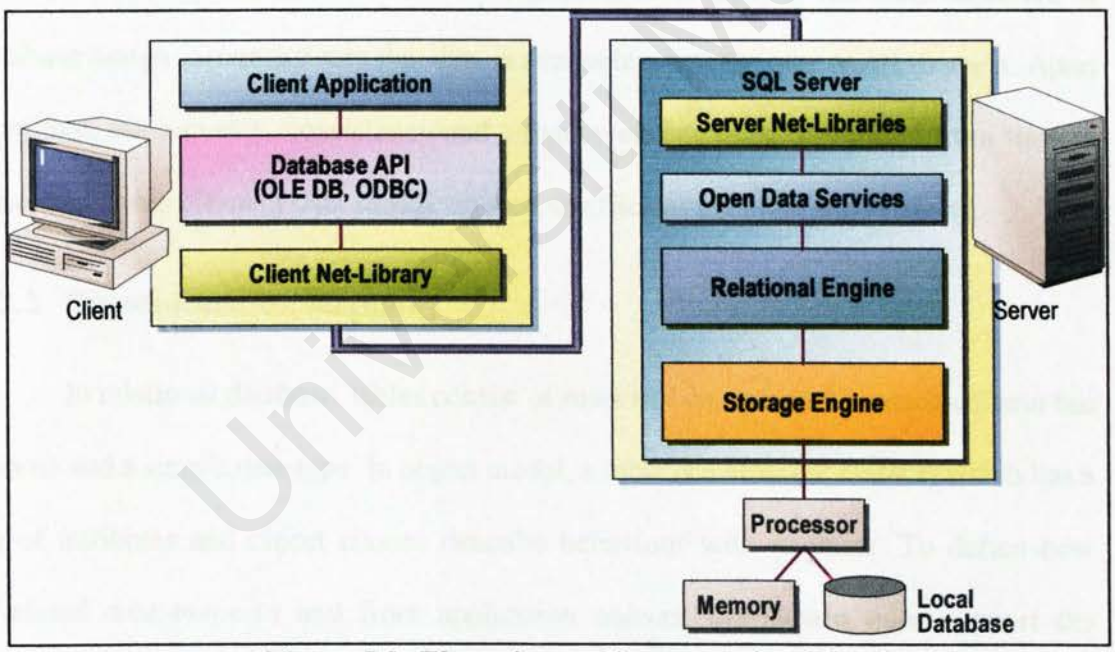


Figure 5.2: Three tier architecture using SQL Server

E-Record Book will be using ASP as the web development tool which acts as the client application to request the pages containing the required data from the SQL Server.

The main purpose of having a three-tier architecture is to assign main functionality to each tier to ensure no function overlapped. Different people could handle each tier using different languages. Therefore, whenever there is error or system fault occurs, the problems can be detected and fixed easily without interfering other tier.

5.3.2 Database Design

Data storage is considered by some to be the heart of an information system (Kendall, 1996). It is a central source of data meant to be shared by many users for a variety of applications. The heart of a database is the OODBMS (object-oriented database management system), which allows the creation, modification and updating of the database; the retrieval of data; and the generation of reports. The main objective of database design is to make sure that data is available when the user wants to use it. Apart from that, the accuracy, consistency and integrity of data must be assured from time to time, to provide efficient data storage as well as efficient updating and retrieval.

5.3.3 Object-Relation Mapping

In relational database, tables consist of rows and columns where each column has a name and a simple data type. In object model, a table is a class (or classes) which has a set of attributes and object classes describe behaviour with methods. To define how relational data maps to and from application objects, the system must support the following mapping capabilities:

- Table-class mapping
- Table(s)-inherited classes mapping

5.3.3.1 Table-class mapping

Table-class mapping is a simple one to one mapping of a table to a class and the mapping of columns in a table to properties in a class.

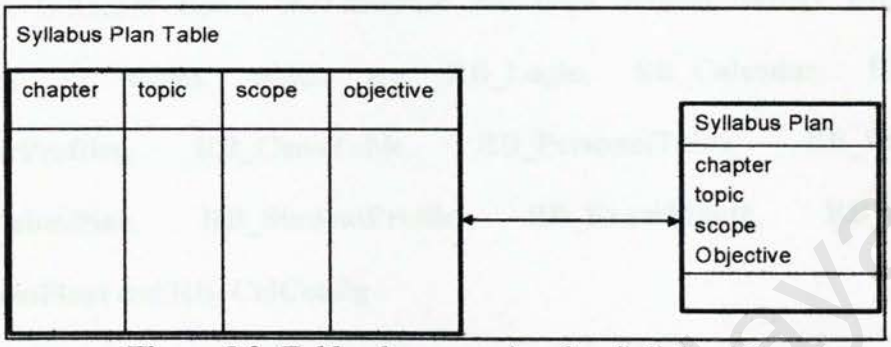


Figure 5.3: Table-class mapping for Syllabus Plan class

5.3.3.2 Tables-inherited classes mapping

In tables-inherited classes mapping, which allows the translation of is-a relationships that exist among tables in the relational schema into a class inheritance relationships in the object model.

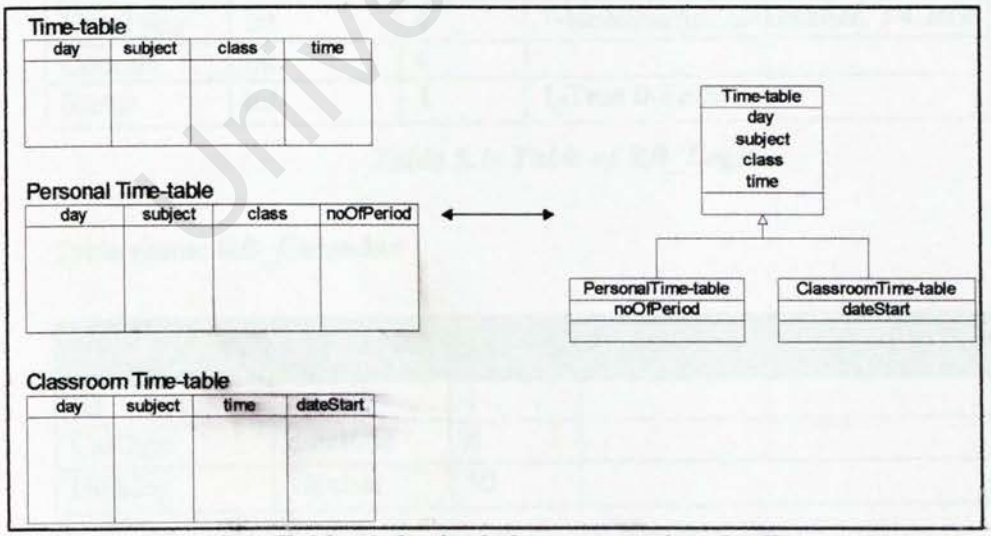


Figure 5.4: Tables-inherited classes mapping for Time-table class

5.3.4 Data Dictionary

Data dictionary can be defined as descriptions of the database structure and contents. Data dictionary defines the field, field type and descriptions of each table.

In E-Record Book, one database had been defined namely **ERBMS** and contained 12 tables, which are **RB_Login**, **RB_Calendar**, **RB_Events**, **RB_UserProfiles**, **RB_ClassTable**, **RB_PersonalTable**, **RB_LessonPlan**, **RB_SyllabusPlan**, **RB_StudentProfile**, **RB_ExamMonth**, **RB_ExamMid**, **RB_ExamFinal** and **RB_CalConfig**

Database Name: **ERBMS**

Table name: **RB_Login**

Field Name	Data Type	Length	Note
No	int	4	Auto Number
UserID	varchar	10	
Name	varchar	50	
Password	varchar	20	
UserType	int	4	1-Headmaster, 2-Teacher, 3-Clerk
Counter	int	4	
Status	int	4	1-True 0-False

Table 5.1: Table of RB_Login

Table name: **RB_Calendar**

Field Name	Data Type	Length	Note
ID	int	4	
CalDate	datetime	8	
Holiday	varchar	50	

Table 5.2: Table of RB_Calendar

Table name: **RB_Events**

Field Name	Data Type	Length	Note
EventID	int	4	
SemesterType	int	4	
EventWeek	int	4	
StartDate	datetime	8	
EndDate	datetime	8	
EventTopic	varchar	300	
EventDetails	varchar	500	

Table 5.3: Table of RB_Events

Table name: **RB_UserProfiles**

Field Name	Data Type	Length	Note
No	int	4	Auto Number
UserID	varchar	10	
StaffID	varchar	10	
Salutation	varchar	10	
Name	varchar	50	
Gender	varchar	10	Male/Female
Race	varchar	10	
ICNum	nvarchar	50	New IC format
DateOfBirth	datetime	8	(mm/dd/yyyy)
Status	varchar	10	Single/Married
PlaceOfBirth	datetime	8	(mm/dd/yyyy)
Email	nvarchar	50	
GredOfPosition	varchar	10	DGx x-number
ContactNo	varchar	10	
DateStartServices	datetime	8	
SalaryRefNo	varchar	12	
IncomeTaxNo	varchar	12	
KWSPNo	varchar	12	
AccLoanNo	varchar	12	
Address	varchar	50	
NameOfSpouse	varchar	50	
JobOfSpouse	varchar	50	

Table 5.4: Table of RB_UserProfiles

Table name: **RB_ClassTable**

Field Name	Data Type	Length	Note
No	int	4	AutoNumber
UserID	varchar	10	
Name	varchar	50	
StaffID	varchar	10	
Class	varchar	50	
Day	int	8	Monday Tuesday Wednesday Thursday Friday
Subject1	varchar	20	
Subject2	varchar	20	
Subject3	varchar	20	
Subject4	varchar	20	
Subject5	varchar	20	
Subject6	varchar	20	
Subject7	varchar	20	
Subject8	varchar	20	
Subject9	varchar	20	

Table 5.5: Table of RB_ClassTable

Table name: **RB_PersonalTable**

Field Name	Data Type	Length	Note
No	int	4	AutoNumber
UserID	varchar	10	
Name	varchar	50	
StaffID	varchar	10	
Class	varchar	50	
Day	int	8	Monday Tuesday Wednesday Thursday Friday
Subject1	varchar	20	

Class1	varchar	10	
Subject2	varchar	20	
Class2	varchar	10	
Subject3	varchar	20	
Class3	varchar	10	
Subject4	varchar	20	
Class4	varchar	10	
Subject5	varchar	20	
Class5	varchar	10	
Subject6	varchar	20	
Class6	varchar	10	
Subject7	varchar	20	
Class7	varchar	10	
Subject8	varchar	20	
Class8	varchar	10	
Subject9	varchar	20	
Class9	varchar	10	

Table 5.6: Table of RB_ClassTable

Table name: **RB_LessonPlan**

Field Name	Data Type	Length	Note
ID	int	4	AutoNumber
UserID	varchar	10	
Name	varchar	40	
DueDate	datetime	8	
DueTime	int	4	
LastModified	datetime	8	
Class	varchar	10	
Subject	nvarchar	50	
Topic	ntext	16	
Content	ntext	16	
Results	ntext	16	
Status	int	4	0-Not validated 1-Validated
Comment	ntext	16	

Table 5.7: Table of RB_LessonPlan

Table name: **RB_SyllabusPlan**

Field Name	Data Type	Length	Note
UserID	varchar	10	
StaffID	varchar	10	
Name	varchar	50	
ID	int	4	
Class	varchar	20	
Subject	varchar	50	
ChapterNo	int	4	
Topic	varchar	50	
Objective	varchar	100	
Contents	varchar	300	
Comments	varchar	500	

Table 5.8: Table of RB_SyllabusPlan

Table name: **RB_StudentProfile**

Field Name	Data Type	Length	Note
No	int	4	AutoNumber
StudID	varchar	10	
StudDateRegister	datetime	8	
StudName	varchar	50	
StudClass	varchar	15	
StudTeacher	varchar	50	
StudGender	varchar	10	Male/Female
StudDateOfBirth	datetime	8	
StudICNum	varchar	14	New IC format
StudParentName	varchar	50	
StudJobParent	varchar	20	
StudAddress	varchar	100	
StudContactNo	varchar	10	

Table 5.9: Table of RB_StudentProfile

Table name: **RB_ExamMonth**

Field Name	Data Type	Length	Note
No	int	4	AutoNumber
StudID	varchar	20	
StudName	varchar	50	
StudClass	varchar	50	
StudTeacher	varchar	50	
Subject1	varchar	50	
Mark1	varchar	3	Range:0-100
Subject2	varchar	50	
Mark2	varchar	3	Range:0-100
Subject3	varchar	50	
Mark3	varchar	3	Range:0-100
Subject4	varchar	50	
Mark4	varchar	3	Range:0-100
Subject5	varchar	50	
Mark5	varchar	3	Range:0-100
Subject6	varchar	50	
Mark6	varchar	3	Range:0-100
Subject7	varchar	50	
Mark7	varchar	3	Range:0-100
Subject8	varchar	50	
Mark8	varchar	3	Range:0-100
Subject9	varchar	50	
Mark9	varchar	3	Range:0-100
Subject10	varchar	50	
Mark10	varchar	3	Range:0-100

Table 5.10: Table of RB_ExamMonth

Table name: **RB_ExamMid**

Field Name	Data Type	Length	Note
No	int	4	AutoNumber
StudID	varchar	20	
StudName	varchar	50	
StudClass	varchar	50	
StudTeacher	varchar	50	
Subject1	varchar	50	

Mark1	varchar	3	Range:0-100
Subject2	varchar	50	
Mark2	varchar	3	Range:0-100
Subject3	varchar	50	
Mark3	varchar	3	Range:0-100
Subject4	varchar	50	
Mark4	varchar	3	Range:0-100
Subject5	varchar	50	
Mark5	varchar	3	Range:0-100
Subject6	varchar	50	
Mark6	varchar	3	Range:0-100
Subject7	varchar	50	
Mark7	varchar	3	Range:0-100
Subject8	varchar	50	
Mark8	varchar	3	Range:0-100
Subject9	varchar	50	
Mark9	varchar	3	Range:0-100
Subject10	varchar	50	
Mark10	varchar	3	Range:0-100

Table 5.11: Table of RB_ExamMid

Table name: **RB_ExamFinal**

Field Name	Data Type	Length	Note
No	int	4	AutoNumber
StudID	varchar	20	
StudName	varchar	50	
StudClass	varchar	50	
StudTeacher	varchar	50	
Subject1	varchar	50	
Mark1	varchar	3	Range:0-100
Subject2	varchar	50	
Mark2	varchar	3	Range:0-100
Subject3	varchar	50	
Mark3	varchar	3	Range:0-100
Subject4	varchar	50	
Mark4	varchar	3	Range:0-100
Subject5	varchar	50	

Mark5	varchar	3	Range:0-100
Subject6	varchar	50	
Mark6	varchar	3	Range:0-100
Subject7	varchar	50	
Mark7	varchar	3	Range:0-100
Subject8	varchar	50	
Mark8	varchar	3	Range:0-100
Subject9	varchar	50	
Mark9	varchar	3	Range:0-100
Subject10	varchar	50	
Mark10	varchar	3	Range:0-100

Table 5.12: Table of RB_ExamFinal

Table name: **RB_CalConfig**

Field Name	Data Type	Length	Note
ID	int	4	AutoNumber
calendar_height	nvarchar	20	
calendar_width	nvarchar	50	
day_abbr	int	4	
Calendar_Title	nvarchar	50	
HeaderRowColor	nvarchar	50	
MonthRowColor	nvarchar	50	
DayCellColor	nvarchar	50	
TodayCellColor	nvarchar	50	
PublicCellColor	nvarchar	50	
HeaderText	nvarchar	50	
MonthText	nvarchar	50	
DayText	nvarchar	50	
PublicText	nvarchar	50	
TodayText	nvarchar	50	
HeaderTextSize	nvarchar	50	
MonthTextSize	nvarchar	50	
DayTextSize	nvarchar	50	
PublicTextSize	nvarchar	50	
TodayTextSize	nvarchar	50	
ShowHeaderText	int	4	
ShowMonthText	int	4	

HighlightPublic	int	4	
HighlightToday	int	4	

Table 5.13: Table of RB_CalConfig

5.4 User Interface Design

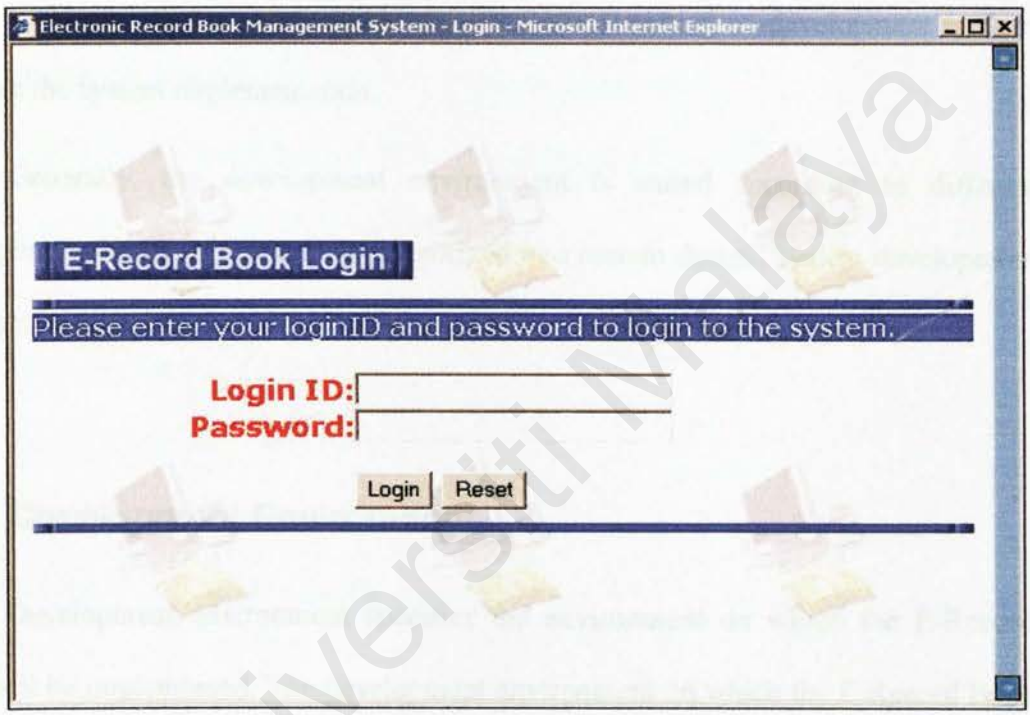


Figure 5.5: Login interface for E-Record Book

Chapter 6 - System Implementation

6.1 Introduction

System implementation in software development is a process to convert system requirements into program codes. The initial stage of system implementation involves setting up the development environment. This includes setting up development tools to facilitate the system implementation.

Generally, the development environment is suited according to different development phases, which can be categorized into system design, system development and report writing process.

6.2 Development Environment

Development environment specifies the environment on which the E-Record Book will be implemented. The development environment on which the E-Record Book is build on is important as it plays an important role in determining the successful implementation of E-Record Book.

6.3 System Design

Although system design is clearly stated in Chapter 5, nevertheless, during the initial stage of system development, a number of considerations and adjustments were

done to the initial system design in order to match the actual needs and requirements.

6.4 System Development

The basic tools used for the system development are:

- Internet Information Service (Web Server)
- Microsoft Windows 2000 Server (Operating System)
- SQL Server 2000 (Database Management System)
- Active Server Pages (ASP) platform
- Microsoft Visual InterDev 6.0
- Notepad and EditPlus (Editor for HTML)
- Swish 2.0 (Image design tools)
- Xara Webstyle 3.0 (Image design tools)
- Adobe Photoshop 7.0 (Image design tools)

6.5 Reports Writing

All the problems encountered, together with solutions found throughout the processes (from system implementation until system evaluation) were recorded as well as result from system testing and system integration.

6.6 System Coding – Coding Approach, Style and Scripting Language

6.6.1 Database Implementation

For E-Record Book, the database is stored in a distributed server in which any data creation, updates or data retrieval will be connected directly to the database server through ActiveX Data Objects (ADO).

The database includes tables to keep users' details including users' authentications information, profiles, creation of records and many others. E-Record Book is an client-server application in which the users can create, edit and delete any records directly into the E-Record Book database.

After the E-Record Book is completed and tested successfully, all the data were flush from the database. All the unnecessary tables were eliminated from E-Record Book database to avoid data overlapping and to reduce workload of the entire system when deployment.

6.6.2 Application Server Configuration

Internet Information Server (IIS) is a Microsoft's offering for Web publishing and web server that allow users of windows NT/2000 to serve web page on the Internet. IIS is available in both Professional and Server version of Windows 2000.

All Web page files should be placed into the default directory of **\Inetpub\wwwroot** and referred the login page named login.asp. System Administrator can choose to create a virtual directory instead of placing all the web page in the default

root folder.

6.6.3 Virtual directory creation

To enable users to access this system from the internet, a virtual directory has to be created on the server. This is done using IIS. The virtual directory corresponds to the actual directory where all the system scripts are found. To enable user browse through E-Record Book, a directory has been also created named **erecordbook**. This directory acts as an alias which is a name that Web browsers use to access that directory. It is more secure to prevent any users from knowing the physical location of the files on the server and cannot use that information to modify the files.

Below is the virtual directory made during development phase:

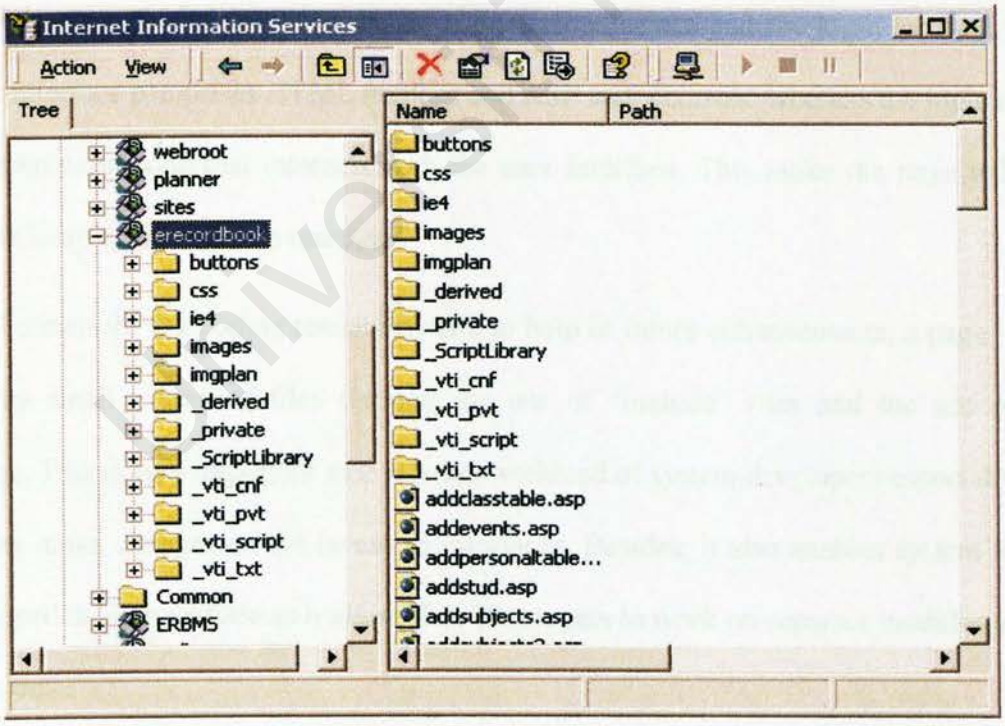


Figure 6.1: Virtual directory for E-Record Book

6.7 Program Implementation

6.7.1 Coding Approach

The methodology used in this development of the E-Record Book is the incremental prototyping methodology. This phase will begin with module design, followed by the implementation of preliminary prototype. On completion of the preliminary prototype, additional functions are added into prototype. This phase is interactive and may require trace backs to previous stages within the incremental prototyping phase if error were found. It ends with the complete implementation of module.

ASP with VB Script and Java Script is used to develop the entire E-Record Book. Forms are divided into two sections: the user interface and the logic (or code). The user interface comprises HTML markup and ASP web controls, whereas the logic is the programmatic code that interacts with the user interface. This make the page will look more simple and easier to manage.

To increase the coding readability and to help in future enhancements, a page is formed by small pieces of files through the use of "include" files and the use of comments. This is very important as it reduces workload of system developers especially when they make changes on the layout of interfaces. Besides, it also enables system to be developed in shortest time as it allows few developers to work on separate modules at the same time.

6.7.2 ASP page

Microsoft Active Server Pages (ASP) is a server-side scripting environment that can use to create and run dynamic, interactive Web server applications. With ASP, HTML pages, script commands, and COM components can be combined to create interactive Web pages or powerful Web-based applications, which are easy to develop and modify. Figure below indicated an ASP page (checklogin.asp) with a few built-in objects (highlighted in red) that make it easier to gather information sent with a browser request, to respond to the browser and to store information about a particular user, such as user-selected preferences.

```
<%@ Language=VBScript %>
<html>
<head><meta http-equiv="Refresh" content="2; URL=login.asp"></head>
<body bgcolor="aqua" leftMargin="0" background="images/ebook.jpg" topMargin="0" marginheight="0"
marginwidth="0">
<%
Dim strUsername, strPassword
strUserID = Request("loginID")
strPassword = Request("psword")

'reference to ADO Connection
objConn = Session("conn")

'declare variables
Dim rsUsers , rsType , rsCounter , rsStatus , rsName

'create ADO Recordset
Set rsUsers = Server.CreateObject("ADODB.Recordset")
rsUsers.ActiveConnection = objConn
<!-- more codes were not shown here
     Please refer to appendixes -->
```

Figure 6.2: checklogin.asp of E-Record Book

Chapter 7 - System Testing

The main function of testing is to establish the presence of defects in a program and to judge whether the program is usable in real application. Nevertheless, testing can only demonstrate the presence of errors. It cannot show that there is no error in the program. Therefore, a more suitable approach must be chosen to reduce the possibility of errors in a program.

Bottom-up approach is adopted in system testing for E-Record Book. Each module at the lowest level of the system hierarchy is tested individually. Then, all the tested modules would be related to the next module testing. This approach is repeated until all the modules are tested successfully.

7.1 Types of Testing

In general, the testing process of E-Record Book can be shown in the following figure. All the details will be further explained in subsequent sub-sections.

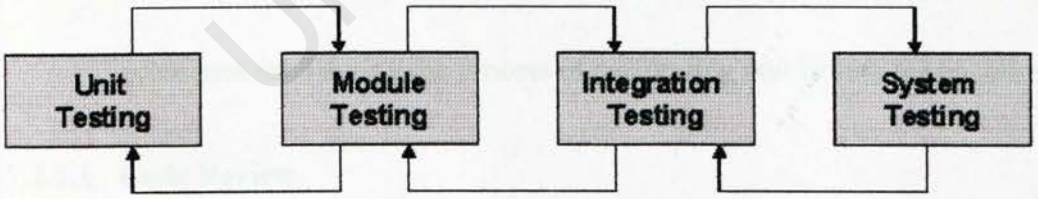


Figure 7.1: Testing Process

7.1.1 Unit Testing

Unit test is the process to test the individual component to ensure that they function properly. Each component is tested independently without the interference from other system components. Unit test is performed concurrently with the development process.

The codes in the components used in the E-Record Book are analyzed to determine the various states that the components will encounter. The test involves running the components in the browser and trying to identify the source of error from the error messages printed on the screen. After all syntax errors are eliminated, test cases are developed to test the codes. Input data is provided to the component in the system and the output is verifying its correctness. Test cases are developed carefully in order to capture the different behaviors of the component. Testing are also involves boundary testing. Other than that, testing also checks to ensure that data from the database is being indexed correctly as the database server is located on a different machine. For example, the *adduser.asp* in E-Record Book was written and it was tested alone based on its functionality and requirements.

Techniques used during the process of performing unit testing are as follows:

7.1.1.1 Code Review

Before a source code is deploying, codes are reviewed line by line to discover any syntax error as well as semantic error. If errors are discovered, they are corrected immediately. Input is typed in and the output is verified for accuracy. This is done by

double checking manually to verify that the query results yield records that exist in the repository and that the users does have the rights to view the records.

7.1.1.2 Tracing

This method is faster compared to code review techniques and it is efficient in discovering errors. During the compilation, the VB compiler will detect type of errors in a program and display the error type as well as the line number in which the error occurs. In order to debug the error, the line of number was traced and correction was made instantly.

7.1.1.3 Test Cases

To test the selected component in E-Record Book, input data and conditions were chosen to allow the component to manipulate the data and the output could be observed. It is important to perform test cases in a convincing way so that the test data will exhibit all possible behaviors.

7.1.1.4 Other Techniques

Other techniques are debugging and proofing correct code behind to discover any error during the development. This method able to subject the code in a more structured way to establish its correctness.

7.1.2 Module Testing

Module testing is performed without other system modules. A module consists of a collection of dependent components to perform a particular task or function. Different

possible test cases are applied to the module and the test results would be verified. Unusual results will be analyzed and they would help in debugging sub-modules in order to produce the desired output.

For example, E-Record Book includes ASP files such as *adduser.asp*, *removeuser.asp*, *unlockuser.asp* to perform the functions in user profiles module.

7.1.3 Integration Test

Integration test is needed when all modules are integrated. The main focus in integration test is to navigate the interfaces repeatedly to detect any interface mismatch problem.

Several important aspects are checked to ensure that the flow of the data in E-Record Book is well organized and are user friendly to all the system users. E-Record Book uses a bottom up testing. Each module is tested for its ability to function after integration. The flow of information from one module to another is verified for accuracy.

In this case, several modules in E-Record Book such as Login Module, User Profiles Module, Timetable Module, Syllabus Plan Module and others were tested for integration.

7.1.4 System Testing

The sub-systems are integrated to make the entire system. Therefore, the main purpose in system testing is to find errors that result from unanticipated interactions between sub-systems. Besides, it is used to validate whether the system meets its

functional and non-functional requirement.

Problems might occur by the time the new developed system is integrated. The test covers the performances, reliability, accuracy and other criteria. Testing is carried in the manner as though the system is in use.

For example, all the integrated modules in E-Record Book were tested as a whole to check for its accuracy, consistency, functionality and reliability.

Chapter 8 - Evaluation and Conclusion

Evaluation is the ultimate phase of developing a system and an important phase before delivery the system to the end users. Evaluation was related to user environment, attitudes, information priorities and several other concerns that are to be considered carefully before effectiveness can be concluded. At all phases of the system approaches, evaluation is a process that occurs continuously, drawing on a variety of sources and information.

8.1 Problems Encountered

8.1.1 Setting up server

Many problems faced in setting up the relevant servers in the E-Record Book. This problem is due to lack of experiences in dealing with the servers. Many problems are also faced with Microsoft SQL Server 2000 which requires a different set of installation procedures compared to IIS.

8.1.2 Interconnecting Server

Problems are faced trying to interconnect the IIS and MS SQL Server 2000 . The IIS is unable to detect the database server even though connection string is written properly. The database server must be configured based on login authentication for selected user with the chosen password due to security measurement.

8.1.3 Debugging ASP Pages

During the development phase of E-Record Book, I was a first time user in using Microsoft Visual InterDev 6.0. I found it hard to debug the written ASP codes. This is because the debugger for ASP pages was not enabled. At early stage, any error on ASP codes can only be detected when browsed in web browser.

8.1.4 Evaluation by End Users

Due to insufficient time, the developed E-Record Book system did not managed to be evaluated by teachers in SMK Sri Rahmat, Johor Bharu. Therefore, valuable feedbacks were not received although the testing has been done thoroughly in development site.

8.2 Solutions

8.2.1 Setting up Server

Information about setting up IIS and MS SQL Server 2000 were obtained from Internet, books and friends in order to find out the best way to set up the servers. Trial and error was performed during the setup of servers but finally they have been successfully setup.

8.2.2 Interconnecting Server

The problem of interconnecting the IIS server and the database server is solved through setting-up trust relationships among the related server. This was done by

providing the correct type of provider, user id and password, the source of database and the persistent security level in the *global.asa* file.

8.2.3 Enabling Debugging for ASP Pages

In Ms Visual InterDev 6.0, the selected ASP page can be debugged if the page was in a "Set As Start Page" mode. Another method to easily debug ASP pages is to check the "Debugging Flags" in Application Debugging under the Application Configuration of IIS 5.0 Errors found in ASP pages were easily shown in a highlighted in yellow under the Microsoft Development Environment.

8.2.4 Improving Existing System

Certain modules in E-Record Book were improved to provide the same functionality as in the manual record book in real life. Feedbacks were received from FSKTM lecturers in order to develop a more manageable E-Record Book.

8.3 Strength

8.3.1 Wide-accessibility

The E-Record Book is a client server application. It also acts as an Web based application which has provided wide-accessibility to users where users can access from everywhere in the world. Client-side software only requires the installation of a web browser to access the E-Record Book. Furthermore, browsers are available across all platforms.

8.3.2 Confidentially and Integrity of Information

The strength of the handling the records of teachers and students depends on the access control of users in the system. For example, information on syllabus plan and lesson plan from teachers cannot be accessed by clerks. This feature protects confidentially of information and also maintains the integrity of the information in the record book.

8.3.3 Better Monitoring and Validation

The capability for headmaster in school to monitor and validate the lesson plan for every teacher can create a better management of record book. The headmaster can also gives any comment based on the written syllabus plan and daily lesson plan from teachers. This has made the task of monitoring and validating becomes more easier and effective.

8.3.4 More Informative Record Book

E-Record Book provides school's objectives, vows, organization chart and events to headmaster, teachers and clerks. They can use the given information to learn more about the school and keep themselves updated with the latest events in school. Clerks or school administrator able to perform the updating tasks regarding the events and public holidays in school in order to provide information to headmaster and teachers.

8.3.5 Better Management of Record Book

E-Record Book was designed to improve the existing manual record book. Therefore, tasks such as adding, updating and deleting students profiles and subjects were given to school administrator where as planning daily lesson plan, syllabus plan, time table, updating students examination records were handled by teachers. As for headmaster, he or she can only monitor and validate the certain records from teachers. This has created a more manageable record book because every level of users has its own roles and access control.

8.4 Limitation

8.4.1 Platform

The E-Record Book is limited to certain platforms in term of openness. It supports Window 95, Window 98, Window NT, Window 2000 and Internet Explorer 4 or above. Besides, it needs Ms SQL Server 2000 as the database server to manage the data of the system.

8.4.2 Language Support

The E-Record Book only provides the information about the vows and school objectives in Malay version.. It is due to information given from the present school was in Malay language. Translation could not be made as the terms of some words are ambiguous and may be different from original meaning once they have been translated.

8.4.3 Handing in Lesson Plan

E-Record Book did not manage to provide headmaster with the capability to set a dateline on handing in the daily lesson plan for validation. Therefore it is expected for teachers to write their lesson plan accordingly to the schedule.

8.4.4 Calculating Marks and Grades

Teachers using E-Record Book only manage to update their students examination records based on monthly, mid term and final term. The calculation of passes and fails, type of grades were not generated due to limitation of time in developing E-Record Book.

8.5 Future Enhancement

As mentioned before, E-Record Book is still not fine enough to work at its full efficiency. Some refining work needs to be done to the system to increase its usability and reliability. The aspects to be refine and some suggestions to upgrade the system are as below:

8.5.1 Setting up Dateline of Lesson Plan

Headmaster's capability on setting up dateline of handing in lesson plan for teachers can be added to make sure that teachers update their daily lesson plan and submit them on time. As for teacher who is absent on the certain lesson, he or she can post a notification in the lesson plan to inform the headmaster about it.

8.5.2 Generating Grades and Reports

In order to make the E-Record Book more informative, grades can be generated based on the marks given for each student's subject. Therefore, report of performance based on that particular student can be made viewable in the system.

8.5.3 Forums and Announcements

Forums and announcements can be added into system so that headmaster, teachers and clerks can discuss on certain topic or post any new announcement in school. It will act as another discussion site besides meeting for the users to view their opinions and comments.

8.5.4 Upload and Download Examination Papers

E-Record Book can provides the capability for teachers to upload their recent examination paper for the references of other teachers. Therefore they can exchange the examination paper among themselves by selecting the required paper. This can be done by downloading the files into own storage device.

8.5.5 MultiLanguage

Optional language such as English or Malay can be selected when user login into the system. This will make the contents of E-Record Book more standardized and easy to understand.

8.6 Other features

- Online help given for users to perform their tasks in E-Record Book.
- Data created in the E-Record Book are interrelated to each other. The system will give warning to the user if the users try to delete some data that have linking to other data from other table.
- Clerks able to create packages of subjects for students
- Activities performed by users will be recorded in log files to track their performances when using the system.
- Strategies on teaching subjects were included in the system as a reference for teachers.

Appendix A – Installation and Configuration

A.1 Installation & Setup of IIS 5.0

Microsoft Internet Information Services (IIS 5.0) is available on the Windows 2000 Professional, Server, Advanced Server, Windows XP and Windows NT platform. E-Record Book is developed under the Windows 2000 Advanced Server platform. This is due to the requirements of the E-Record Book database server which is SQL 2000 Server and it only supports Windows 2000 Server series.

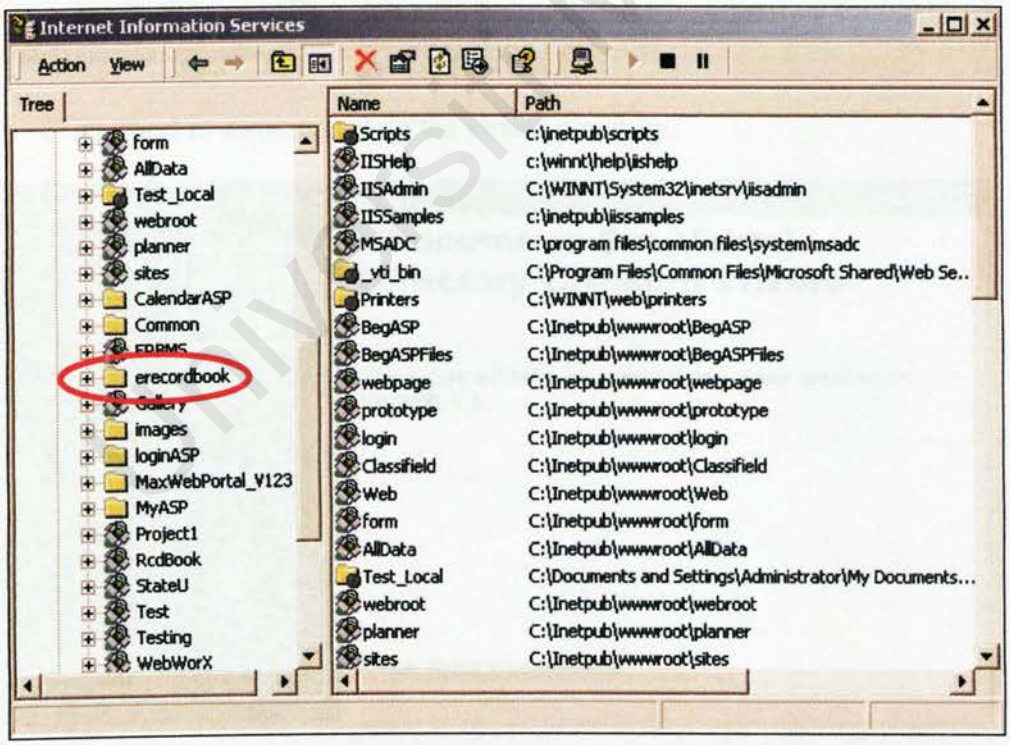
For development phase, it is important for E-Record Book to have the IIS 5.0 installed in the Windows Advanced Server platform. To install the IIS, please follow the steps below:

A.1.1 Installation of IIS 5.0

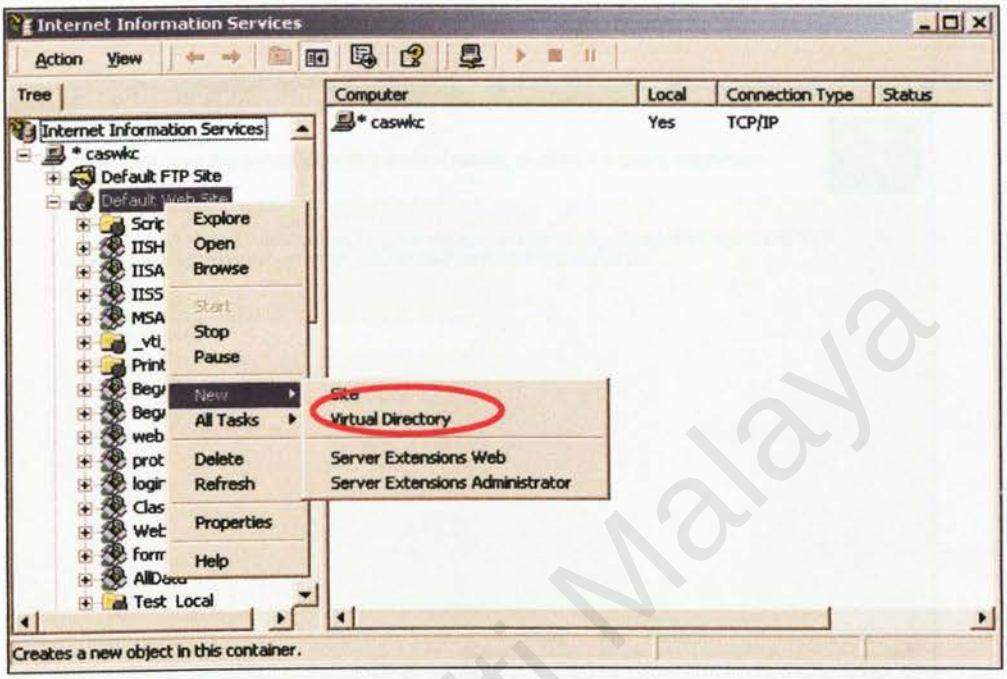
1. First, go to **Start -> Setting -> Control Panel**. In Control Panel window, double click on the **Add/Remove Programs**.
2. In the **Add/Remove Programs** menu, select the **Add/Remove Windows Components**. When the window of **Windows Components Wizard** pop-up, scroll down for the **Internet Information Services (IIS)** and make sure to check the checkbox beside it. Then click the Next button to proceed
3. After the IIS has been successfully installed, click Finish to close the **Windows Components Wizard**.

A.1.2 Setting up E-Record Book virtual directory

1. Before the E-Record Book website was being set up, copy the **erecordbook** folder into the **\InetPub\wwwroot** at the root directory (example: C:\InetPub\wwwroot). The **erecordbook** folder which includes all the application files has been created manually in the root directory.
2. To create a virtual directory for E-Record Book, go to **Start -> Programs -> Administrative Tools -> Internet Services Manager**
3. In the **Internet Services Manager** window, the erecordbook folder will be seen. But a virtual directory needs to be created before the contents of E-Record Book can be browsed through Internet browser.



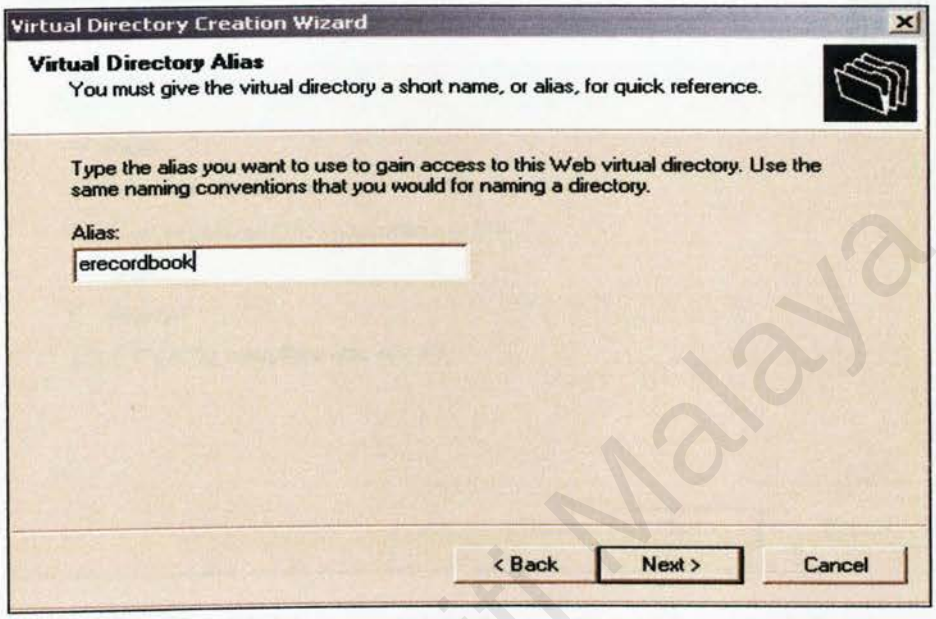
4. To start creating the virtual directory, right click on **Default Web Site** and click on **New -> Virtual Directory**



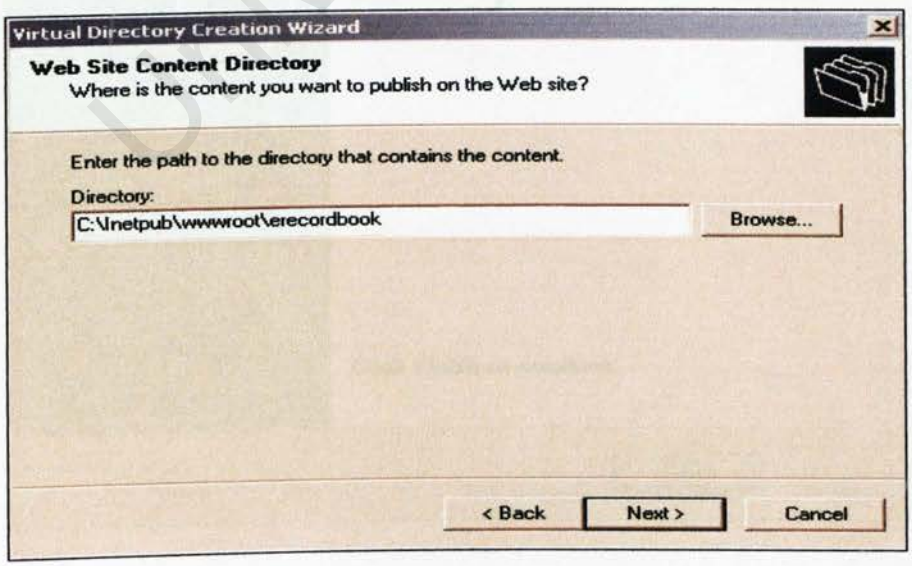
5. Click **Next** to continue set up the virtual directory.

6. Specify the **Virtual Directory Alias** of the website URL which will be used by users to access E-Record Book. It will be named **erecordbook.**

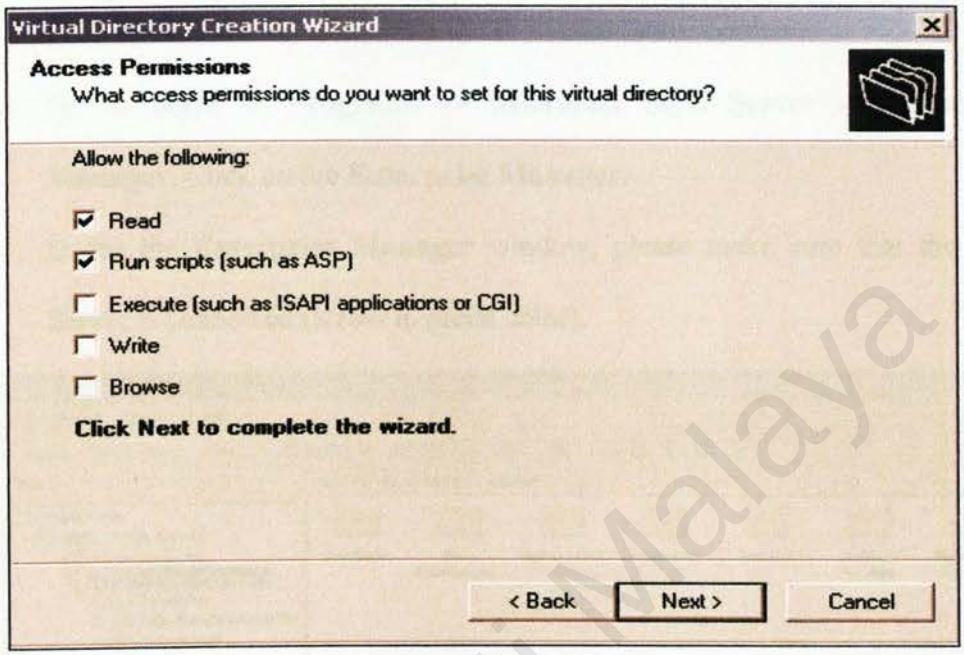
   Then, click Next to proceed.



7. Browse for erecordbook folder which has been created earlier in the wwwroot folder. (Example: C:\InetPub\wwwroot\erecordbook). Then, click Next.



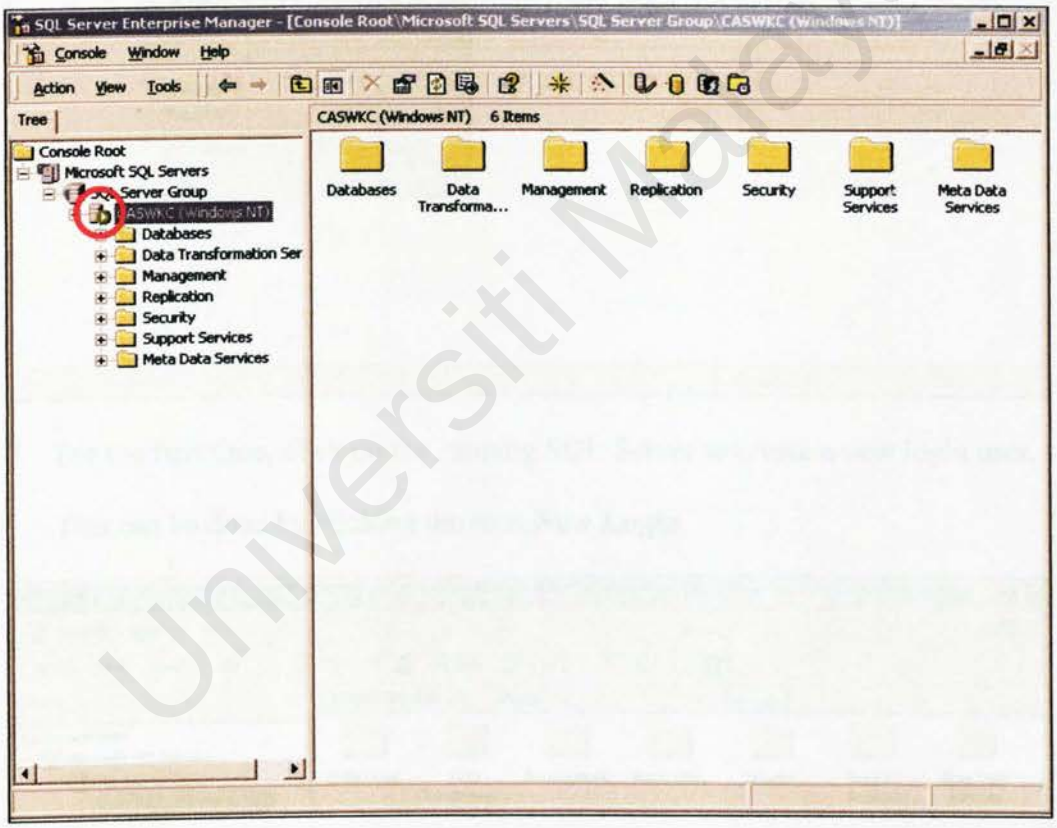
8. Make sure that the **read** and **run scripts** options are checked. Then, click **Next** to complete the wizard.



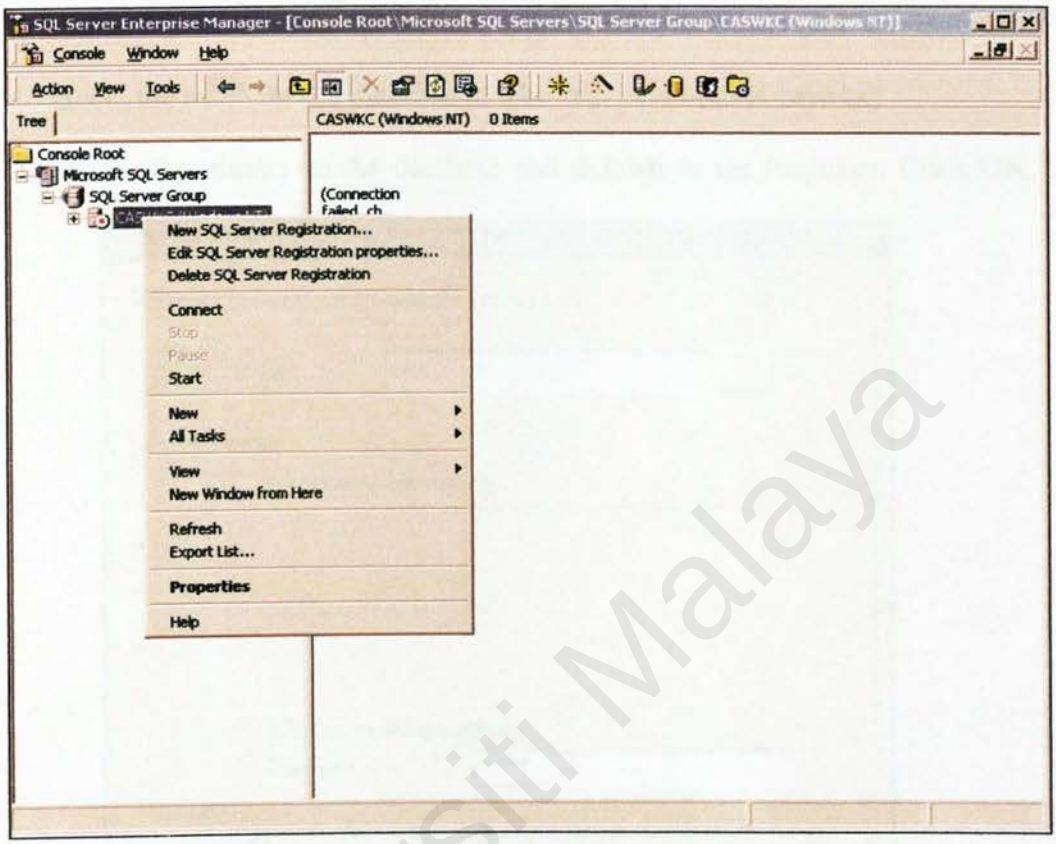
9. Finally, click **Finish** to complete the creation of the virtual directory.

A.1.4 Setting up E-Record Book's SQL Database

1. Please make sure that the Microsoft SQL Server 2000 Enterprise Edition has been installed before setting up the E-Record Book database.
2. Go to **Start -> Programs -> Microsoft SQL Server -> Enterprise Manager**. Click on the **Enterprise Manager**.
3. Under the **Enterprise Manager** window, please make sure that the SQL Server is connected (arrow in green color).

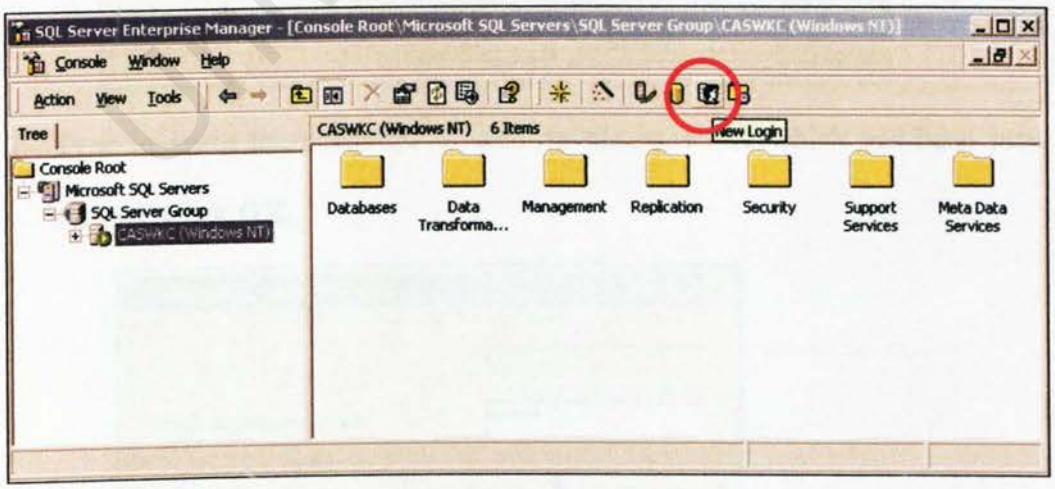


4. If not connected, right click the server and click **Connect**.



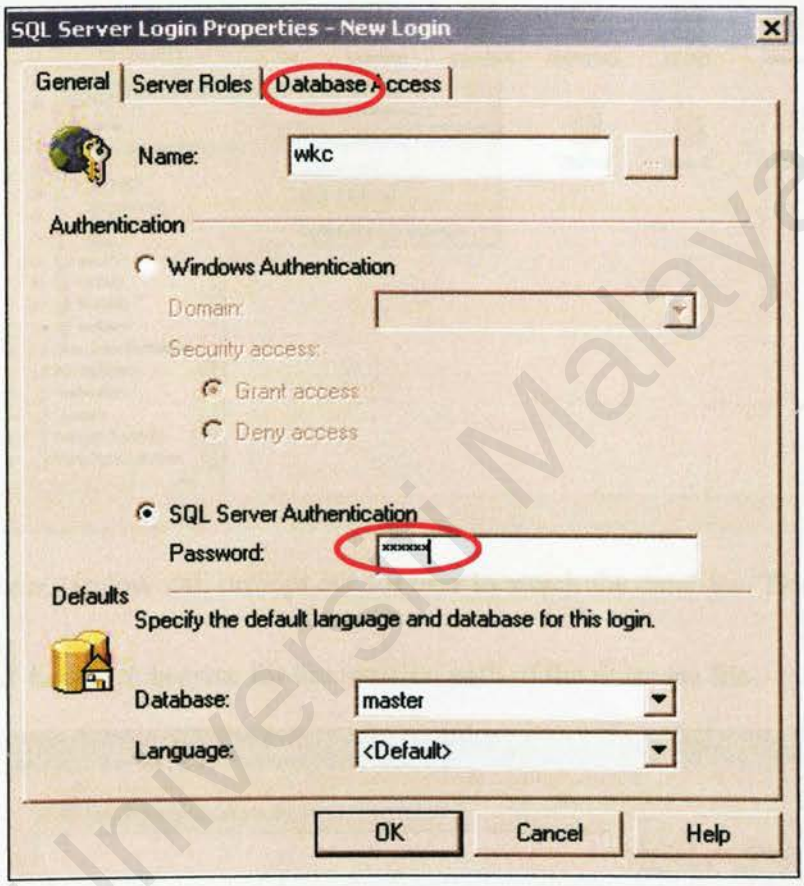
5. For the first time, click on the running SQL Server to create a new login user.

   This can be done by clicking the icon **New Login**.

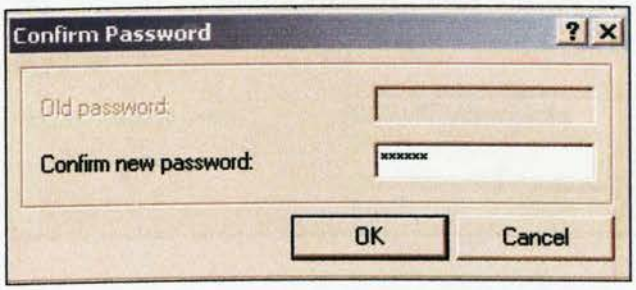


6. A new login window will pop up. Then type in the user name. Select the SQL Server Authentication option and type in the password. (For the E-Record Book, the name for the database is **wkc** and password is **caswkc**)

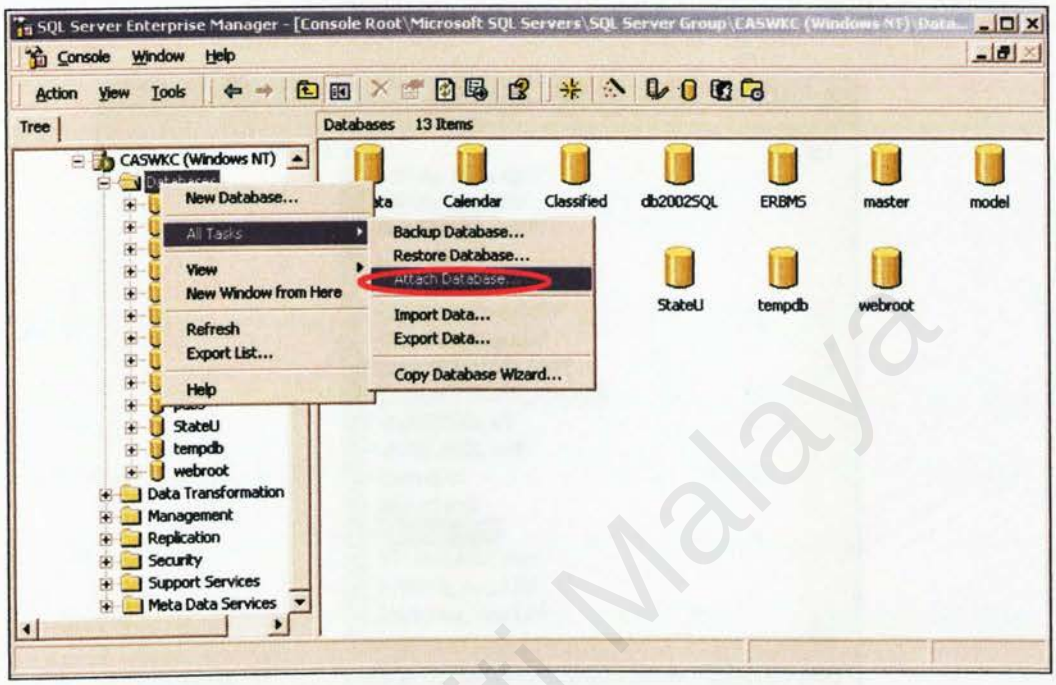
Choose the **master** as the database and **default** as the language. Click **OK**.



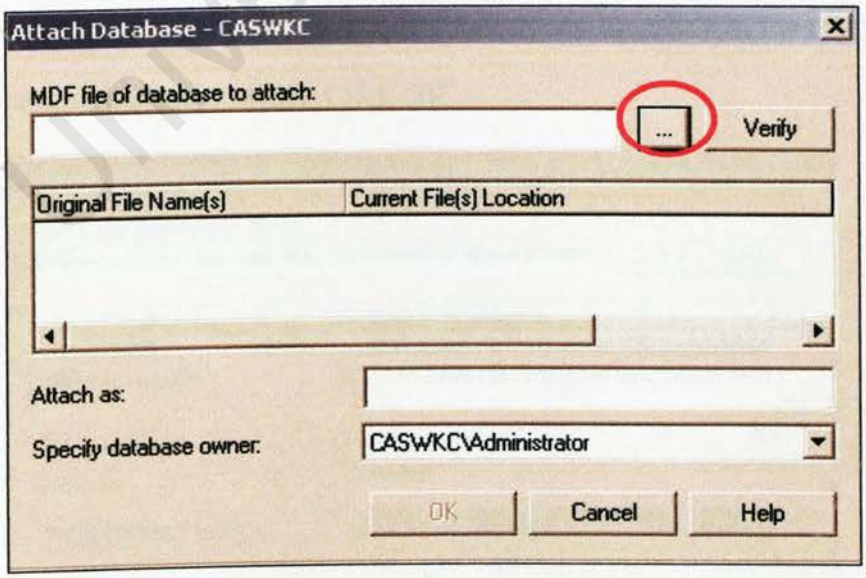
7. Type in again the password to confirm the password which has been just entered. Click **OK**.



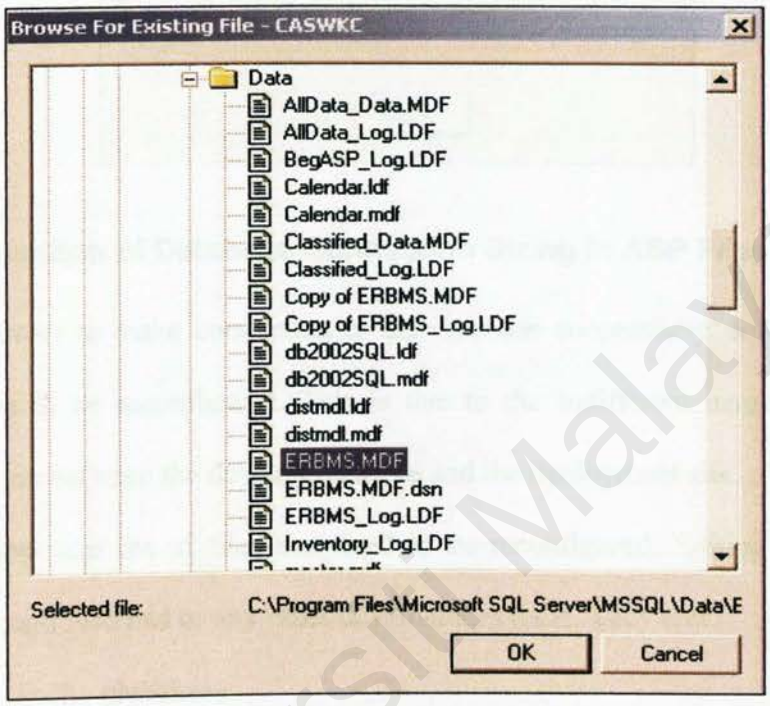
8. To set up the database, click on the **Databases** folder in the server. Then right click it and go to **All Tasks -> Attach Database..**



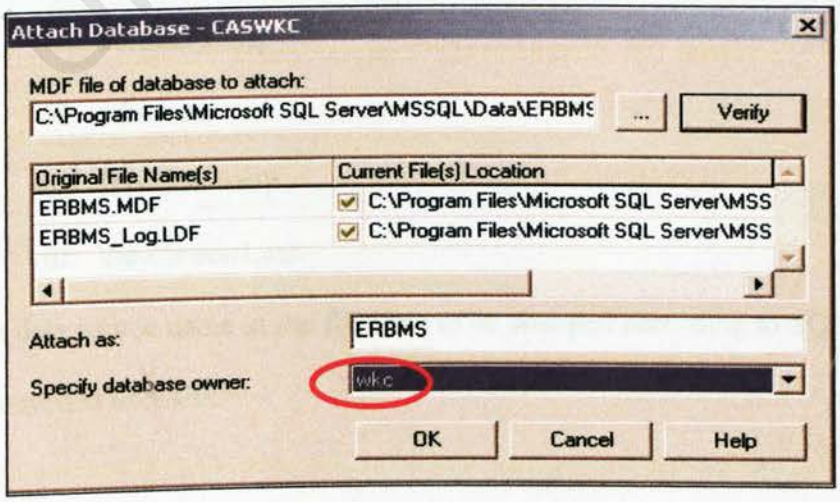
9. A new window will prompt out for user to attach the database. Then click the ... button to browse for the location path of the database file.



10. Browse for the path of the database file named **ERBMS.mdf** (For example, the path is C:\ Program Files\ Microsoft SQL Server\ MSSQL\ Data\ ERBMS.mdf). Then click **OK** to attach the database.



11. The **Attach Database** window will show the name of the database attached and its location. Then specify the database owner by selecting the user name that has been created earlier. Click **OK**.



12. After the database has been successfully attached, a message will prompt out. Then click **OK**.

A.1.4 Configuration of Database Connection String in ASP Files

1. In order to make connection to the database successfully, a few ASP files have to be reconfigured. This is due to the indifferent name of the SQL Server between the development site and the deployment site.
2. Below is a list of files that need to be reconfigured. Editing can be done through Notepad or any other development tools. They are:
    i. global.asa
    ii. calendar.asp
    iii. calendar1.asp
    iv. callesson.asp
    v. editlesson.asp
    vi. editor.asp
    vii. lessonplan.asp
    viii. viewlesson1.asp
3. The data source name in the files has to be changed according to SQL Server deployed at user site.

Appendix B – User Manual Guide

B.1 User Manual

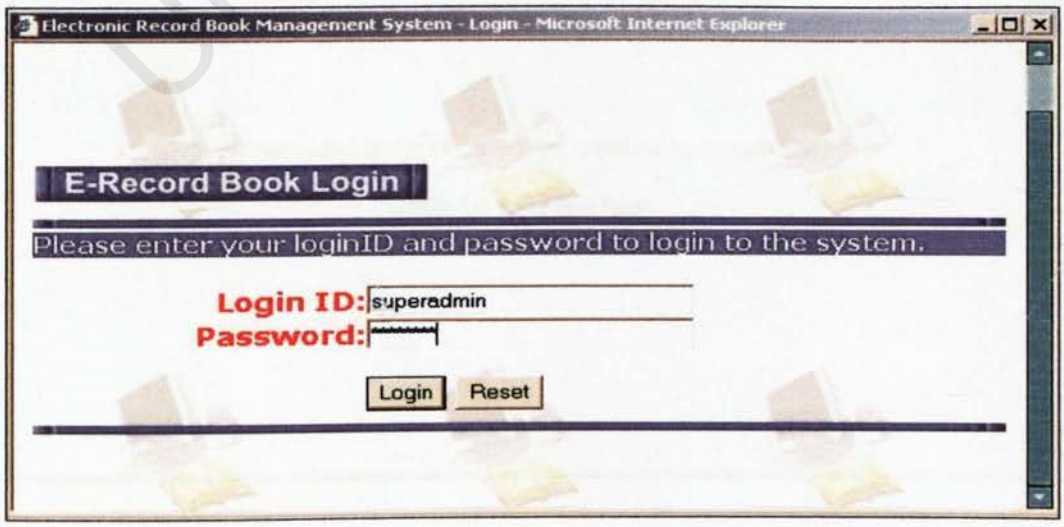
E-Record Book is an online system that helps to reduce the headmaster, teachers and clerks workload in school. This manual provides as a guide to help users to use E-Record Book more effectively to achieve the stated goal.

This manual is divided mainly into three main parts, which are School Administrator Section, Teachers Section and Headmaster Section.

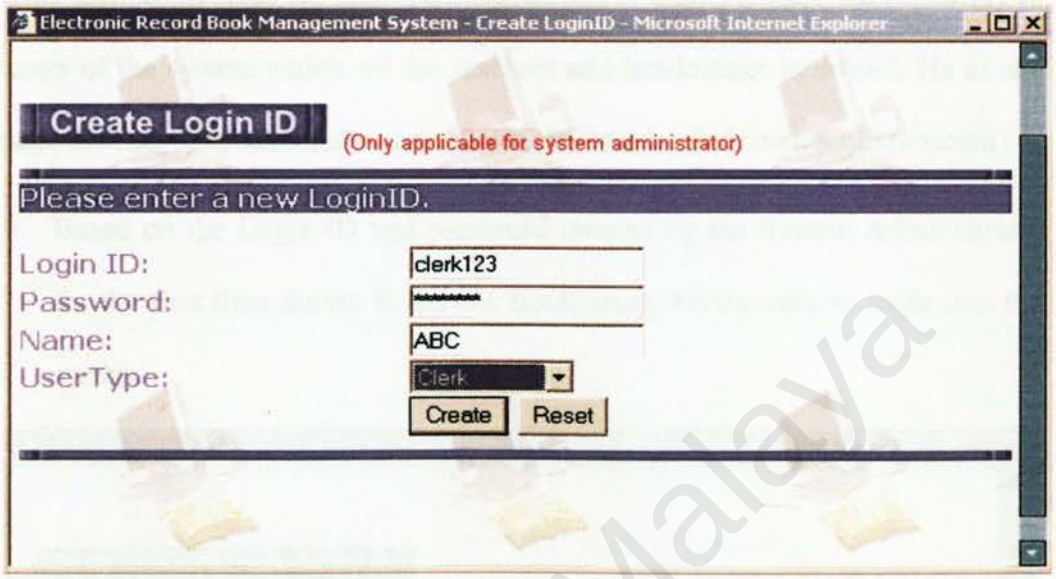
B.1.1 System Administrator Section

In order to start using the system, the System Administrator must create a Login ID for clerk in school to provide access for other users to use the system. Therefore he must login into the system and enter the create user module.

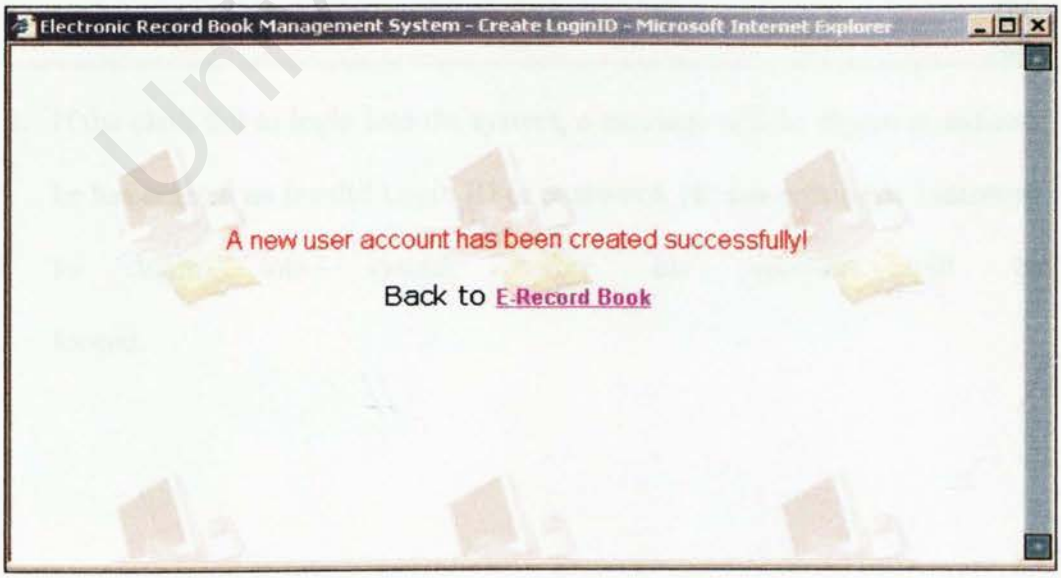
1. System Administrator needs to enter his Login ID and password which have been created during the installation of the E-Record Book.



2. After the System Administrator successfully login, he can create new Login ID for clerk in school.



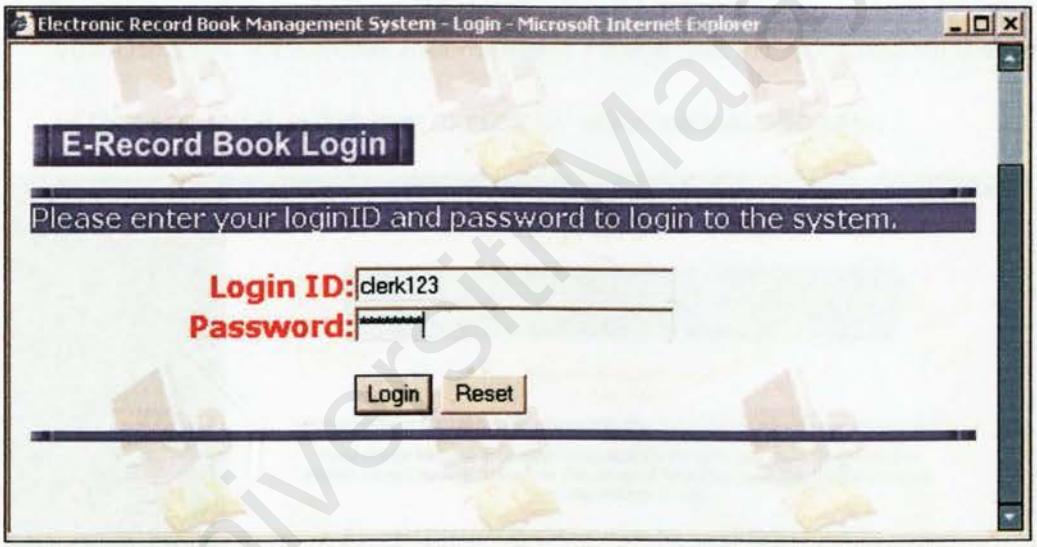
3. Once the Login ID for the clerk has been successfully created, a new window will prompt out to inform that a new login has been created. (Only system administrator has this privilege to create new user for the first time of the setup)



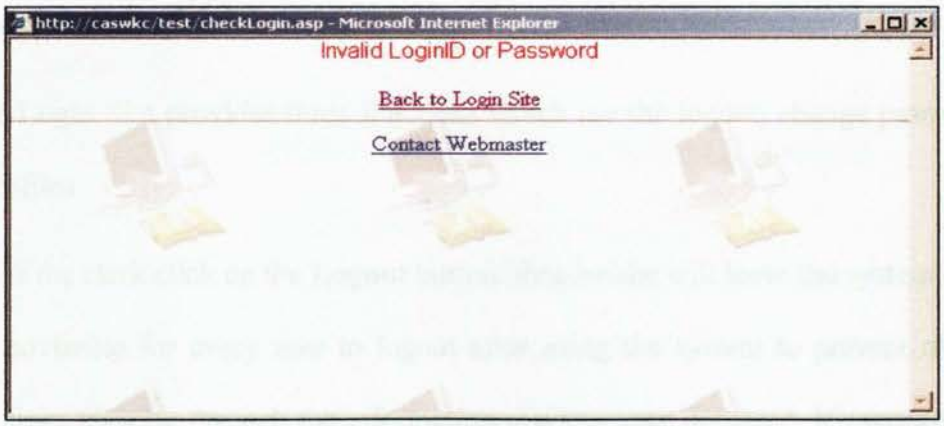
B.1.2 School Administrator Section

This section involves the school administrators or clerks to have the capability to handle users of the system which are the teachers and headmaster in school. He or she can manage the user site, school site and student site besides their own login (account).

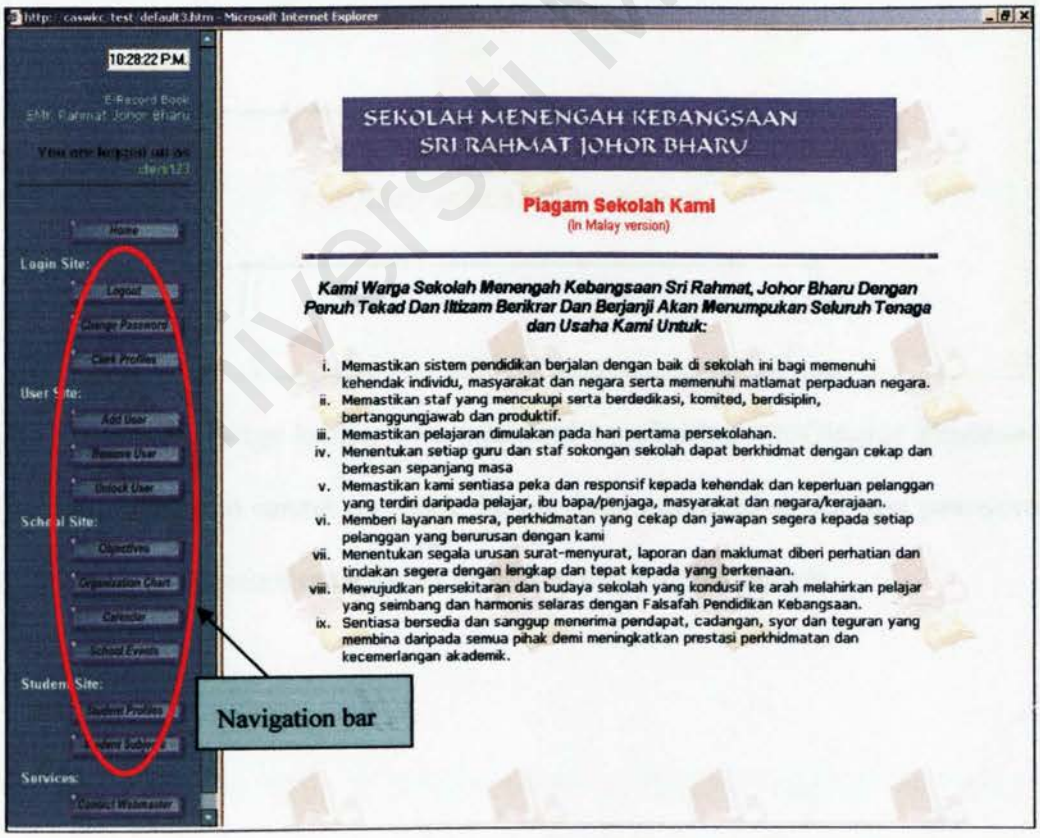
1. Based on the Login ID and password created by the System Administrator for the first time during E-Record Book setup, he/she able to login into the system.



2. If the clerk fail to login into the system, a message will be shown to indicate he has entered an invalid Login ID or password. He can only have 3 attempts to login into system before his account will be locked.



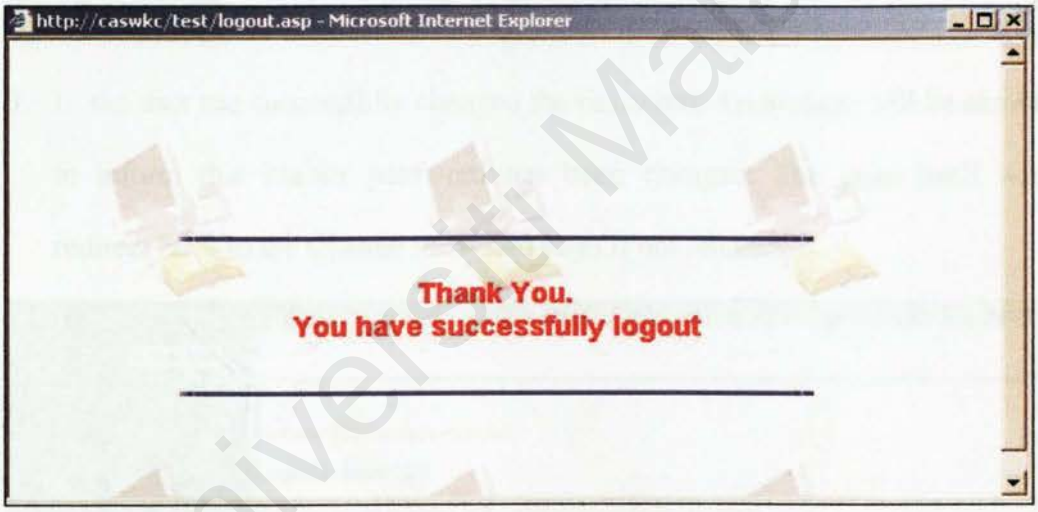
3. Once the clerk has successfully login into the system, he enters the home or default page in E-Record Book which is the 'Piagam Sekolah Kami' (School vows of SMK Sri Rahmat Johor Bharu). Navigation bar is located on the left of the page and it assists user to click to other module with ease.



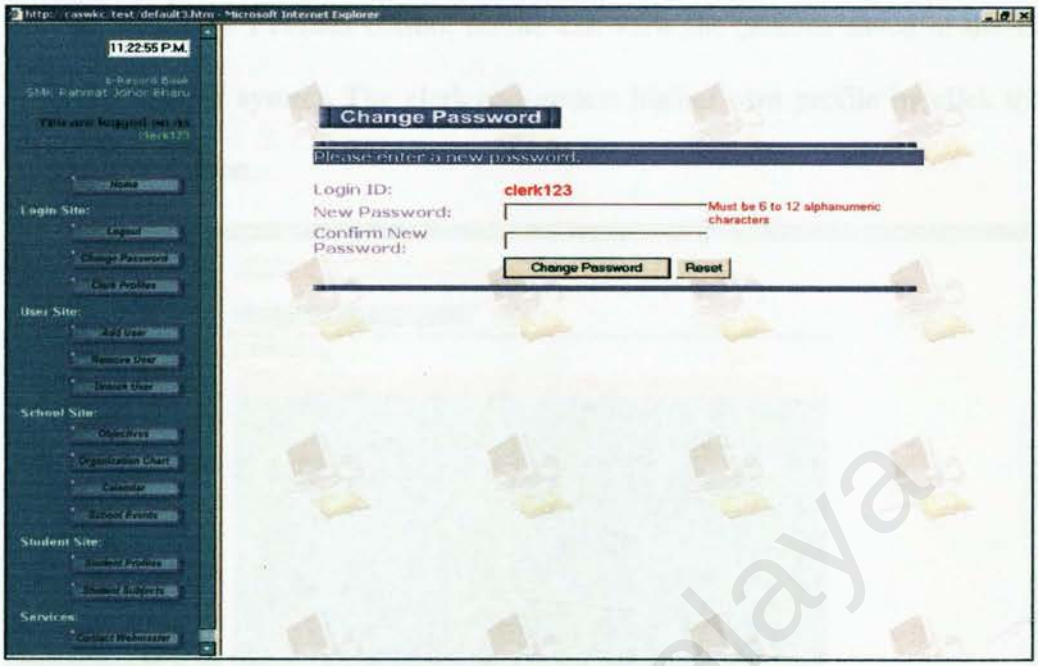
B.1.2.1 Login Site

The Login Site provides three functions which are the logout, change password and user profiles.

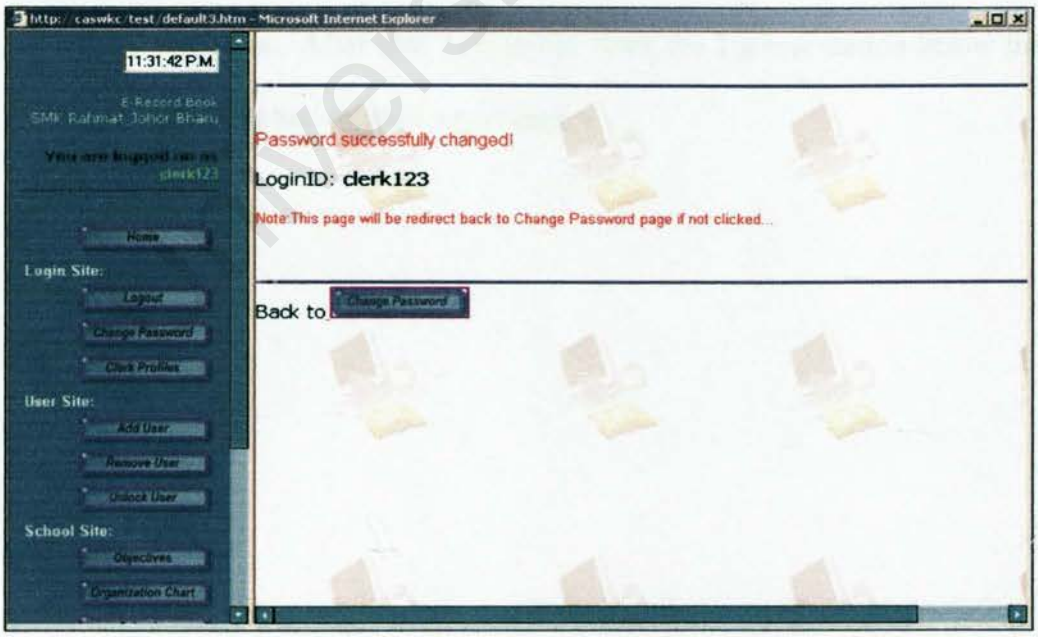
1. If the clerk click on the **Logout** button, then he/she will leave the system. It is advisable for every user to logout after using the system to prevent others from viewing through the information that has been accessed. If successfully logout, a scrren with the message "Thank you. You have successfully logout" will prompt out.



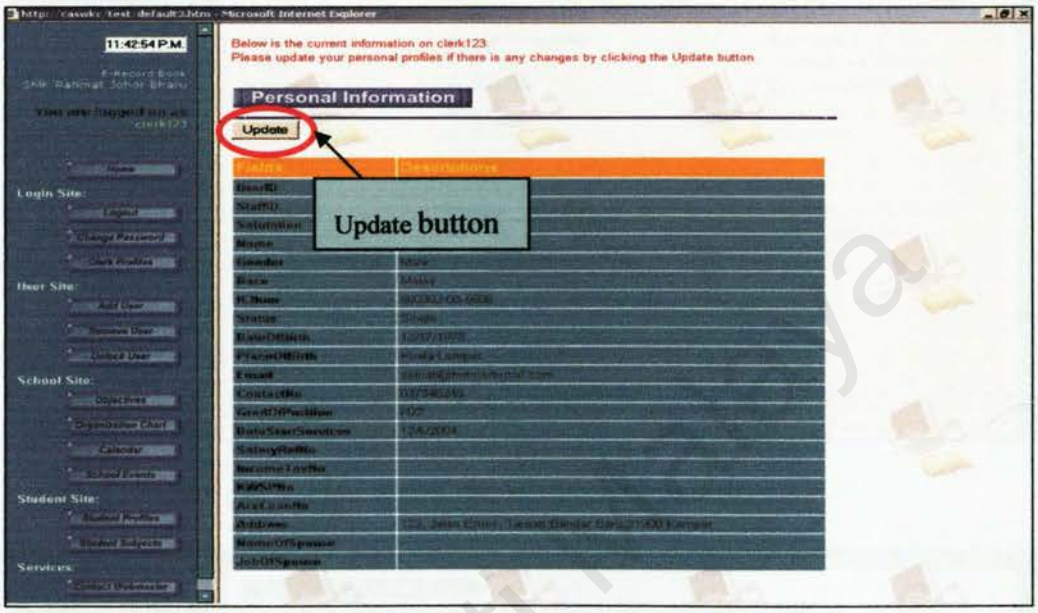
2. Clerk can change his/her own password by clicking the **Change Password** button. He/she cannot use back the previous password as the new password. The new password entered has to be in 6 to 12 alphanumeric.



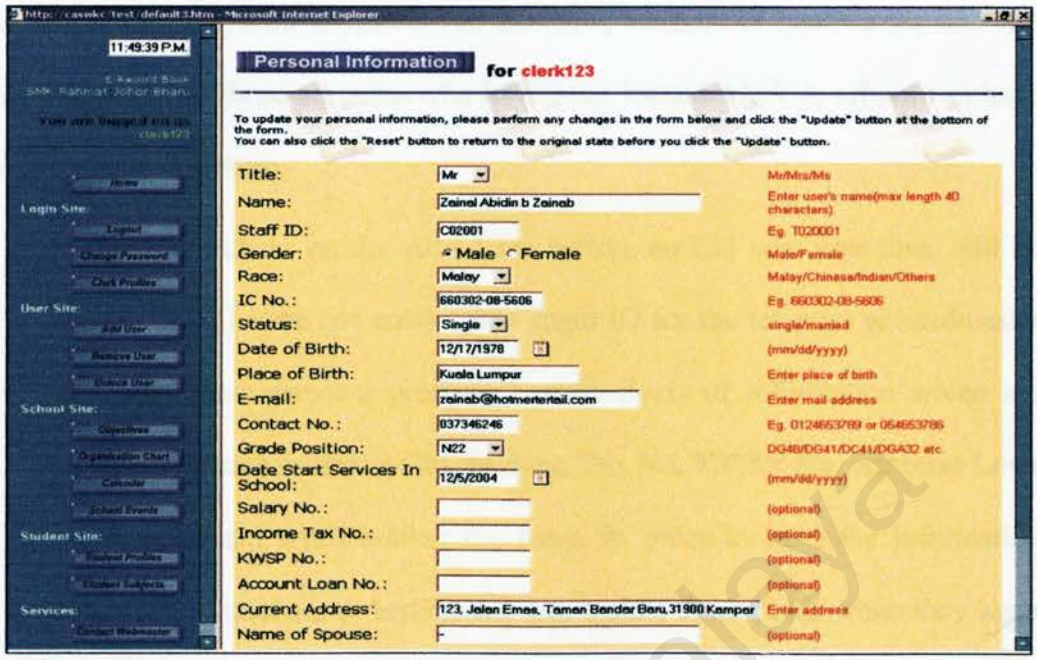
3. If the user has successfully changed the password, a new page will be shown to inform that his/her password has been changed. The page itself will redirect back to the Change Password page if not clicked.



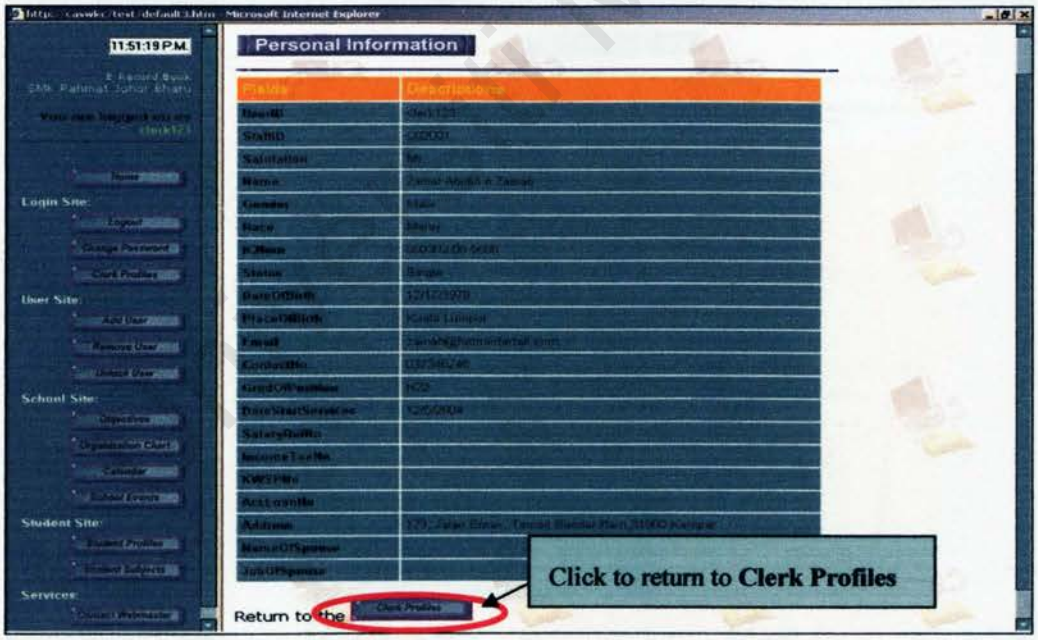
4. For the **Clerk Profiles** button, he/she can view the profiles saved in the E-Record Book system. The clerk can update his/her own profile by click the **Update** button.



5. The clerk can perform changes in the user profiles and update them with the latest information. After that, he/she can click the **Update** button below the form or **Reset** button to undo any changes.



6. Then, he/she can view the changes made to the profiles.

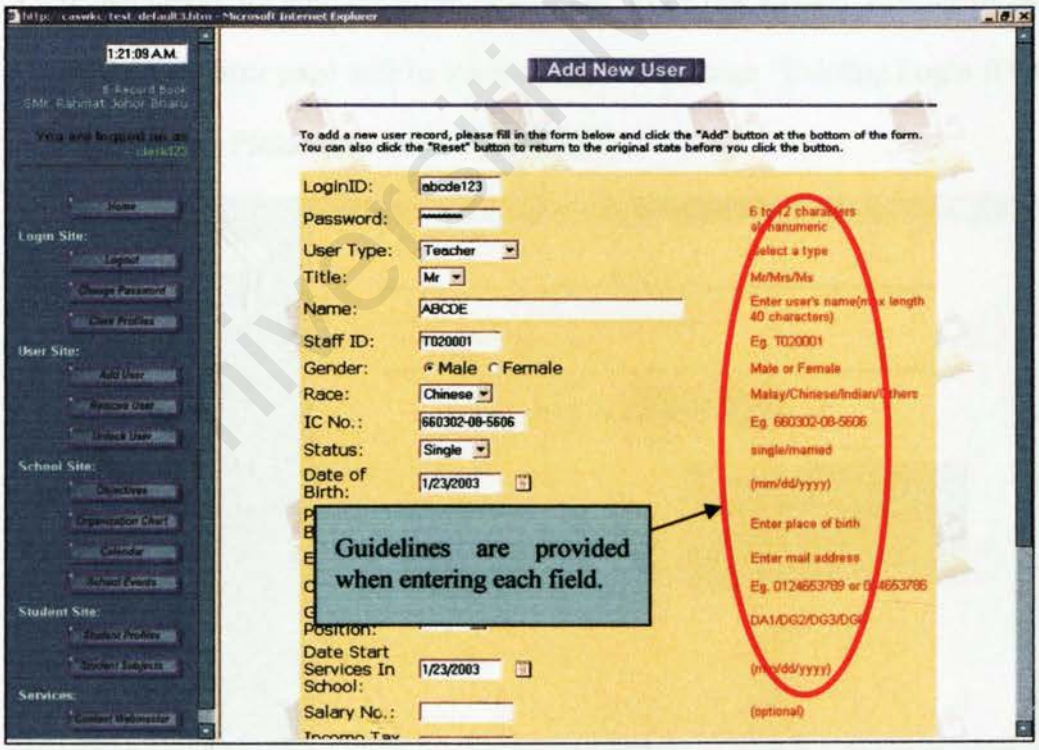


B.1.2.2 User Site

Clerk able to create accounts for teachers and headmaster by adding new users into the system database. He/she can also remove the users who not longer

have the access to the system anymore. For example, teachers or headmaster who has retired or resigned. For those accounts who have been locked, clerk in school can have the capability to unlock them.

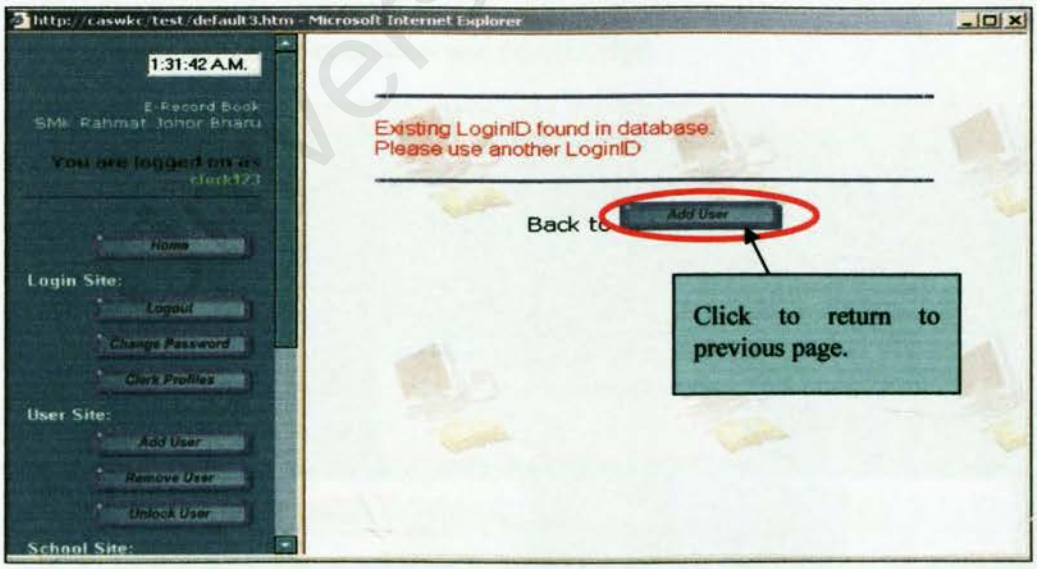
1. If the clerk clicks on the **Add User** button, an add new user form will be shown. Then he/she can enter a new login ID for the teachers or headmaster and enter their personal profiles. Certain fields of information which are confidential such as **Salary No, Income Tax No, KWSP No, Account Loan No** are optional when adding the form. In order to keep the information confidential, teachers or headmaster can update these fields when they login into their own account.



2. When finish filling the form, the clerk has to click the **Add User** button to submit the form. If successful, the page containing all the information about

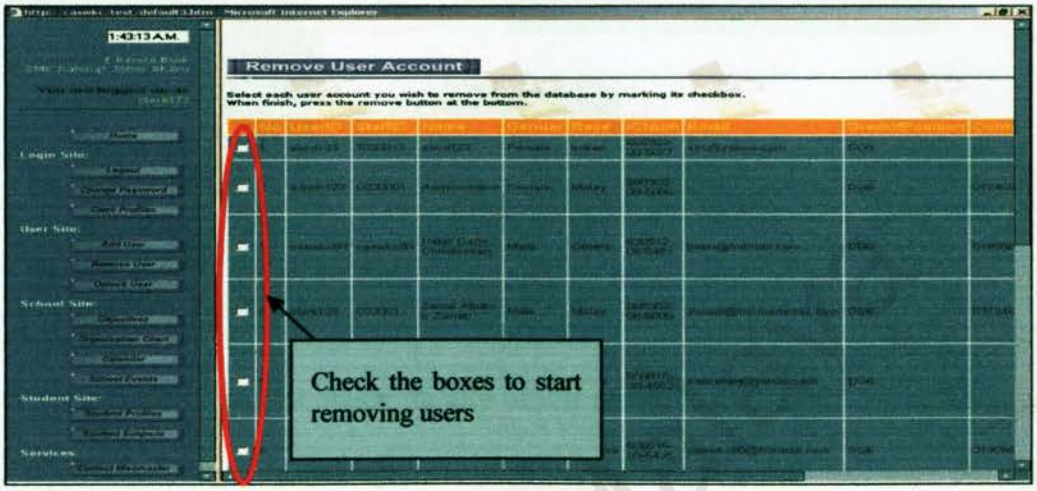
the new user will be shown in order to check if the data entered were correct.

3. If the Login ID entered in the Add User form has already existed in the database, a error page will be shown with the message "Existing Login ID in the database. Please use another Login ID".

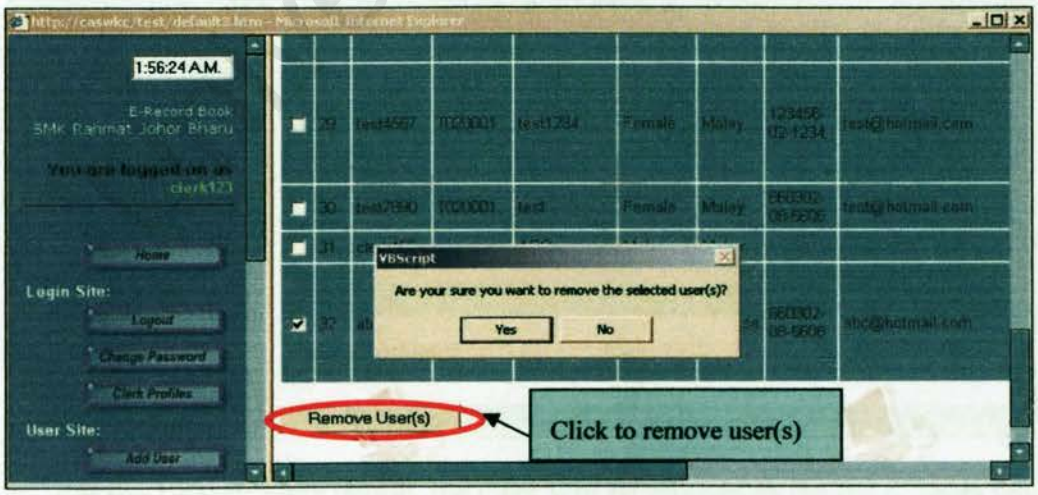


4. Click on the **Add User** button to return to the previous Add User form. Then, change the Login ID and submit the form again.

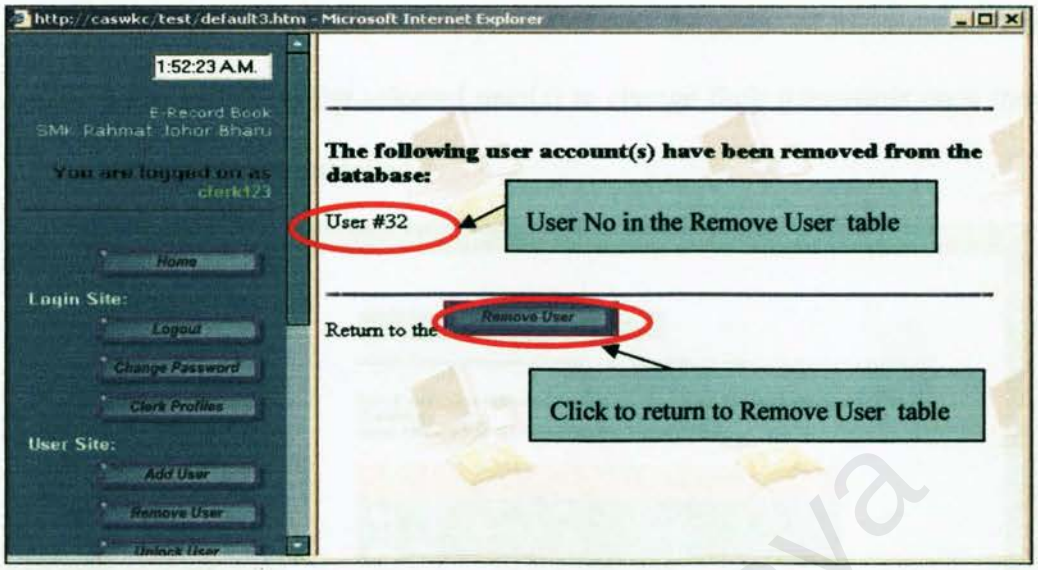
5. If the clerk click on the **Remove User** button, a list of existing user accounts will display. He/she can remove user(s) by checking the box beside the row of the selected user(s).



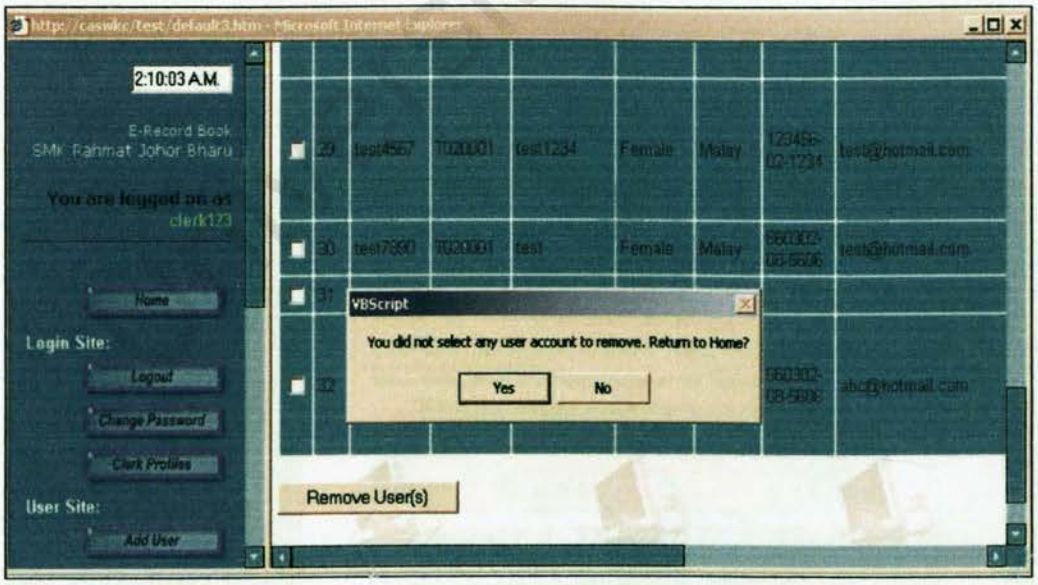
6. Then, the clerk should click on the **Remove User(s)** button below the table. He/she will be asked for confirmation to remove the selected user(s). If the **Yes** button is selected, then the user(s) accounts will be deleted. If **No** button is selected, then the operation will be canceled.



7. The selected user(s) no. who have been removed from the database will be displayed.

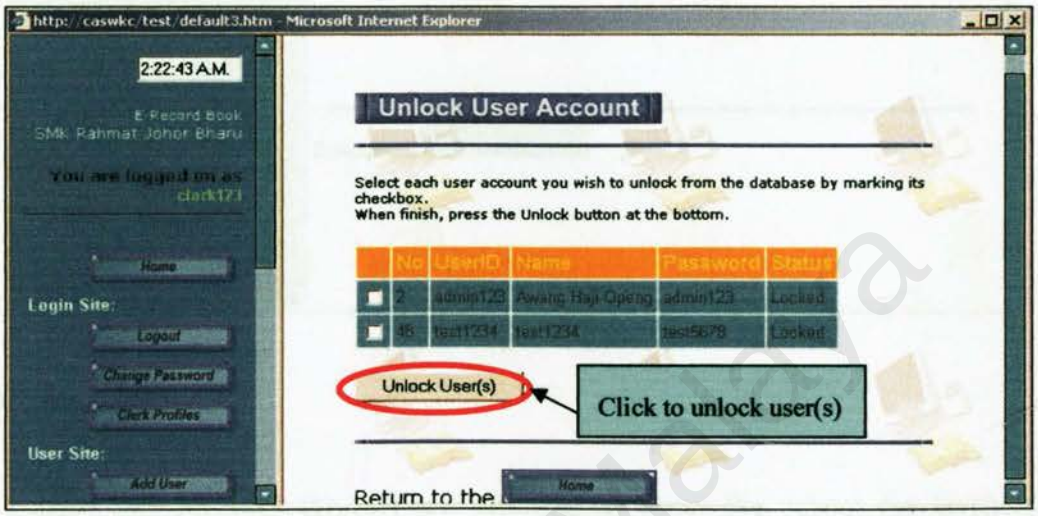


8. If the clerk did not check on any boxes and he/she clicked on the **Remove User(s)** button, a message box will prompt out to notify that user did not select any checkboxes. He/she will asked if the page needs to be redirect back to the home page of E-Record Book.

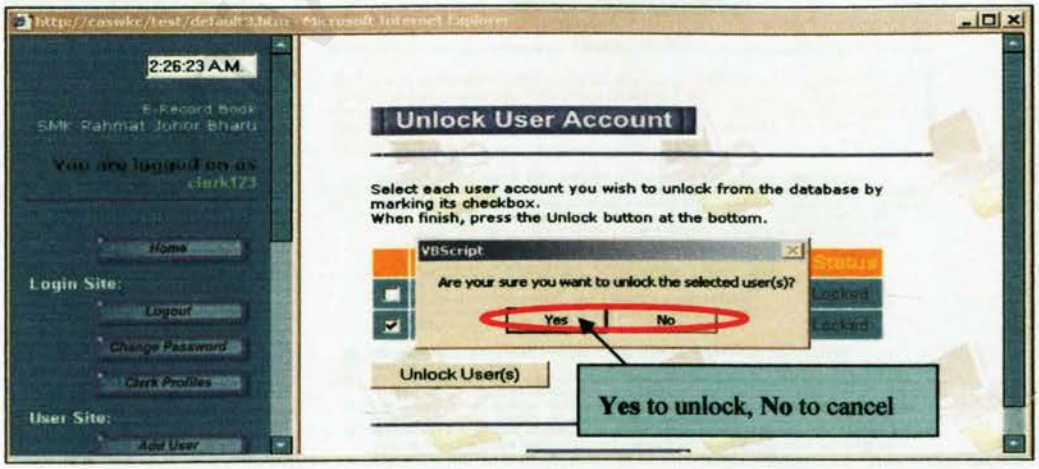


9. For the **Unlock User** button, the clerk will be brought to the list of users who have their account locked. The table contains the information on the Login

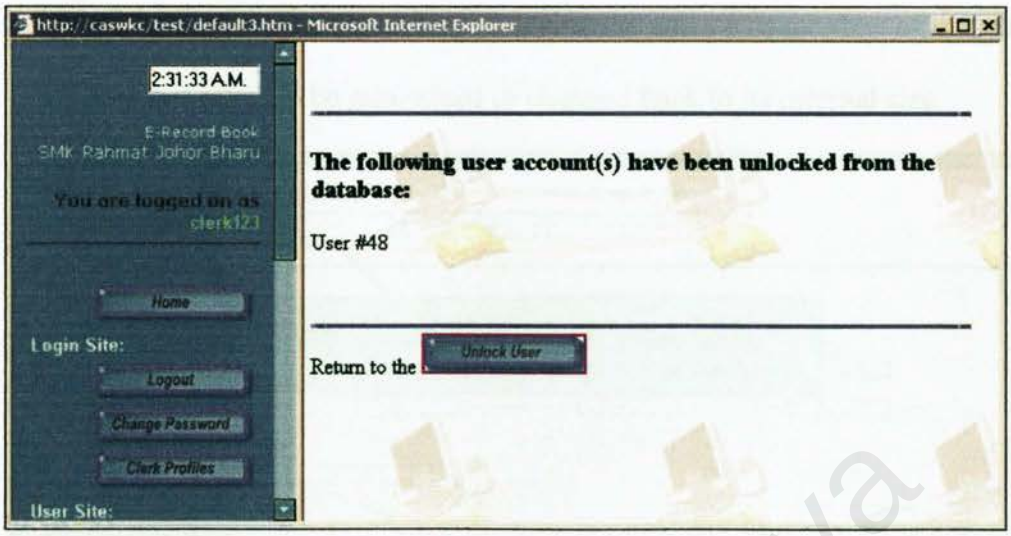
ID (user id), user name, his/her password and the status of his account. The clerk has to inform the selected user(s) to change their passwords once they have their account unlocked.



10. Clerk can select more than one user to unlock their account by having the checkboxes checked. Then, he/she should click the **Unlock User(s)** button to unlock the selected user(s). Again the clerk will be asked for confirmation to unlock the user(s).



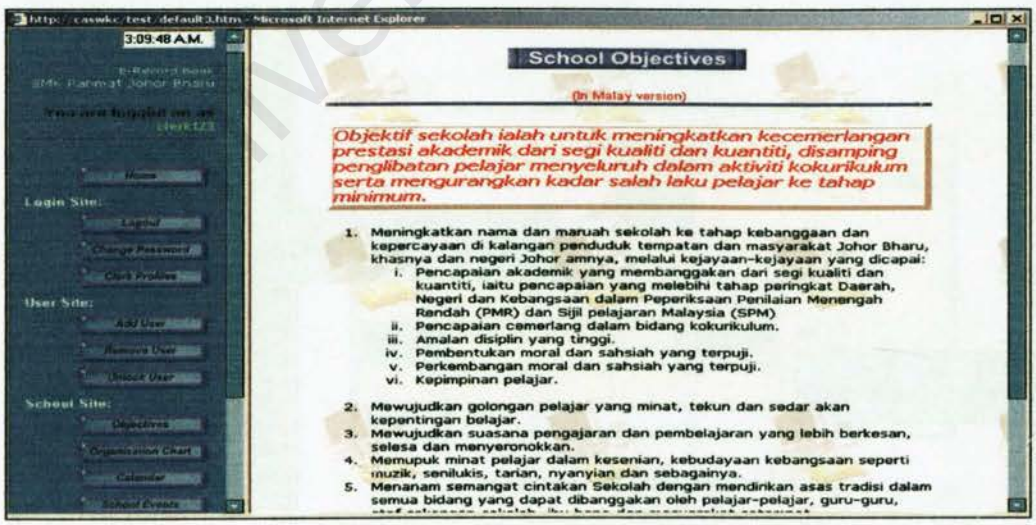
11. After successfully unlock user(s), the clerk will shown which account has been unlocked.



B.1.2.3 School Site

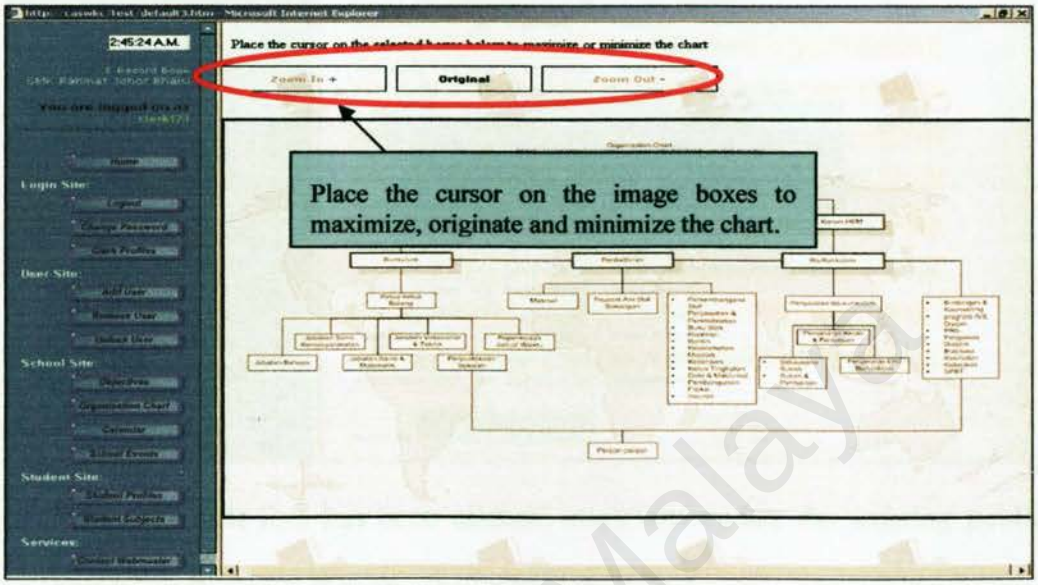
School Site includes the relevant information about the school for teachers, headmaster and clerks references such as school objectives, organization chart, school public holidays and annual events in school.

1. The **Objectives** button will lead the user to the school objectives page.

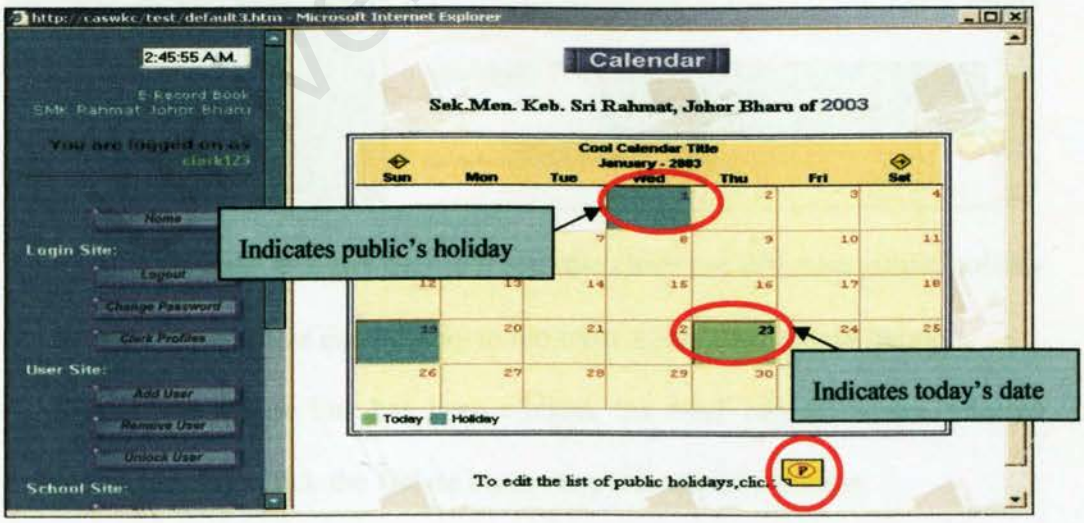


2. While that, the **Organization Chart** button will display the school current organization chart from top to bottom of the school management level. The

organization chart can be enlarged to give the users a better view of the chart. Besides, it can also be minimized or changed back to its original size.

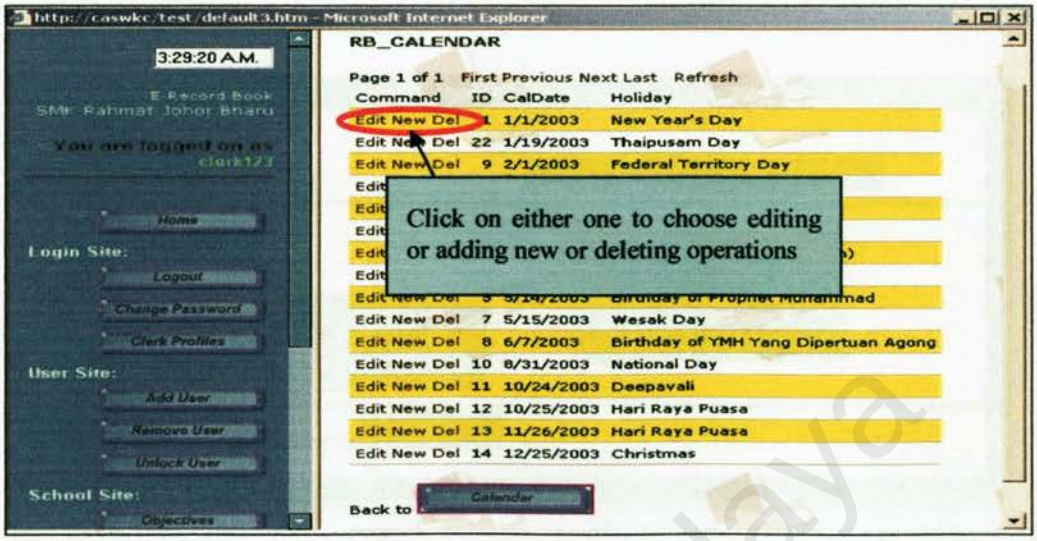


1. When the clerk click on the **Calendar** button, he/she can browse through the current month in the calendar. To create/update/remove any public holidays for the school, he/she has to click on the  icon below the calendar.

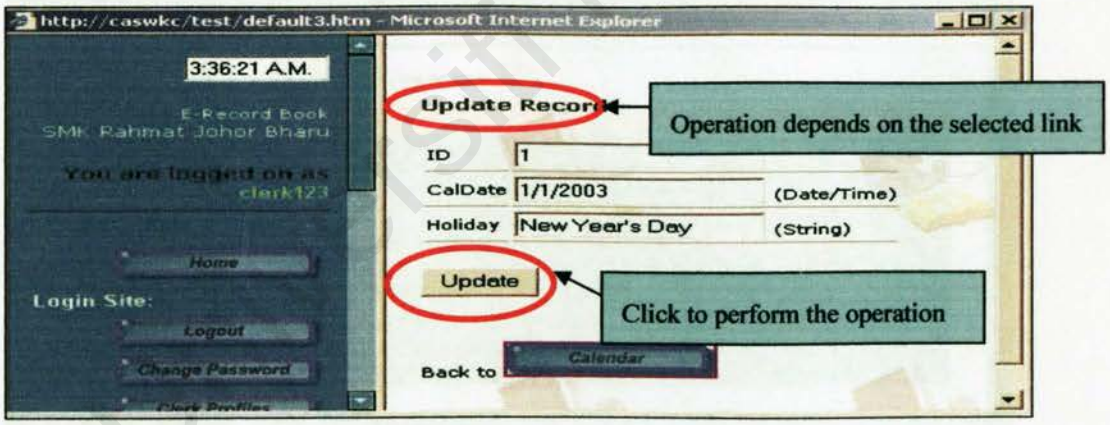


2. Once the icon has been clicked, a list of public holidays which has been created earlier will be displayed in chronological order. Clerk can click the

options on the left of the table to perform editing tasks.

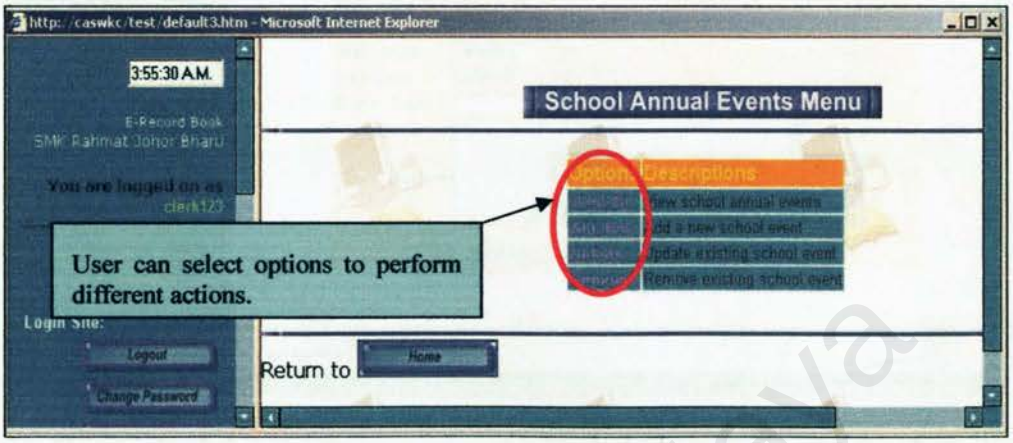


3. When the **edit** link has been clicked, clerk can update the existing public holiday with a new one. Then click **Update** button.

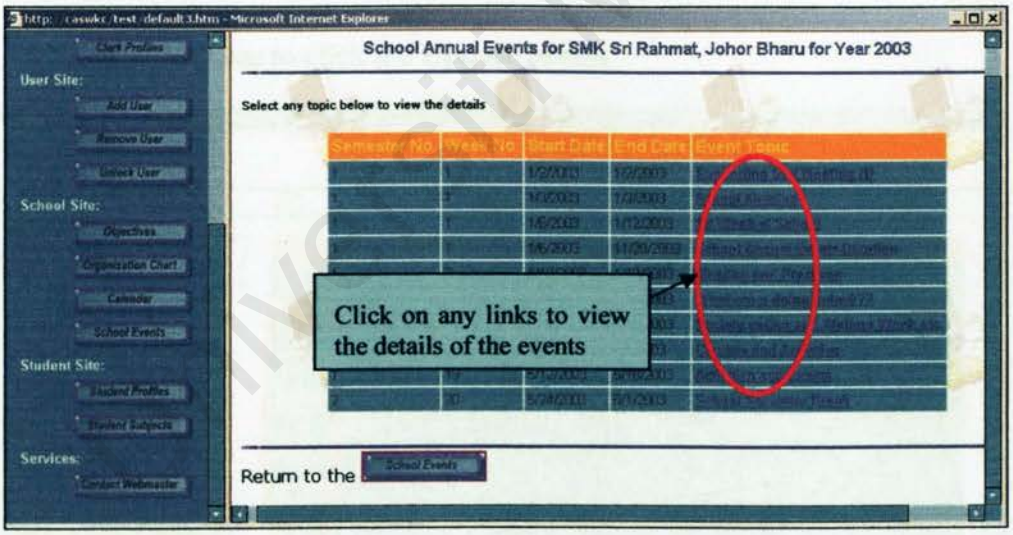


4. Then if the **new** link has been clicked, the clerk can add new public holiday by inserting the date and holiday in the form. Click the **Update** button.
5. Then if the **delete** link has been clicked, the clerk can delete the selected public holiday. Click the **Delete** button to perform this function.
6. For the **School Events** functions, the clerk will enter into the School Annual Events Menu once the button is clicked. The menu provides clerks to be able

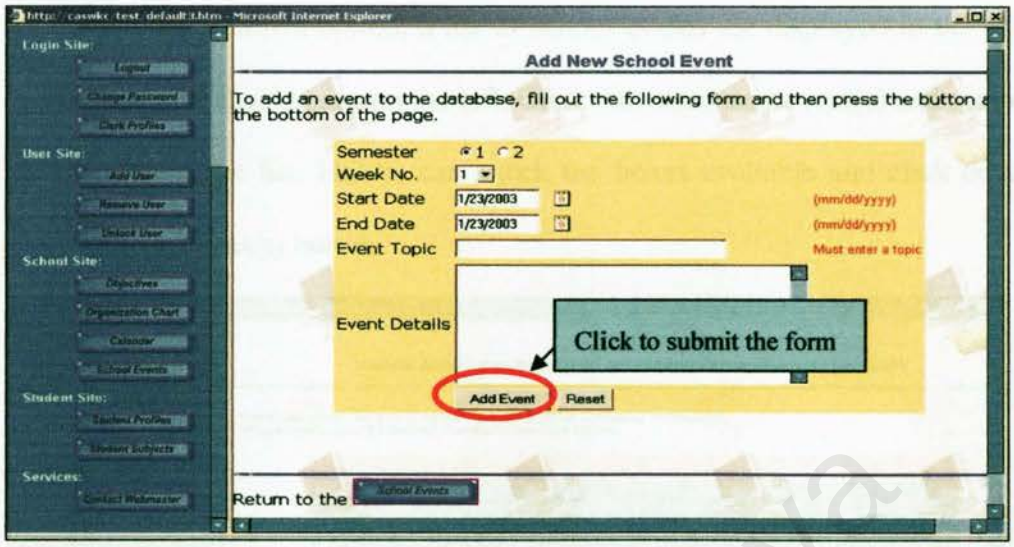
to view all the records of school events, create a new event, update it or may be delete it from the database



7. If **View All** option is clicked, then the clerk able to browse all the events which have been created in the database.

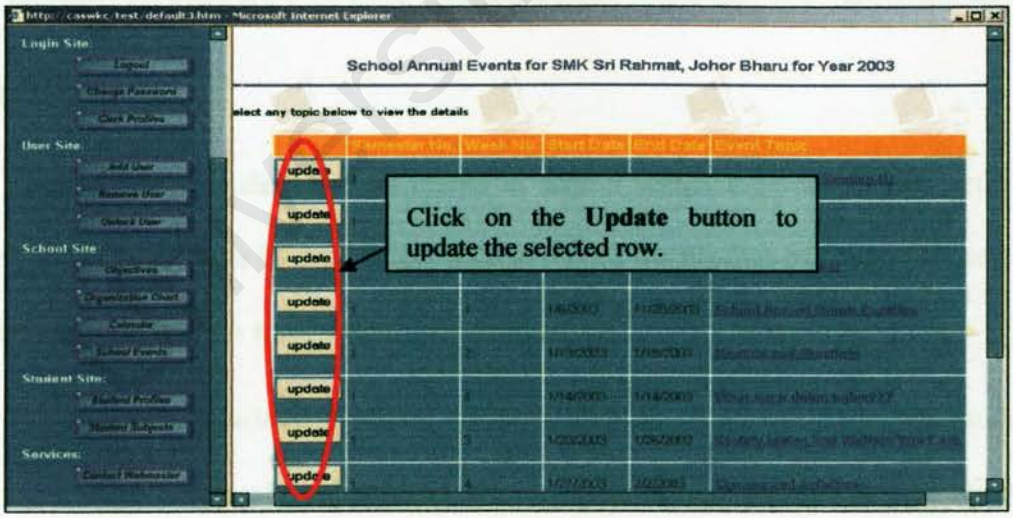


8. Where as if option **Add New** is clicked, **Add School Event** form will display for clerk to enter the relevant information such as semester, week no, start date, end date, event topic and event details. When finish, he/she can click the **Add Event** button to submit or **Reset** button to clear the form.



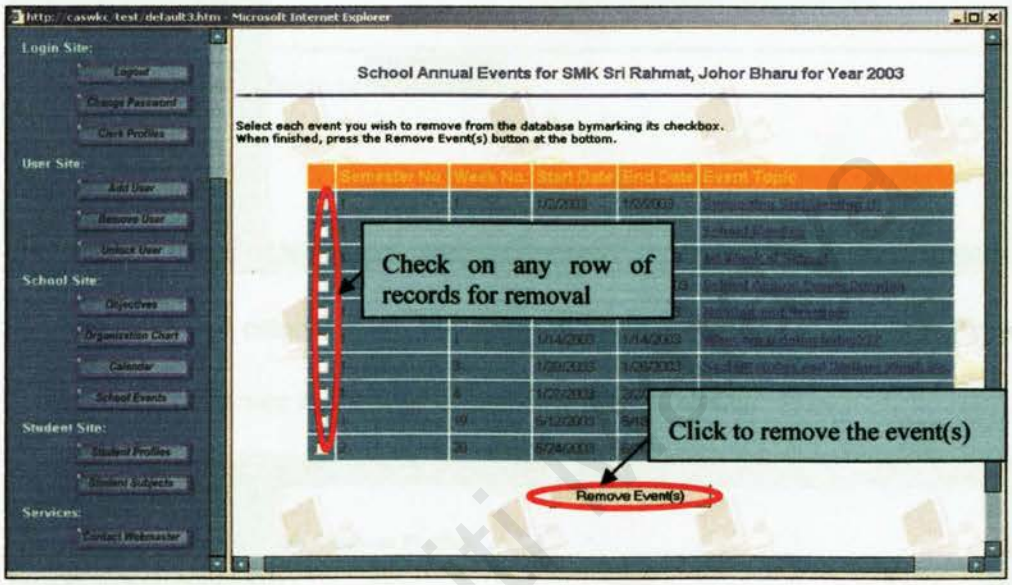
9. Upon submitting the form, a page showing all the information that have been entered for the school event will be displayed.

10. For the **Update** option, clerk can select which event needs to be updated. Then, he/she has to click the **Update** button.



11. When the **Update** button is clicked, another **Update School Event** form which is similar to **Add School Event** form will display. Here clerk is required to make the necessary changes to the selected event and click the **Update Event** button.

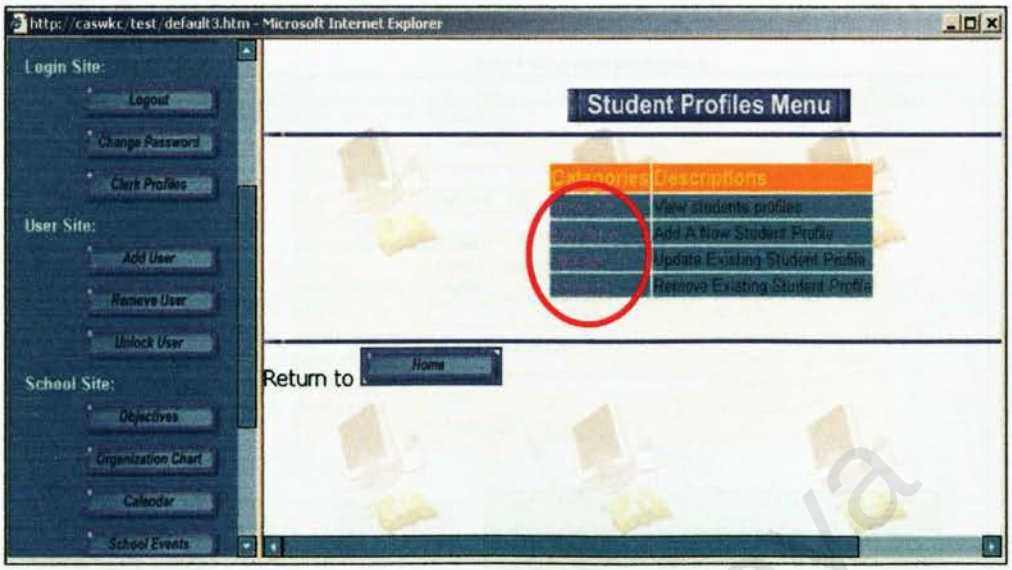
12. As for the **Remove** option, a list of school events are displayed in the table format with checkboxes on the left. Clerk can select any event(s) and remove them from the list. He/she can check the boxes available and click on the **Remove Event(s)** button.



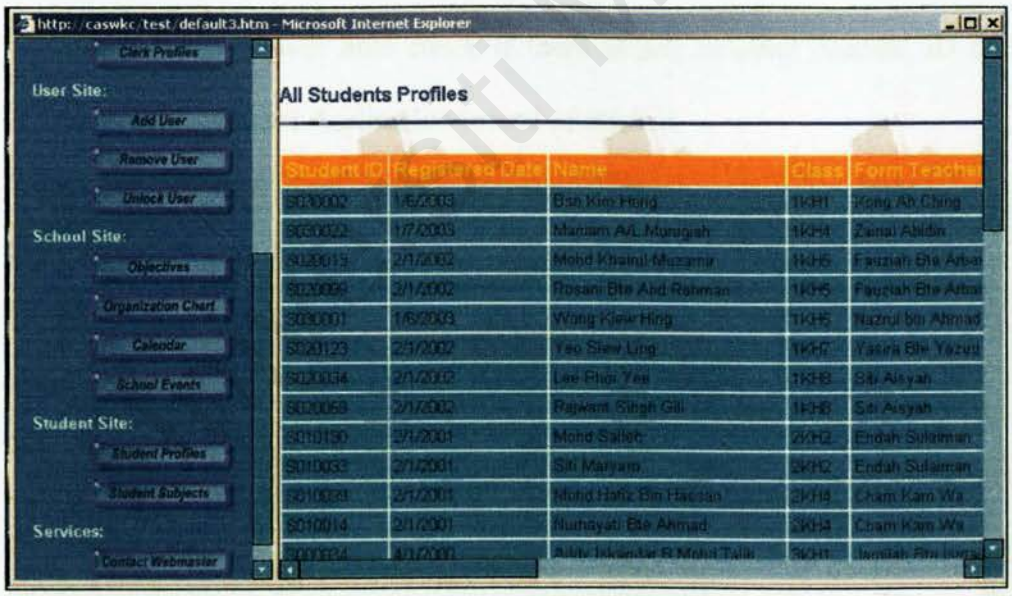
B.1.2.4 Student Site

Student Site is maintained by clerks in school which includes student profiles and their learning subjects in class. Clerk able to record the profiles of students based on their personal information and select the package of subjects for every student.

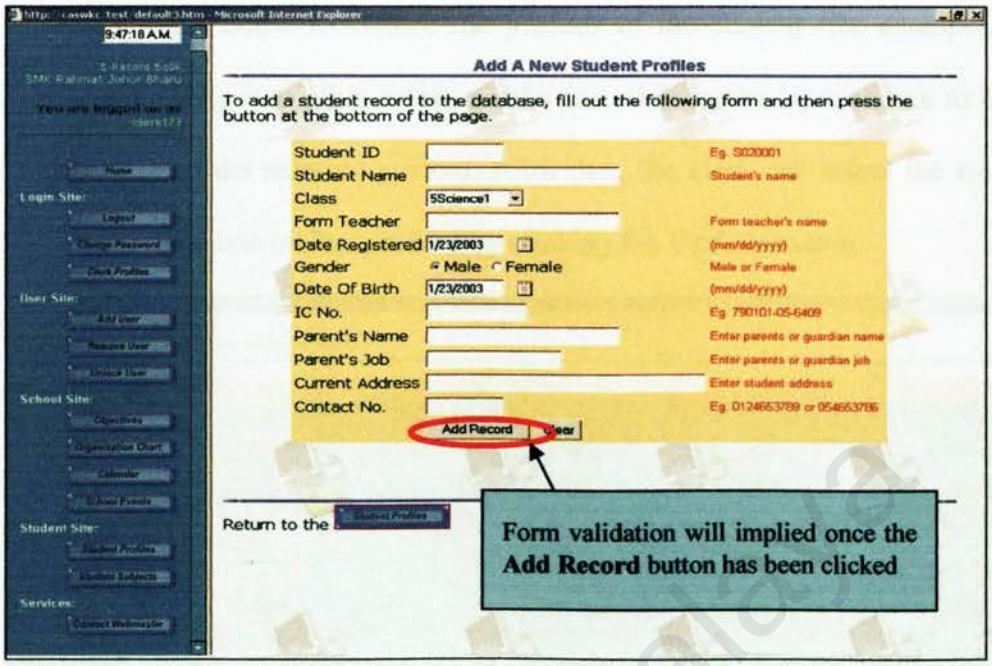
1. When the **Student Profiles** button is clicked, clerk will enter the Student Profile Menu where he/she is able to view all / add / update and remove students from the database.



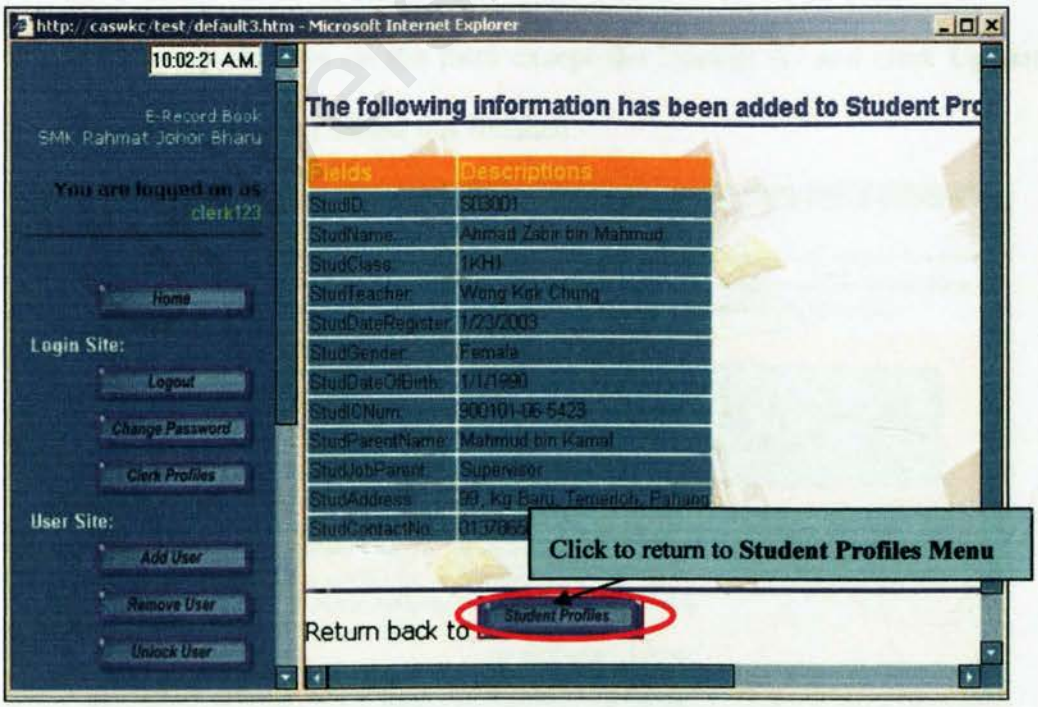
2. The **View All** category will list all the student profiles in school and this will acts as a reference for school to keep track on students.



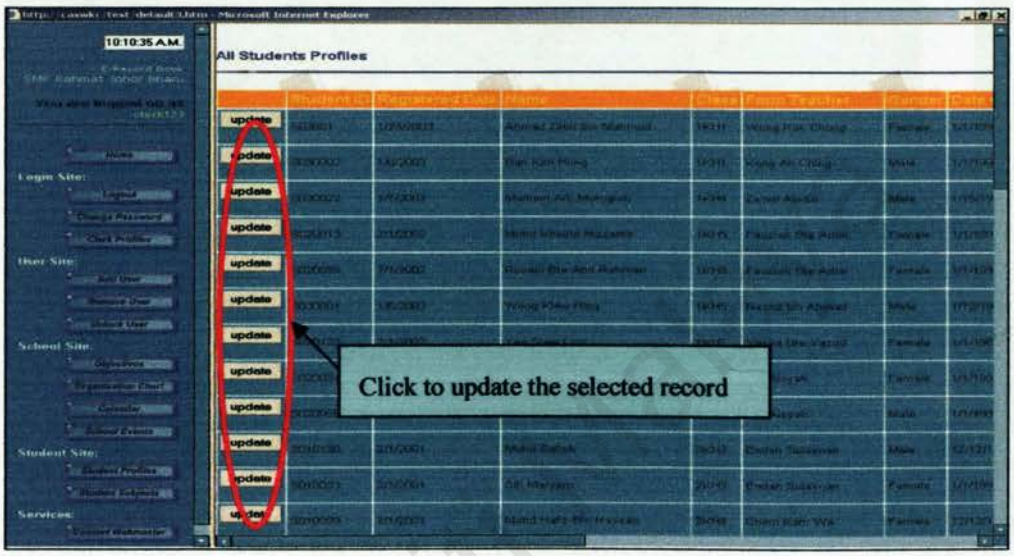
3. Where as, the **Add New** Category provides clerks with the capability to add new student profiles when they are enrolled into the school. An **Add Student Profiles** form will displays once the **Add New** link has been clicked.



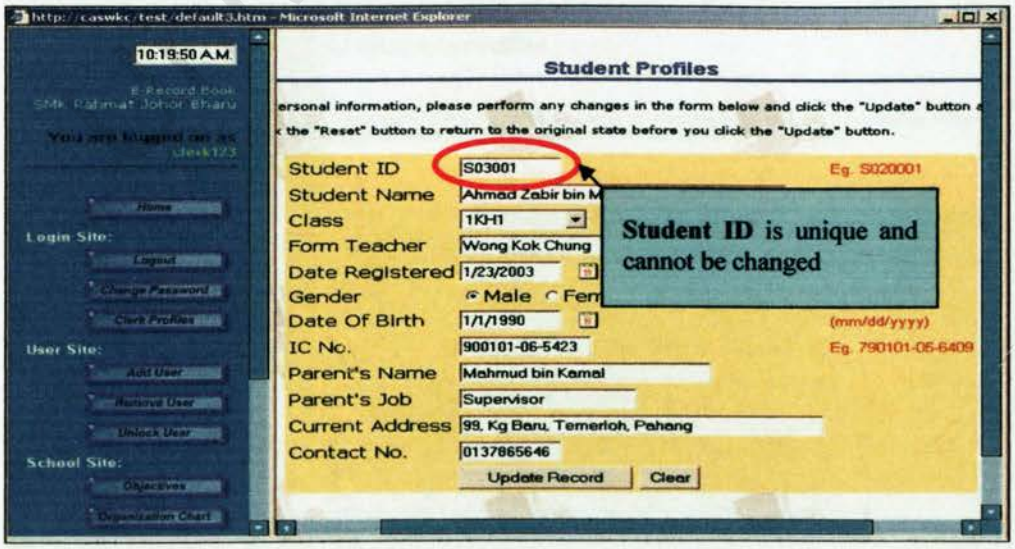
4. Once the form has been submitted, it will display the information that has been entered .It will also check if there is any existing Student ID in the database to prevent duplication record of student.



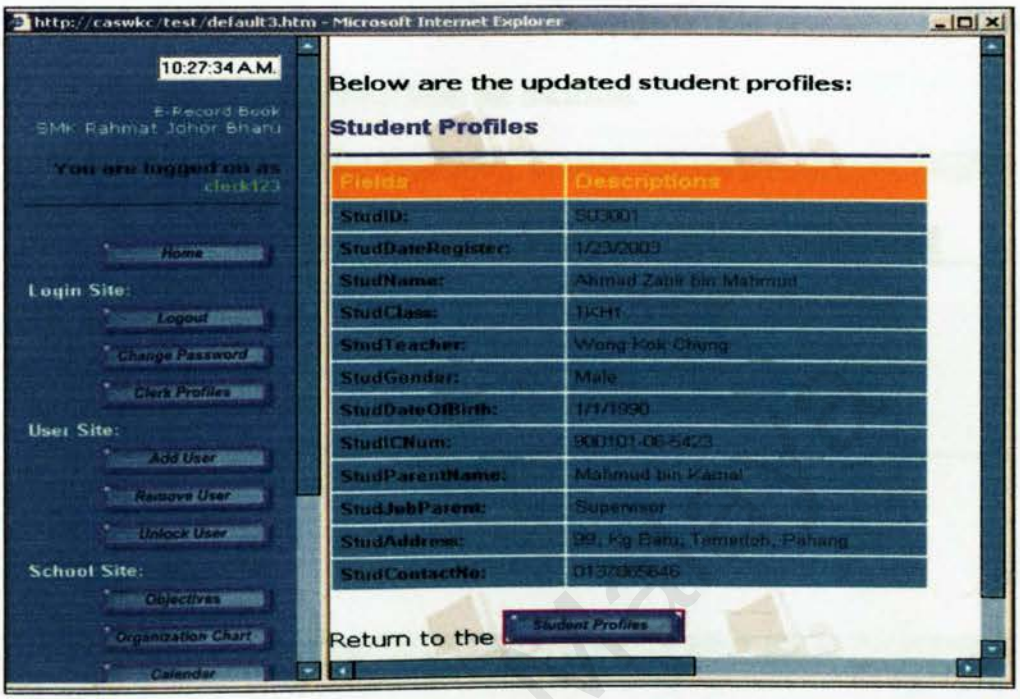
5. If the clerk needs to change the profiles of the student (for example, the student changes to other class), he/she can click on the **Update** link to make changes on the selected student. After that, the clerk can select the student record to update his/her profiles by clicking the **Update** button.



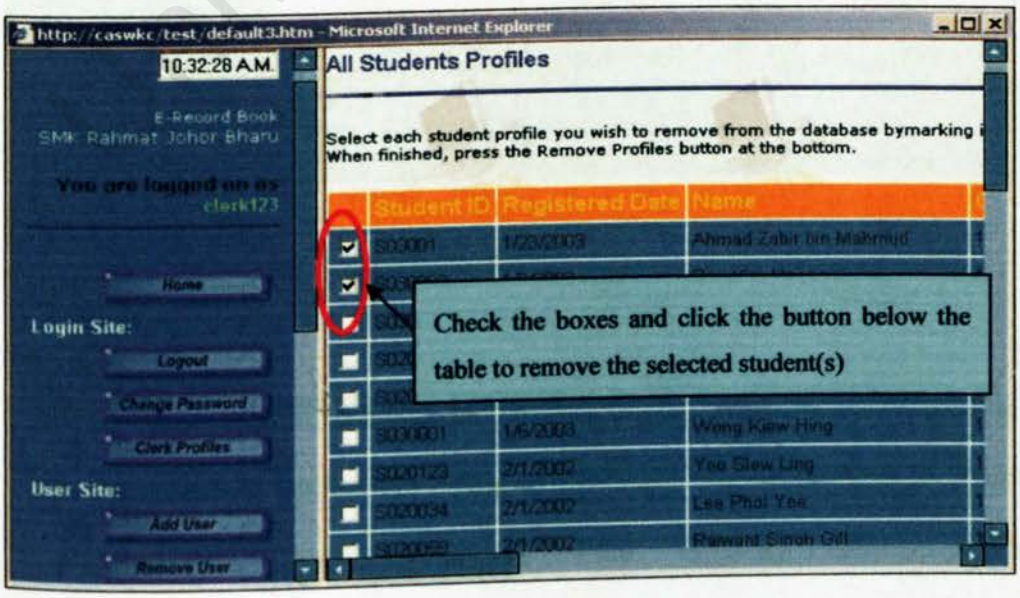
6. Then,. a form with the selected student profiles is displayed. Clerk can perform any changes on the form except the Student ID and click **Update Record** button when he/she has finished.



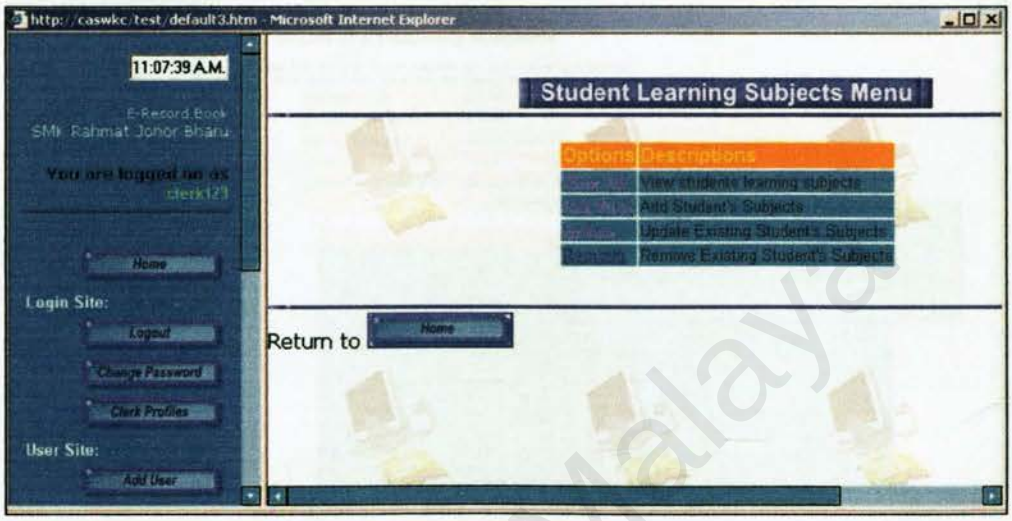
7. The updated profiles on the selected student then will be displayed.



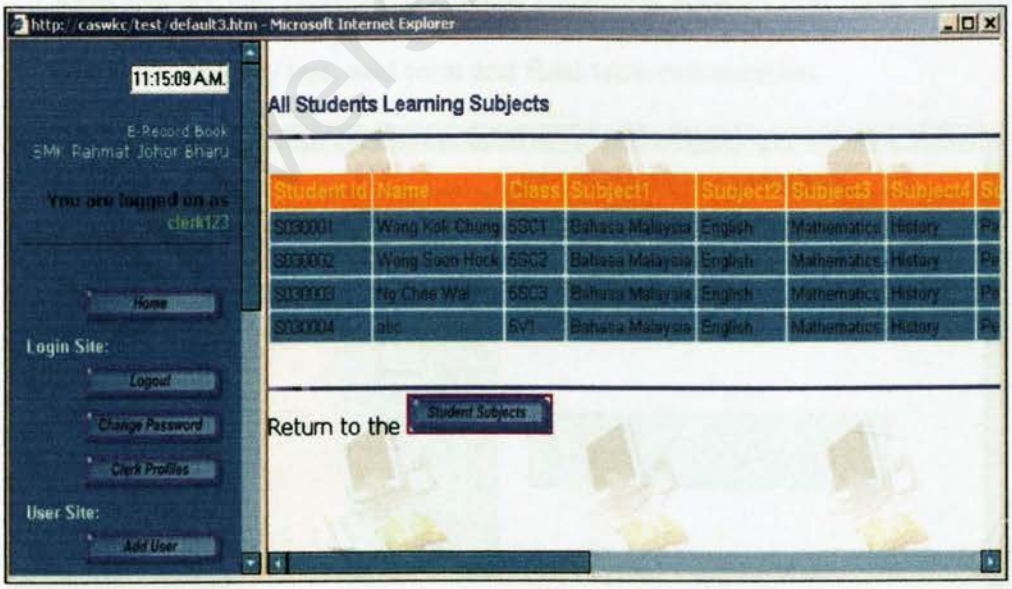
8. As for the **Remove** Category, clerk able to select the student(s) to be remove from the school database. He/she can do this by checking the boxes beside the row of student records and then click the **Remove Profile(s)** button at the bottom of the table.



9. As for the **Student Subjects** module, the same functionality apply to the clerk where he/she able to view all the subjects, add new subjects, update them or even remove them from the database.

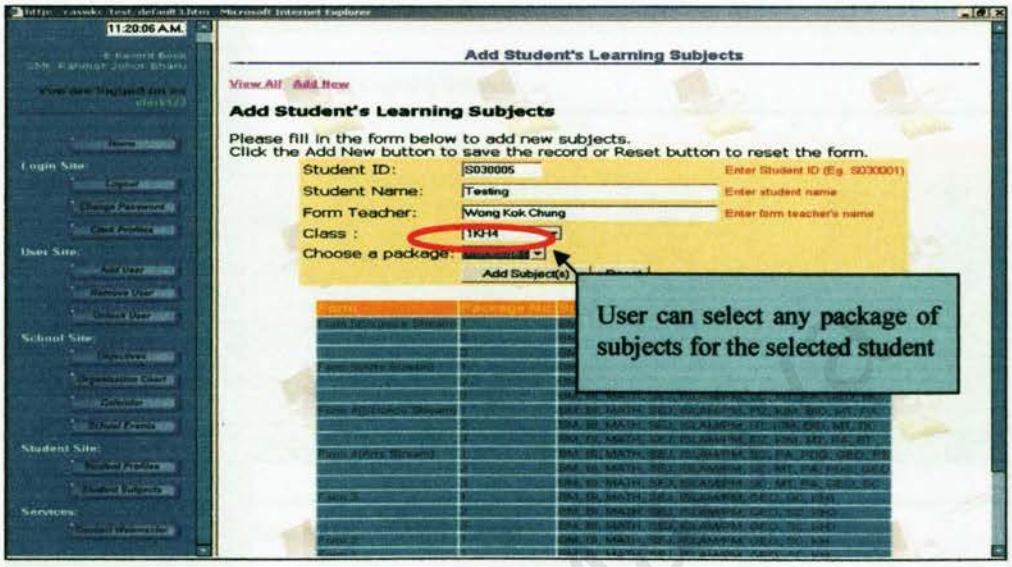


10. For the **View All** option, clerk able to view the subjects taken by every students in school.

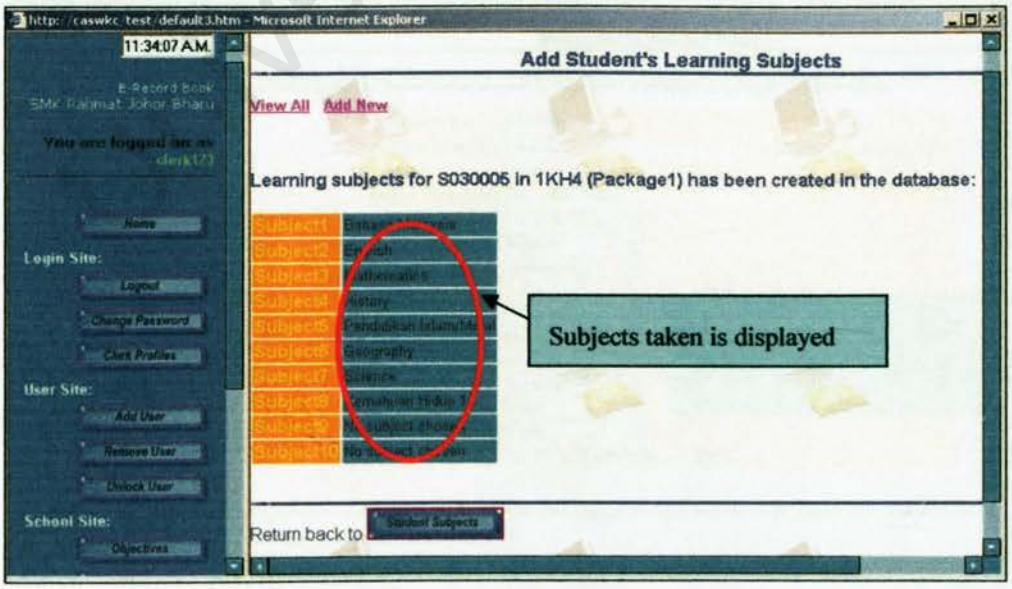


11. In the **Add New** option, clerk able to select the relevant package of subjects for student (for example, the student in Form 5 is taking a different package

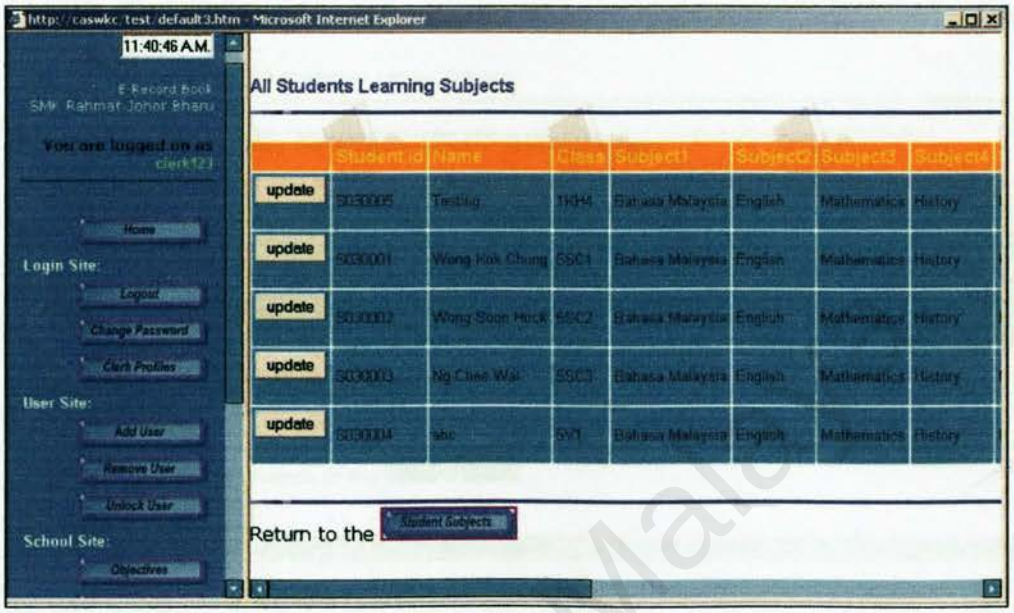
from the student in Form 1). Clerk has the ability to choose any package based on the list of packages provided below the form.



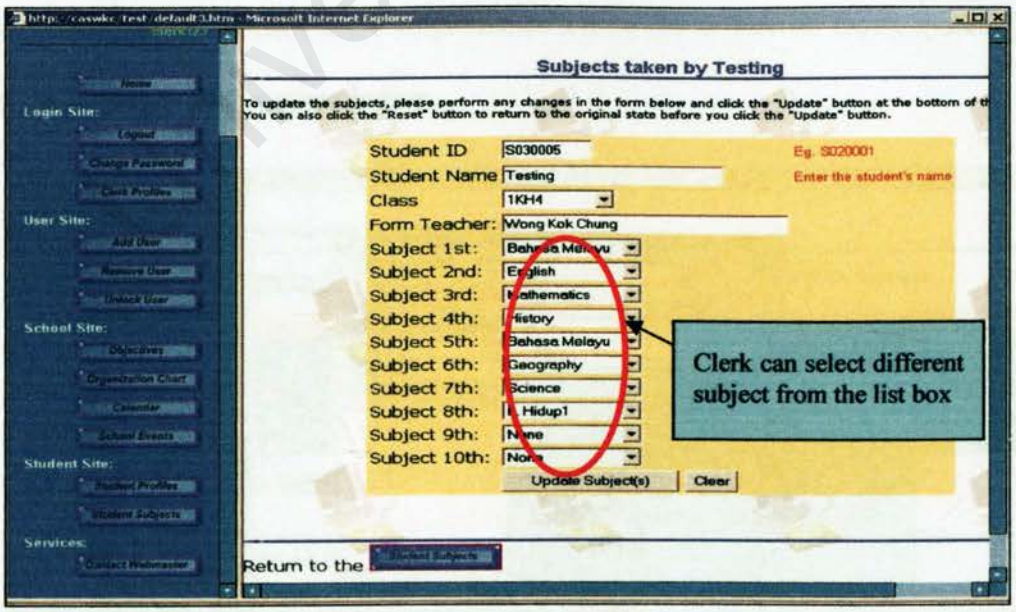
12. The subjects in the package which has been selected by clerk will be displayed when he/she submit the **Add Subject(s)** button. Then the listed form teacher can referred to student's learning subjects and input the marks based on monthly test, mid term and final term examination.



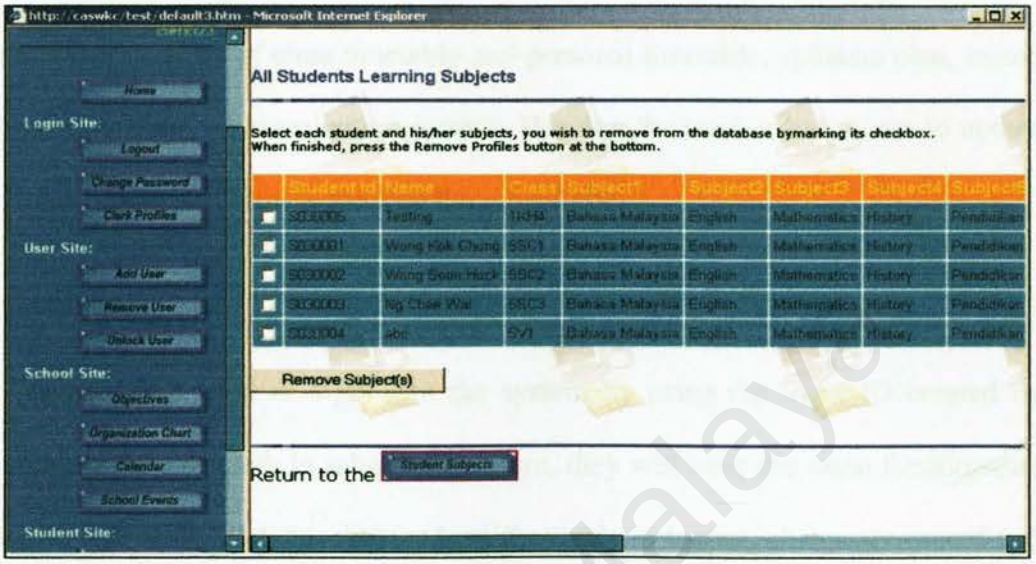
13. Clerk also able to update the subjects taken by students in the **Update** option by clicking the **Update** button.



14. Then a form containing all the subjects which have been selected is displayed. Clerk can select other subjects for the selected student if got any changes.

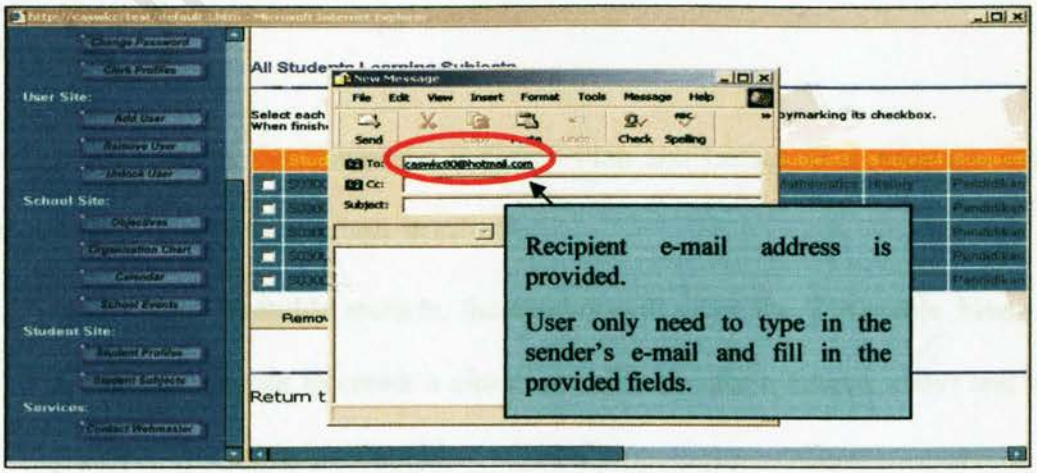


15. As for the Remove option, it is similar to the remove function in other modules such as Student Profiles Module and School Events Module.



B.1.2.5 Services

In the services, the users (clerks, teachers, headmaster) can contact the system administrator if any errors occurred during school hours by sending through e-mail. They can also give feedback on the implementation of the E-Record Book system by clicking the **Contact Webmaster** button.



B.1.3 Teacher Section

The Teacher Section provides teachers with functionality to manage their own timetable which consists of class timetable and personal timetable, syllabus plan, lesson plan, student profiles and examination records. Besides, the teacher has access to update his/her own account and browse through the school site.

B.1.3.1 Login Site

Teachers will need to login into the system by using the login ID created by school administrator or clerk in school. Therefore, they will have the same functionality as the Login Site in **B.1.2.1** for clerks where they able to logout, change password and update their profiles.

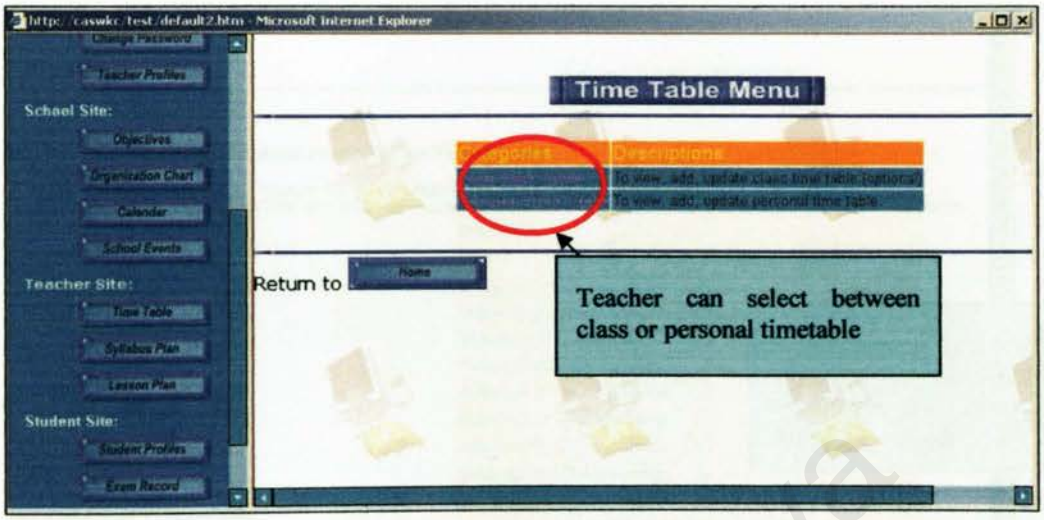
B.1.3.2 School Site

Teachers able to gain access to School Site but they can only browse through the relevant information about the school such as school objectives, organization chart, school public holidays and annual events in school.

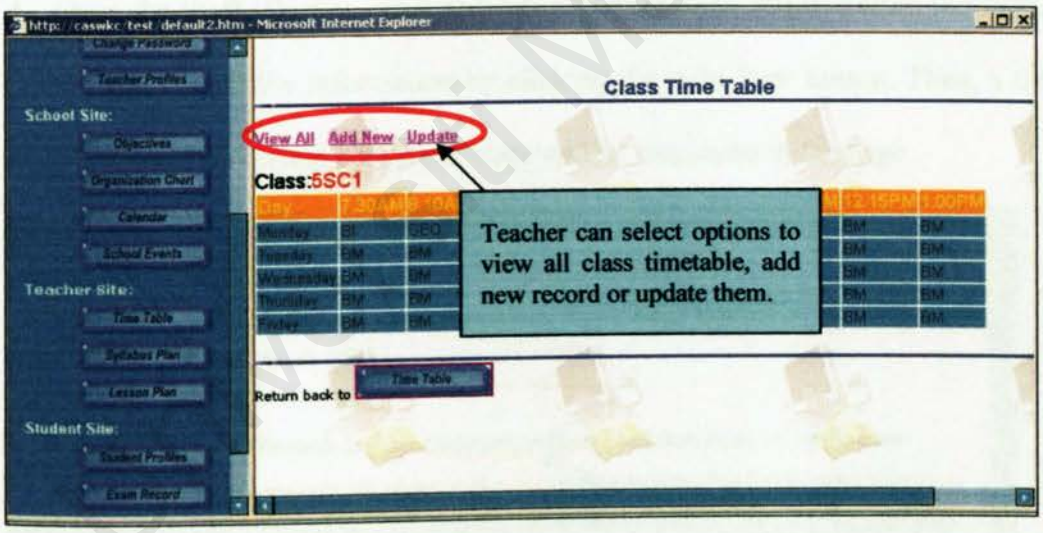
B.1.3.3 Teacher Site

This site contains the Timetable, Syllabus Plan and Lesson Plan modules and provides the functionality for teachers in school.

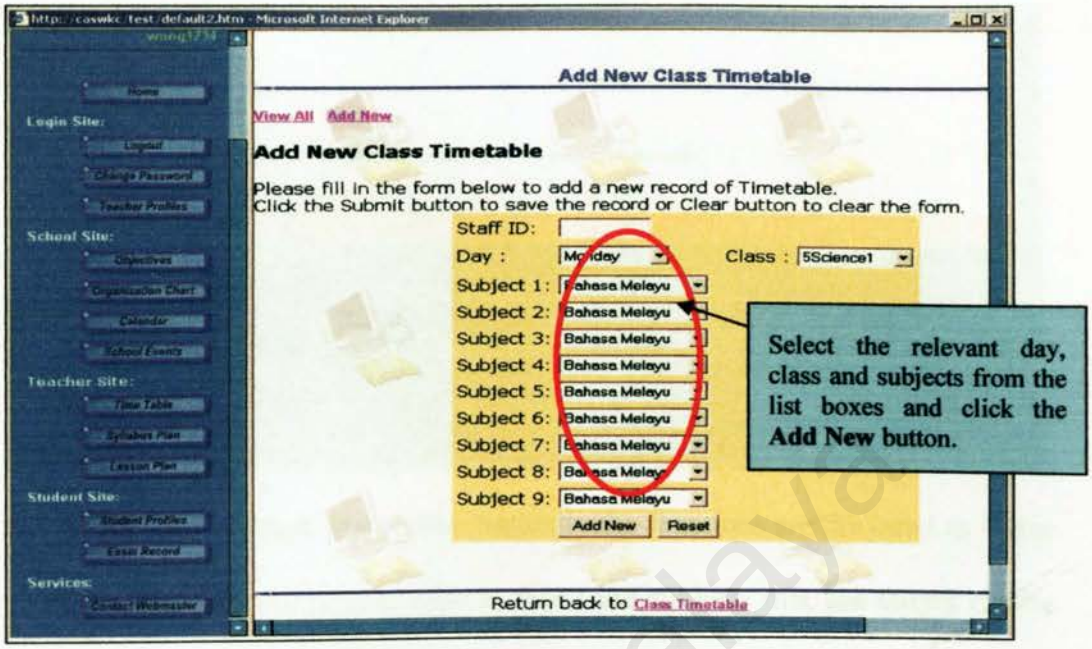
1. Under the **Timetable** module, the teacher will enter the **Timetable Menu**. He able to choose to create a class timetable (for form teacher only) and a personal time table for himself.



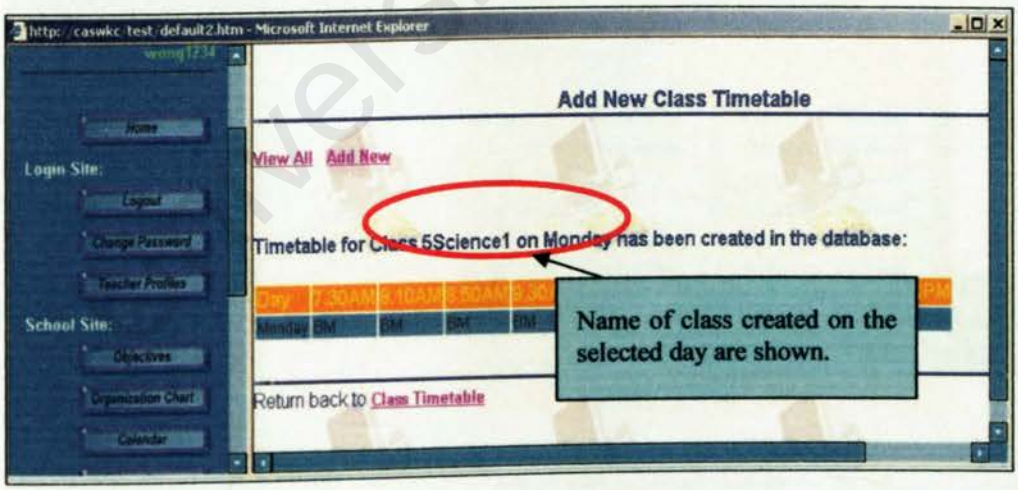
2. If the teacher selects the **Class Time Table** category, she can view, add new or even update the existing time table for the certain class.



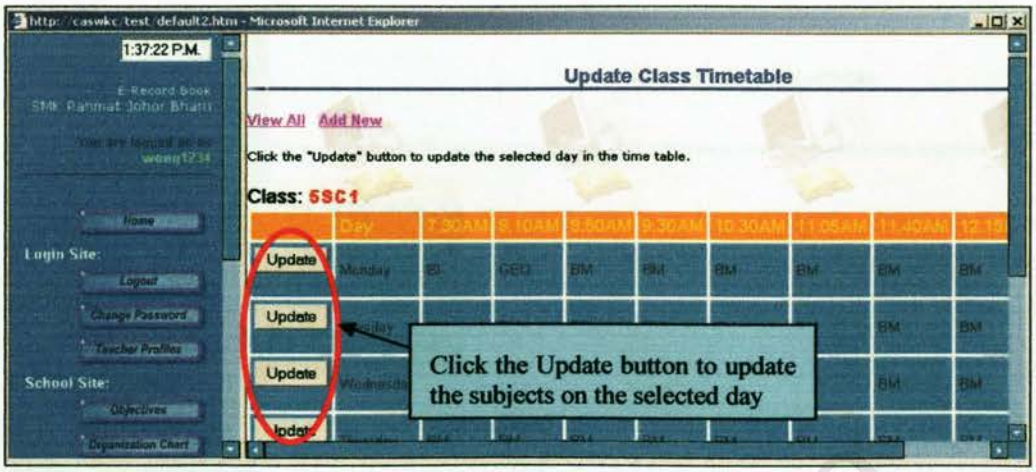
3. If no record is found in the class timetable, the teacher can add new record for certain day which begins from Monday to Friday. She can do this by clicking the **Add New** link.



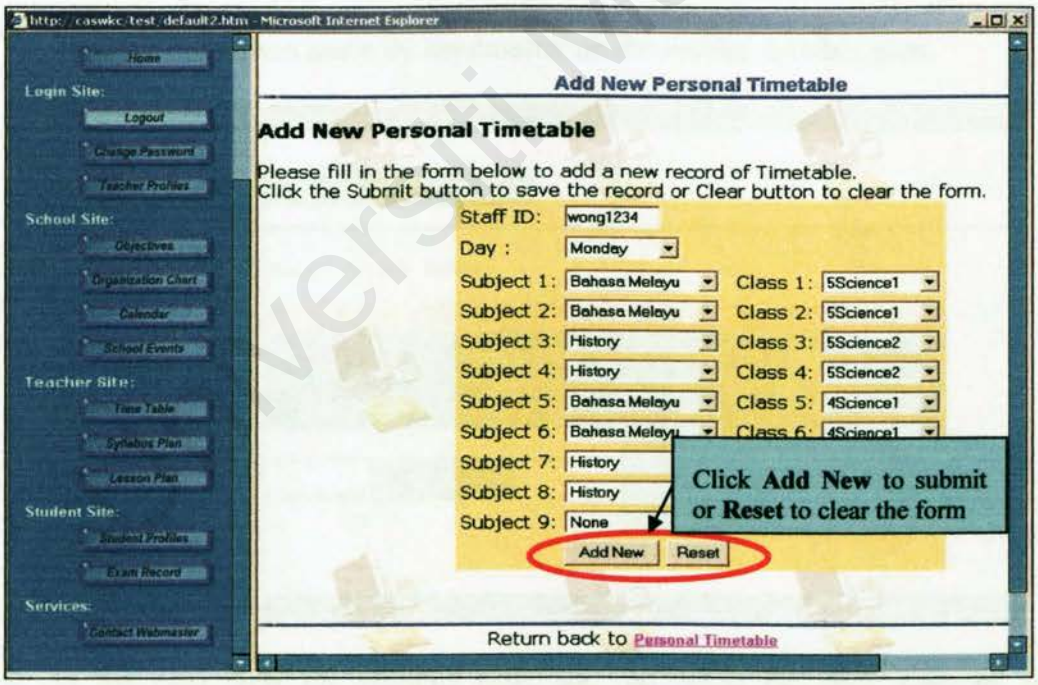
4. Once the Staff ID, day ,class and subjects have been selected from the form, she can submit the information by clicking the **Add New** button. Then, a list of subjects on certain periods in class will be displayed in the page.



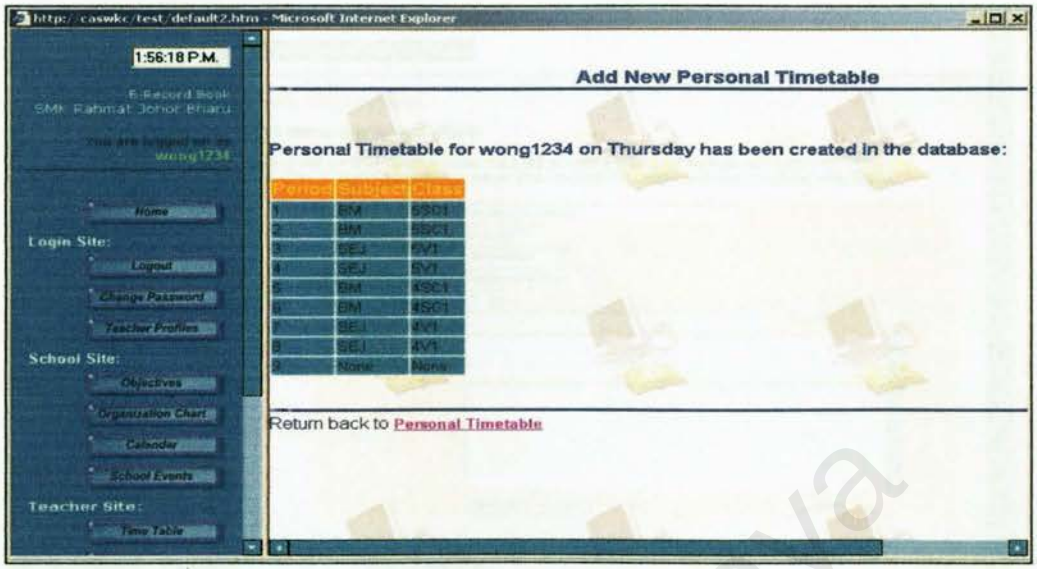
5. To update the existing class timetable, teacher has to click on the **Update** link and select which day that needs to be updated.



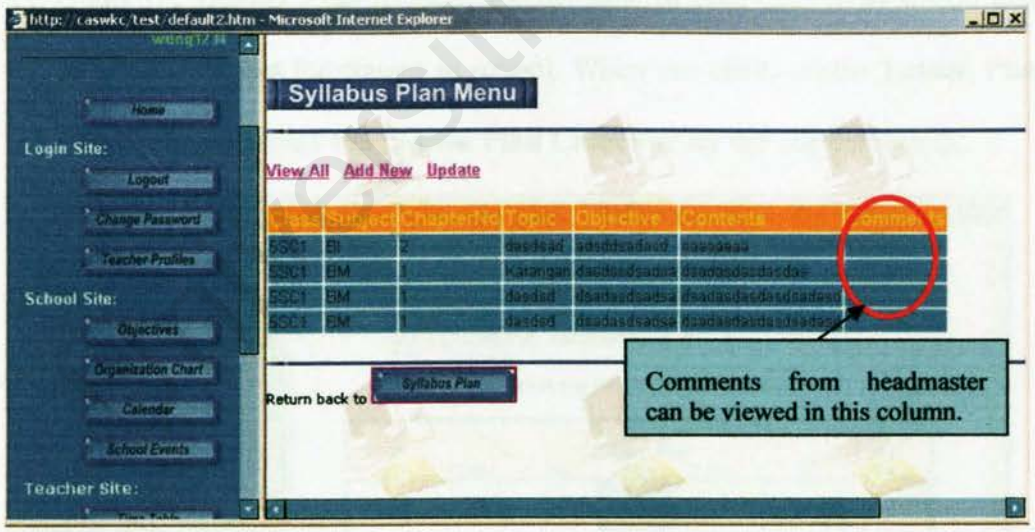
6. As for the **Personal Timetable**, basically the functions are the same as **Class Timetable** but the teacher able to select the subjects that she taught on the selected class and day when she clicks on the **Add New** link



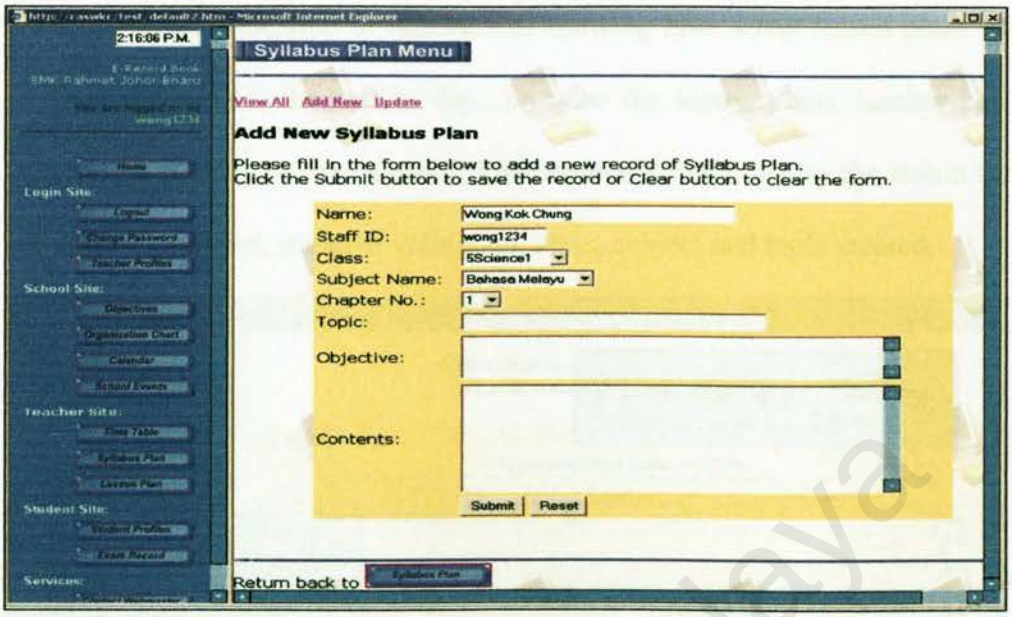
7. When the teacher has selected the relevant subjects for certain classes on that day, she can click the Add New button to submit the form. A new page will display the added information on the personal timetable .



8. Under the **Syllabus Plan** module, teacher able to view her own syllabus plan, add a new syllabus plan or update the existing syllabus plan. She can also view the comments made by headmaster on the written syllabus plan.

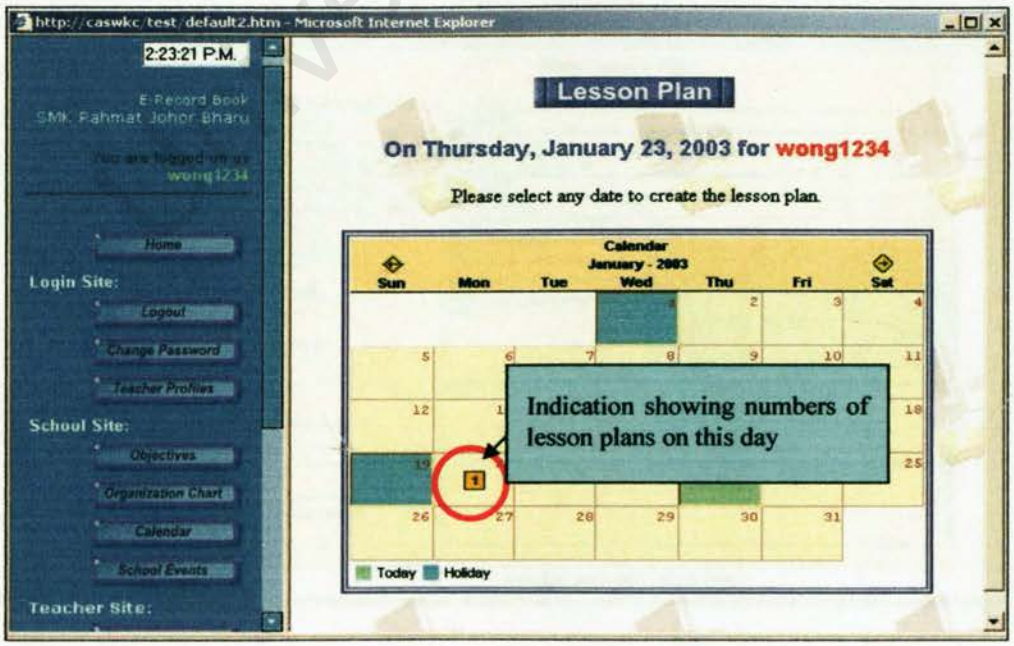


9. In the **Add New** for syllabus plan, she can select the class that she is teaching, the subject, chapter no and type in the topic, objective and contents. Then she can click the **Submit** button to submit the form.

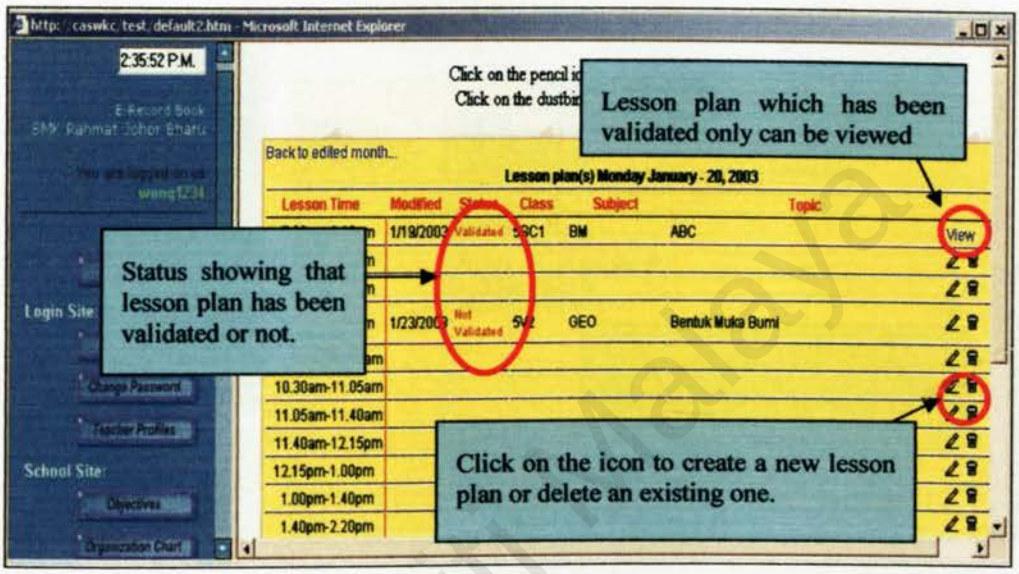


10. As for the **Update Syllabus Plan** module, it is similar with the update timetable function.

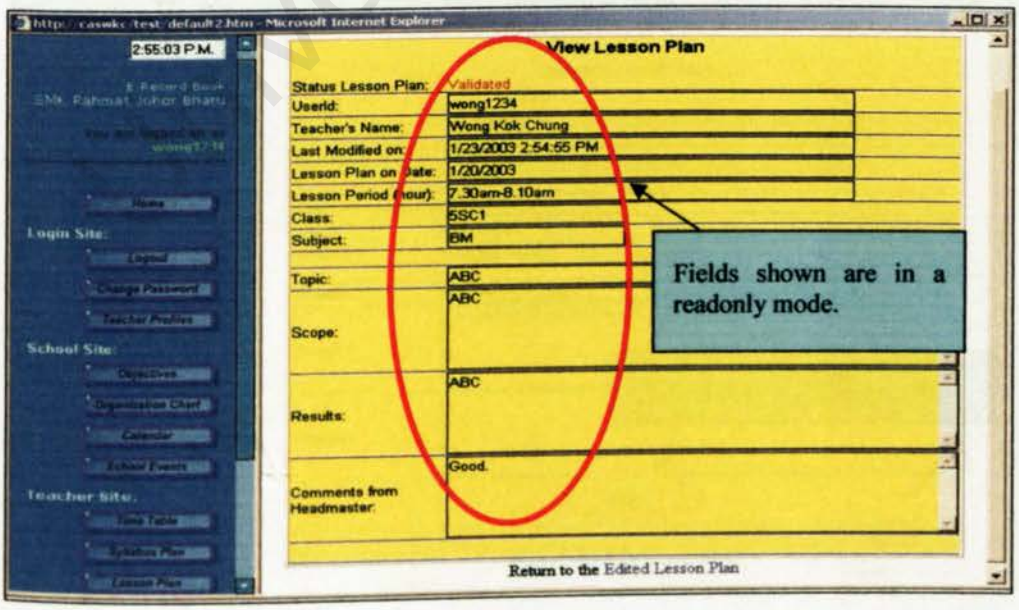
11. Under the **Lesson Plan** module, teacher able to plan their daily lesson plan on certain subject for classes in school. When she clicks on the **Lesson Plan** button, she will enter the **Lesson Plan Calendar** for the current month.



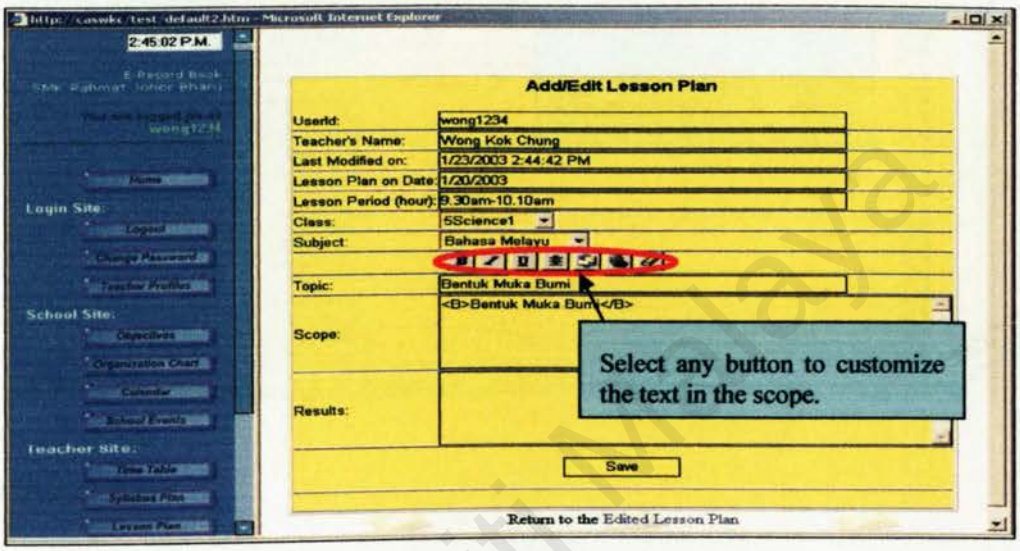
12. There is an indication of numbers in showing how many lesson plans have been created for the certain day. To view the lesson plans, teacher has to click on the day on the calendar. The lesson plan will display the lesson time, date of modified, status of validation, class, subject and topic created.



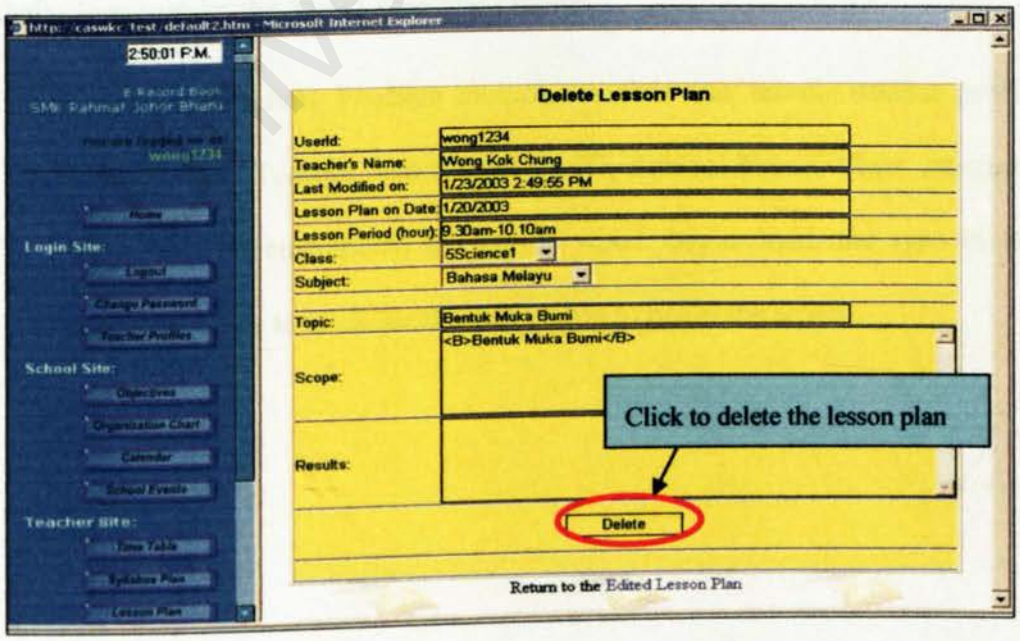
13. To view the validated lesson plan, click on the **View** link. Teacher can only view back the written lesson plan and the comments given by headmaster.



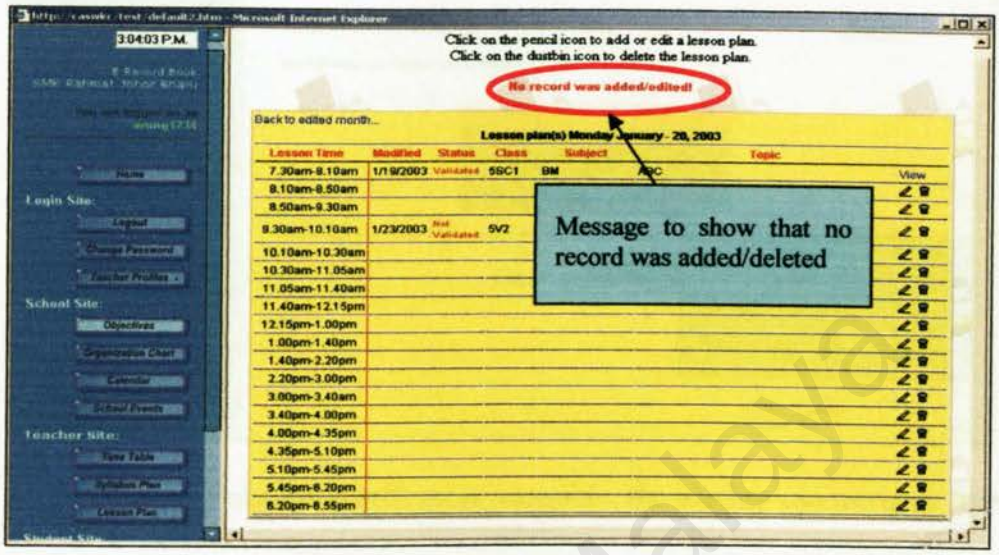
14. To add/edit a lesson plan on the certain lesson time, click on the pencil icon on the right of the table. Then teacher can input the select the subject, class, topic, scope and results from teaching. To submit the form, she can click the **Save** button below the form.



15. To delete the existing lesson plan before it is validated by headmaster, click on the dustbin icon. Then click the Delete button to delete the lesson plan.



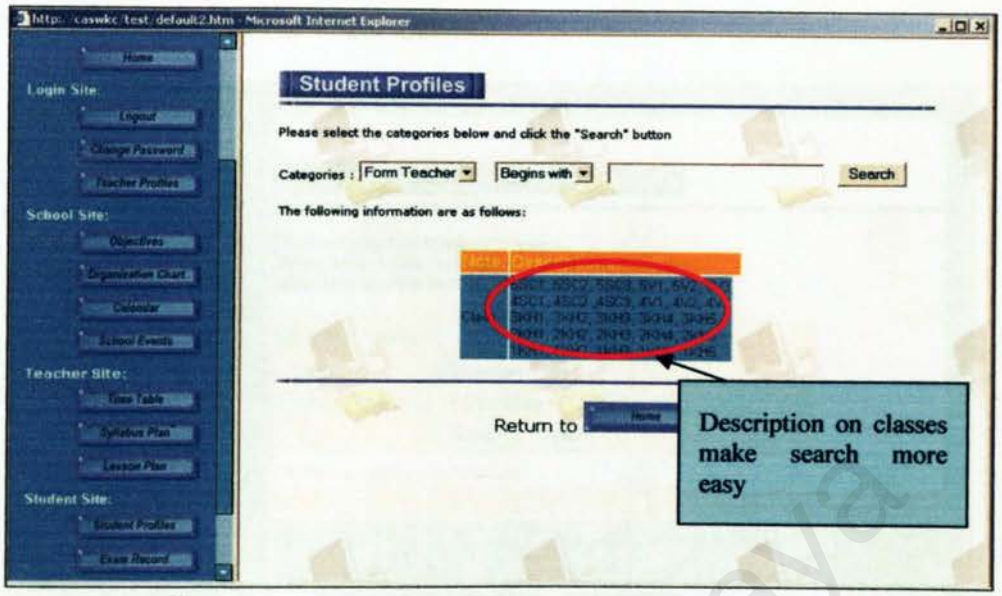
16. If not record was added / deleted, a message "No record added/deleted" will display.



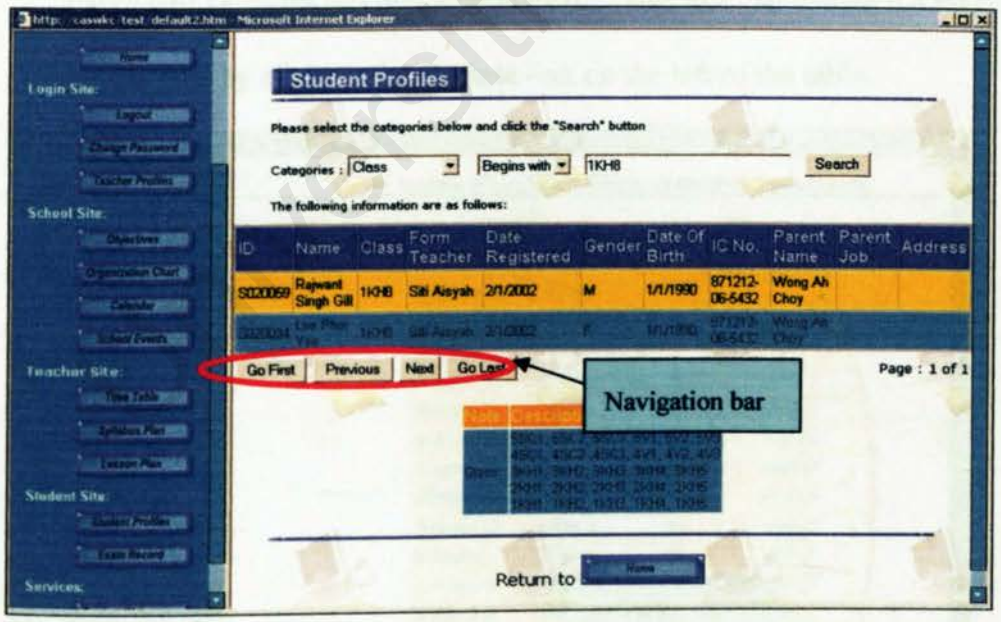
B.1.3.4 Student Site

In this **Student Site**, teachers able to search through the student profiles from the school database under the **Student Profiles** module and update the marks for students under their classes under the **Exam Records** module.

1. Under the **Student Profiles** module, teacher can search student profiles through Form Teacher Name, Student ID, Student Name, Class categories. They can also customized the search which begins with the type in text. Then, they have to click the **Search** button to begin the search.

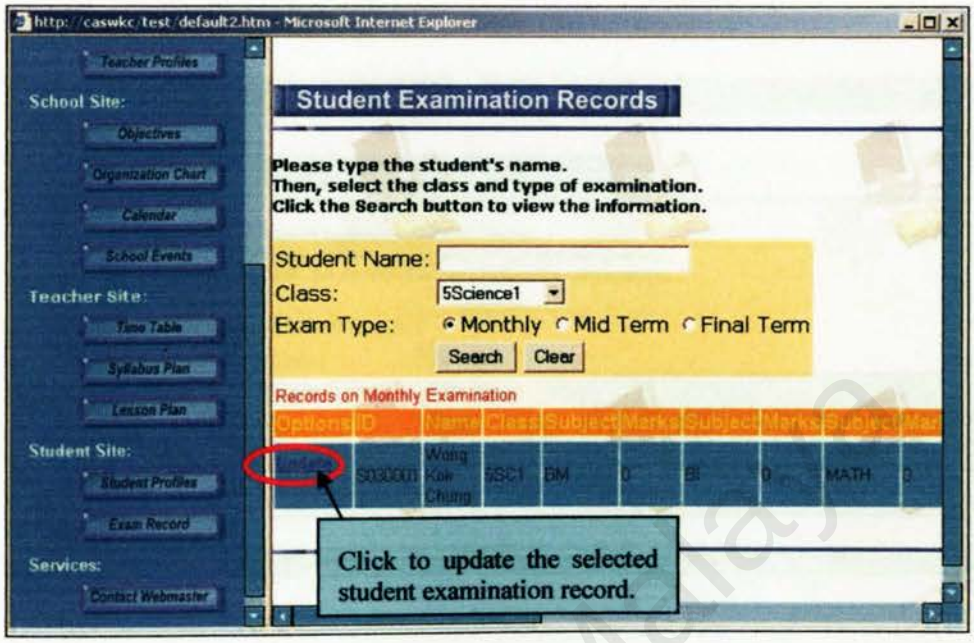


2. In order to make the search more viewable, the teacher can navigate through the table with a highlighted row and a navigation bar at the bottom of the table.

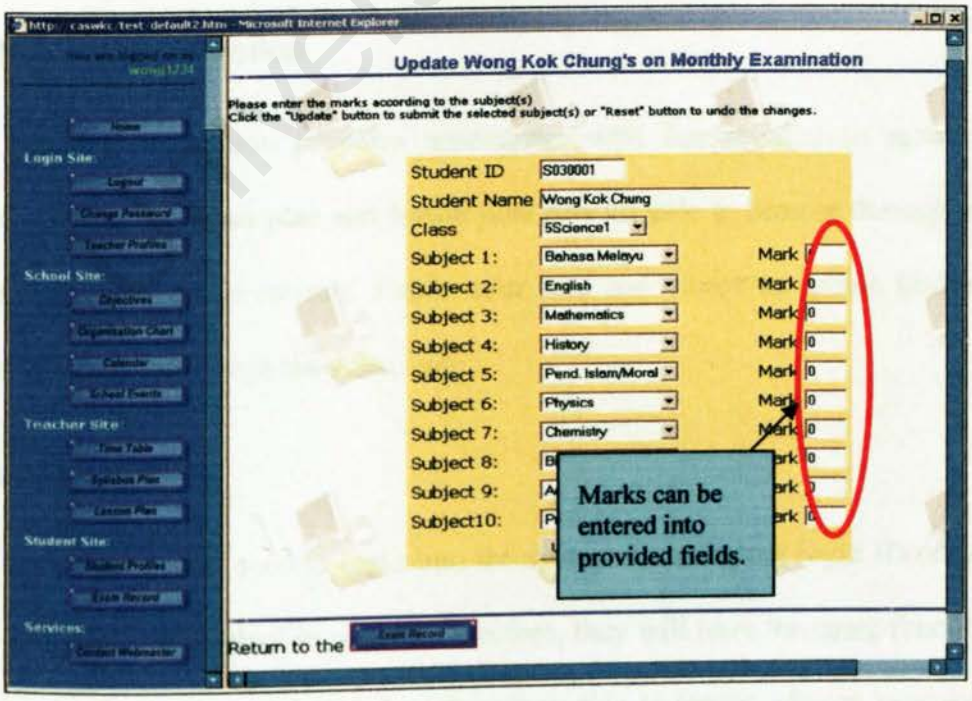


3. Under **Exam Records** module, teacher able to search for the students under her class based on Student Name, Class and Examination Type (monthly, mid and final term). She can type the relevant information into the fields and

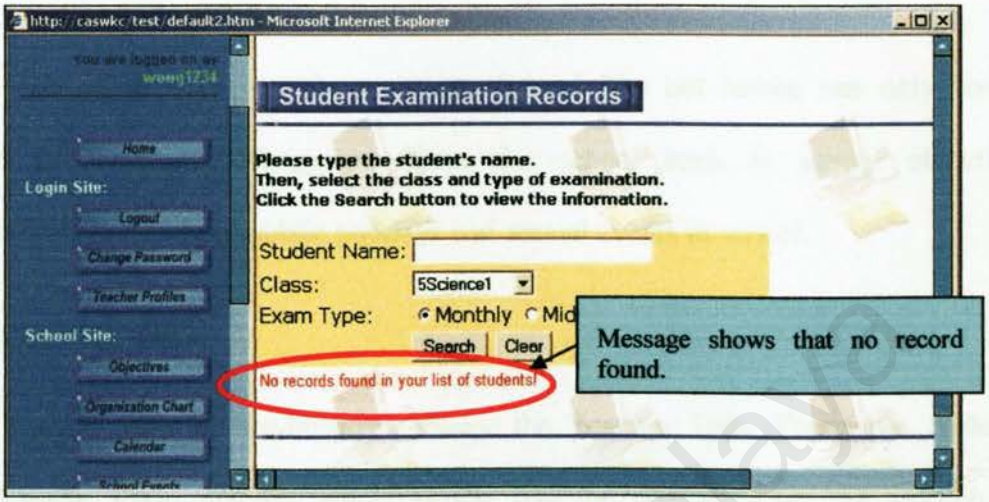
click the **Search** or **Clear** button.



4. Once the records was found, the student(s) with their learning subjects and marks will be displayed into a table form. Then, she can update the marks for the students by clicking the **Update** link on the left of the table.



5. If no record was found, a message indicating "No records found in your list of students" will display.



B.1.3.5 Services

The function provided in the **Services** is the same as the B.1.2.5 for clerk in school.

B.1.4 Headmaster Section

Headmaster Section provides headmaster with functionality to monitor and validate teachers' syllabus plan and lesson plan besides able to browse through student profiles and examination records. Headmaster also has access to update his/her own account and browse through the school site.

B.1.4.1 Login Site

Headmaster will need to login into the system by using the login ID created by school administrator or clerk in school. Therefore, they will have the same functionality as the Login Site in **B.1.2.1** for clerks where they able to logout, change password and

update their profiles.

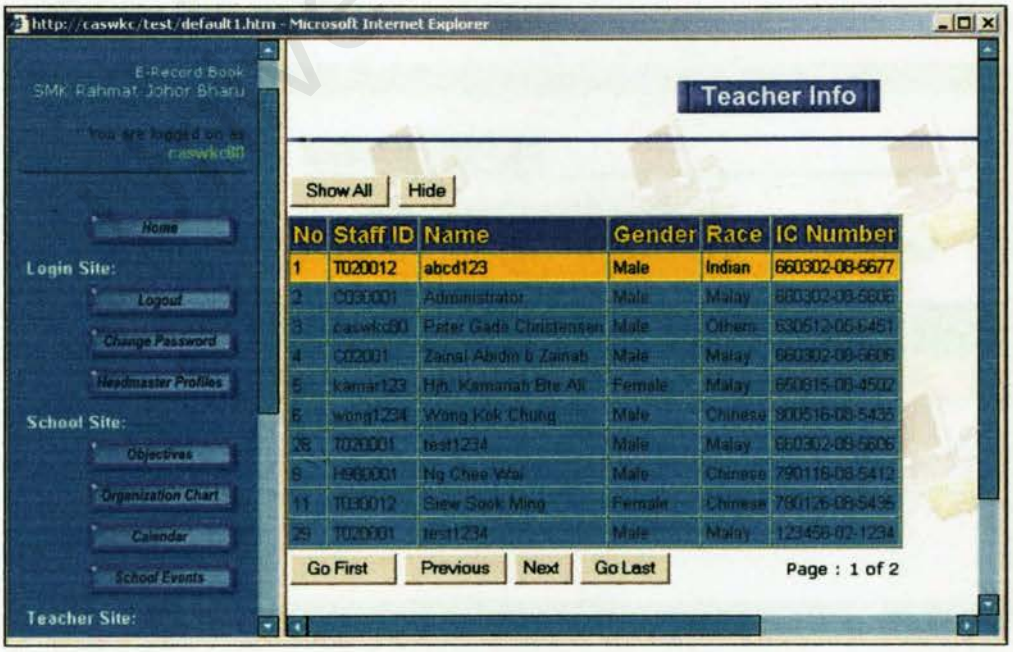
B.1.4.2 School Site

Headmaster able to gain access to School Site but he/she can only browse through the relevant information about the school such as school objectives, organization chart, school public holidays and annual events in school.

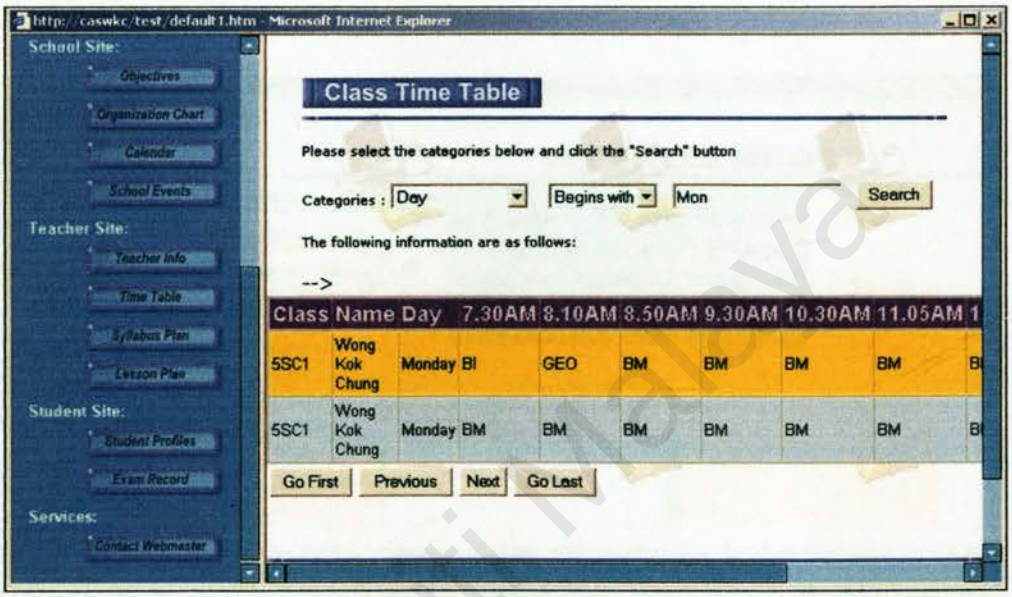
B.1.4.2 Teacher Site

Teacher Site for headmaster contains the **Teacher Info**, **Timetable**, **Syllabus Plan**, **Lesson Plan** modules where he/she has the access to browse through the information from every teacher.

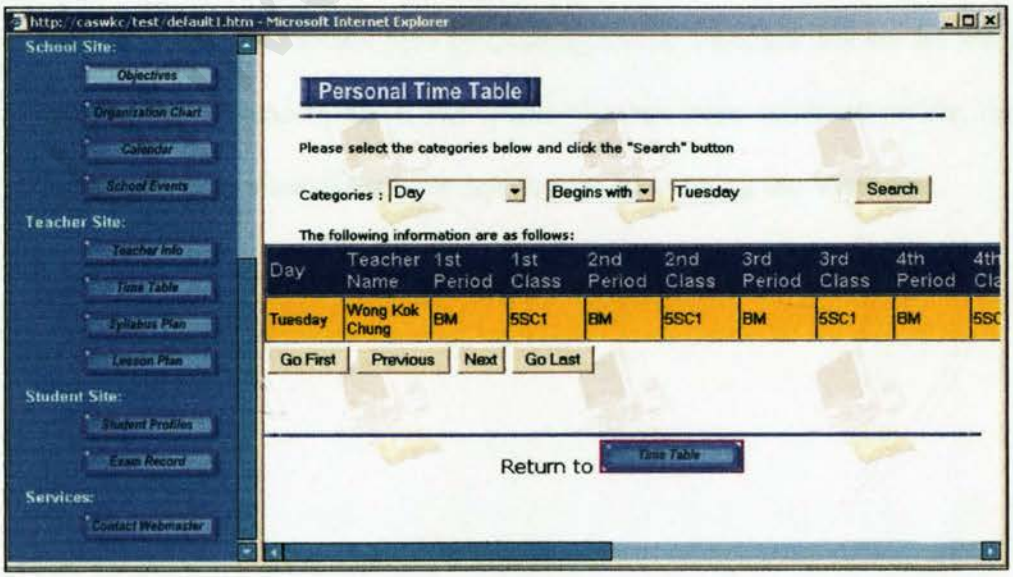
6. Under the **Teacher Info** module, headmaster can browse through teacher profiles He/she can click the **Show All** button to show all the profiles on the teachers or **Hide** button to minimize the table.



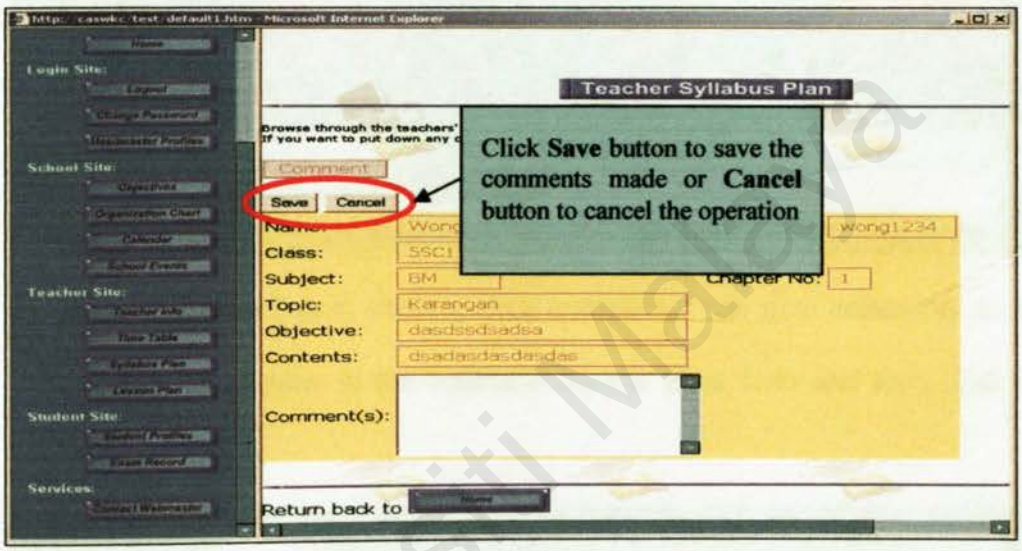
7. As for the **Time table** module, headmaster can select between the class timetable and personal timetable. Inside the **Class Timetable** page, headmaster able to search for information on class timetable based on **Teacher Name, Class** and **Day.**



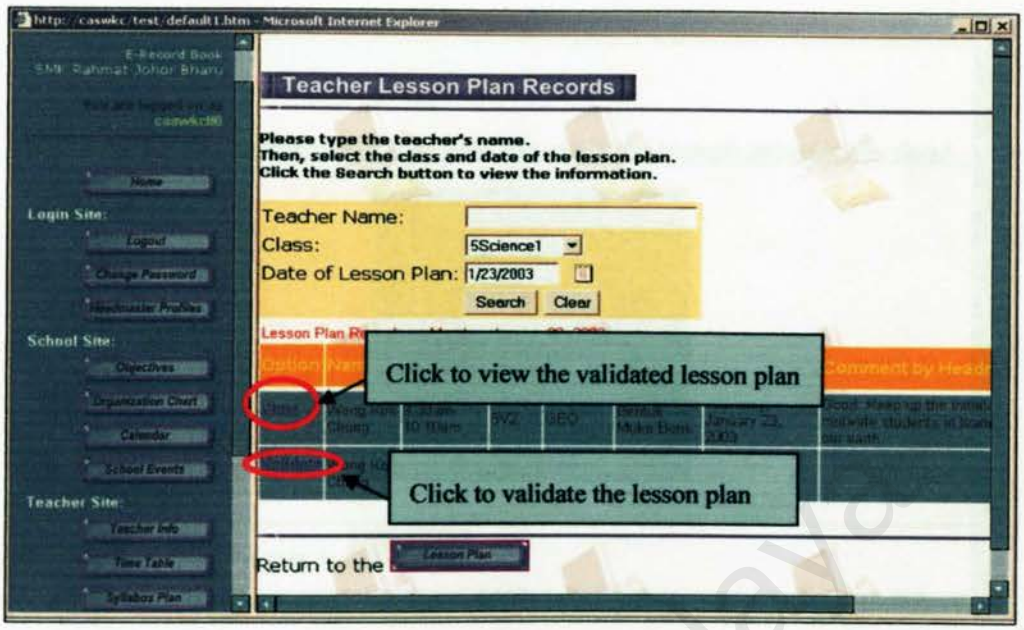
8. Inside the **Personal Timetable** page, headmaster can only search based on **Teacher Name** and **Day.**



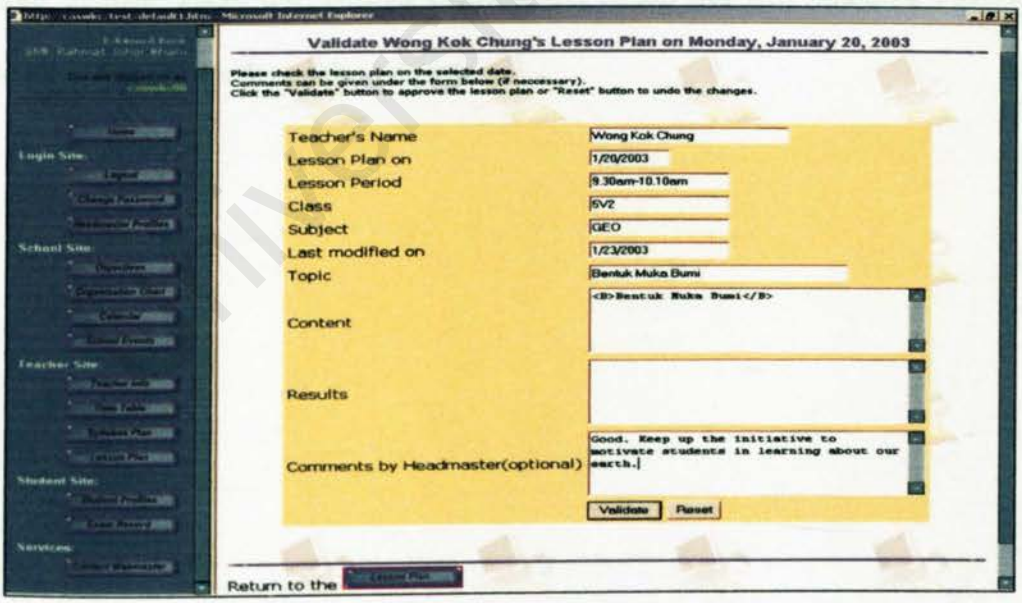
9. Under the **Syllabus Plan** module, headmaster able to monitor the syllabus plan by browsing through every syllabus plan wrote by teacher. He/she can give comments by clicking the **Comment** button and type in the comments in the field given. Then, he/she should save it for the teacher to be able to look at the comments.



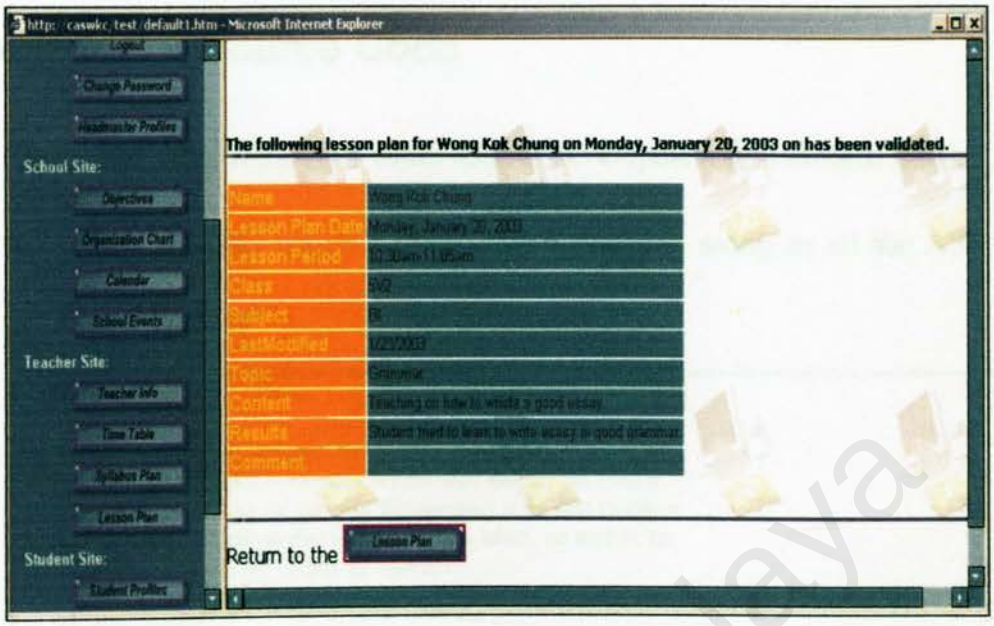
10. Under the **Lesson Plan** module, the headmaster able to search on the selected teacher based the class and date of lesson plan. If records found, he/she can validate the written lesson plan by clicking on the **Validate** link on the left of the records of lesson plan. If the lesson plan has been validated earlier, then he/she can only viewed back the lesson plan by clicking the **View** link.



11. If the **Validate** link is clicked, then headmaster can give comments on the selected lesson plan in the **Validate Lesson Plan** form and then click the **Validate** button.



12. Once the lesson plan has been validated, the status of the lesson plan will changed to "Validated" and the information on the lesson plan will be displayed.



B.1.4.4 Student Site

In this **Student Site**, headmaster able to search through the student profiles from the school database under the **Student Profiles** and **Exam Records** modules. He/she is not able to update the marks on the subjects.

Appendix C – Source Code

Below are some of the ASP codes used in development of E-Record Book.

1. Global.asa (file which provides the connection string to all the ASP files from database)

```
<SCRIPT LANGUAGE=VBScript RUNAT=Server>

'You can add special event handlers in this file that will get run automatically when
'special Active Server Pages events occur. To create these handlers, just create a
'subroutine with a name from the list below that corresponds to the event you want to
'use. For example, to create an event handler for Session_OnStart, you would put the
'following code into this file (without the comments):

'Sub Session_OnStart
'**Put your code here **
'End Sub

'EventName              Description
'Session_OnStart        Runs the first time a user runs any page in your application
'Session_OnEnd          Runs when a user's session times out or quits your application
'Application_OnStart    Runs once when the first page of your application is run for the first time by any user
'Application_OnEnd      Runs once when the web server shuts down

</SCRIPT>

<SCRIPT LANGUAGE=VBScript RUNAT=Server>
'Sub Application_OnStart
'End Sub

Sub Session_OnStart ' Runs the first time a user runs any page in ERBMS application
    'set object connection timeout


    'Create an ADO Connection
    Set objConn = Server.CreateObject ("ADODB.Connection")

    'Create a ADO Recordset
    Set objRS = Server.CreateObject ("ADODB.Recordset")

    'Specify connection string on Open method
    ProvStr = "Provider=SQLOLEDB.1;Password=caswkc;Persist Security Info=True;User ID=wkc;Initial Catalog=ERBMS;Data Source=CASWKC;"
    objConn.Open ProvStr

    Set Session("conn") = objConn

```

```
End Sub

Sub Session_OnEnd '     Runs when a user's session times out or quits your application
    'Close the data connection
    objConn.Close
End Sub

</SCRIPT>

<SCRIPT LANGUAGE=VBScript RUNAT=Server>
Sub Application_OnStart
    '==Visual InterDev Generated - startspan==
    '--Project Data Connection
        Application("ERBMS_ConnectionString") = "Provider=MSDASQL.1;Password=caswkc;Persist Security Info=True;User ID=wkc;Extended
        Properties=""Description=RecordBook;DRIVER=SQL Server;SERVER=CASWKC;UID=wkc;APP=Microsoft Development
        Environment;WSID=CASWKC;Network=DBMSLPCN"";Initial Catalog=ERBMS;"
        Application("ERBMS_ConnectionTimeout") = 15
        Application("ERBMS_CommandTimeout") = 30
        Application("ERBMS_CursorLocation") = 3
        Application("ERBMS_RuntimeUserName") = "wkc"
        Application("ERBMS_RuntimePassword") = "caswkc"
    '-- Project Data Environment
        'Set DE = Server.CreateObject("DERuntime.DERuntime")
        'Application("DE") = DE.Load(Server.MapPath("Global.ASA"), "_private/DataEnvironment/DataEnvironment.asa")
    '==Visual InterDev Generated - endspan==
End Sub
</SCRIPT>
```

2. Holiday1.asp (ASP codes using session to refer to connection object and the use of Recordset)

```
<%
'Reference the Session connection variable
objConn = Session("conn")
Dim rsHoliday
'select all data from RB_UserProfiles
strSQL = "SELECT * FROM PE_Calendar Order by CalDate"

Set rsHoliday = Server.CreateObject("ADODB.Recordset")

rsHoliday.Open strSQL, objConn
%>
<table align="center">
<tr>
<%
For i=0 to 2
    Response.Write "<td class=header>"
    If rsHoliday.Fields(i).Name = "ID" Then
        Response.Write "ID"
    ElseIf rsHoliday.Fields(i).Name = "CalDate" Then
        Response.Write "Day & Date"
    ElseIf rsHoliday.Fields(i).Name = "Holiday" Then
        Response.Write "Holiday"
    End If
    Response.Write "</td>"
Next
%>
```

```
</td>
<%
Do While Not rsHoliday.EOF
    Response.Write "<tr>"
    For i=0 to 2
        Response.Write "<td class=body>"
        Response.Write "<font face=san serif size=2>"
        If rsHoliday.Fields(i).Name = "CalDate" Then
            Dim varDate
            varDate = rsHoliday.Fields(i).Value
            Response.Write FormatDateTime(varDate ,1)
        Else
            Response.Write rsHoliday.Fields(i).Value
        End If
        Response.Write "</font></td>"
    Next
    Response.Write "</tr>"
    rsHoliday.MoveNext
Loop
'close the recordset
rsHoliday.Close
%>
```

3. Logout.asp (session is set to nothing once the user has logout from the system)

```
<%
Response.Expires = -1000  'Makes the browser not cache this page
%>

<html>
<head>
<meta http-equiv="Refresh" content="1; URL=login.asp">
</head>
<body bgcolor="aqua" leftMargin="0" background="images/ebook.jpg" topMargin="0" marginheight="0" marginwidth="0">
<div align="center">
 <td valign="top" align="middle"><!--begin the content of page-->
    <p> </p>
    <p> </p>
    <p> </p>
<%
'set object connection timeout
Session.Timeout = 1

Set objConn = nothing
Set Session("UserID") = Nothing

Session.Abandon
%>
<table>
<img name="line_header" src="images/line.gif" width="100%" height="3px">
<center>
<h3>
<font face=arial color=red>Thank You.<br>You have successfully logout</font>
</h3>
</center>
<img name="line_header" src="images/line.gif" width="100%" height="3px">
</table>
</td></TR></TBODY>
```

4. unlock.asp (usage of ADO Command object to set value for counter and status in RB_Login table)

```
<%
'Reference the Session connection variable
objConn = Session("conn")

' Create the remove command and set its properties
Set cmd = Server.CreateObject("ADODB.Command")
cmd.ActiveConnection = objConn

For i = 1 to Request.Form ("UserNo").Count
    cmd.CommandText = "Update RB_Login Set Counter = 0, Status = 1 Where No = '" & Request.Form("UserNo")(i) & "'"
    ' Execute the command on the Active Connection
    cmd.Execute
Next

'Output some feedback to the user to let them know which products
'were removed from the inventory database. %>
<br><br>
<img name="line_header" src="images/line.gif" width="100%" height="3px">


<h3>The following user account(s) have been unlocked from the database:</h3>

<% FOR EACH user IN Request.Form("UserNo")
    Response.Write "User #" & user & "<br>"
  NEXT %>
```

Appendix D – Interview Questions

1. What is the usage of a Buku Rekod Mengajar Guru Sekolah Menengah for teachers?
2. What are the requirements in writing a record book?
3. What is the format in preparing a good and standard lesson plan?
4. How are the marks and grades of subjects being recorded in the record book?
5. Who are the people in the school that have been given the authority to access the record book?
6. What is the level of security for the information recorded in the record book?
7. Can the teachers be allowed to update their record book other than school hours?
8. Do teachers need to bring along their record books when they are attending classes?
9. Is the record book need to be validate by the school headmaster?
10. When do the teachers hand out the record book to the headmaster for validation?
11. How long will the validation process take in your school?
12. What do you think of the existing manual record book in the school?
13. Do you think the existing record book has fulfilled the requirements of the teaching and learning process in the school nowadays?
14. What is your suggestion in improving this record book?
15. How do you find an electronic record book management system will help the teachers and headmaster to manage the record book?
16. Are the teachers in the school prepared to learn in using the electronic record book management system when it is implemented?

References

A. Web sites

1. http://www.thinkwave.com
2. http://www.a2zwaresolutions.com
3. http://www.gradesandplans.com
4. http://www.moe.gov.my/kpm/sekolah.htm
5. http://www.mdc.com.my
6. http://www.teacherfocus.com
7. http://www.activeserverpages.com
8. http://www.itknowledge.com
9. http://www.microsoft.com

B. Books

1. *Buku Rekod Mengajar Guru Sekolah Menengah* for Sekolah Menengah Kebangsaan Sri Rahmat, Johor Baharu.
2. Pfleeger, Shari Lawrence. (1998). *Software Engineering – Theory and Practice.* 2nd ed. Prentice Hall.
3. Scott, Kendall. (2001). *UML Explained.* Addison-Wesley.

4. Deitel, H. M. et al. (2000). *Internet & World Wide Web – How To Program.* 2nd ed. Prentice Hall.

5. Bahrami, Ali. (1999). *Object Oriented System Development.* Irwin/McGraw Hill.

6. *Buku Panduan Guru dan Kakitangan* for Sekolah Menengah Kebangsaan Sri Rahmat, Johor Baharu.

7. *Titian 2000* for Sekolah Menengah Kebangsaan Sri Rahmat, Johor Bharu.

8. Laudon, Kenneth C. et al. (2001). *Essentials of Management Information Systems – Organization & Technology in the Networked Enterprise.* 4th ed. Prentice Hall.

9. Homer, Alex et al. (1997). *Active Server Pages.* Wrox Press Limited.

10. Johnson, Scot et al. (1997) Special Edition Using Active Server Pages. Que

11. Mastering Web Application Development Using Microsoft Visual InterDev 6.0. Microsoft Publication.

C. Books from the Internet

1. Tungare, Manas. (2000). *A Practical Guide To Microsoft Active Server Pages 3.0.* Manas Tungare.

2. Dalton, Patrick. (1997). *Microsoft SQL Server Black Book.* The Coriolis Group.

3. Whalen, Rozanne M. et al. (2001). *Microsoft SQL Server 2000 – Design.* Element K Press.