University of Malaya

Faculty of Computer Science & Information Technology

School Management System

Name : Tan Swee Ling

Matriks No : WEK990103

Supervisor : Mr. Chiew Thiam Kian

Moderator: PM Dr. Lee Sai Peck

PROJEK ILMIAH TAHAP AKHIR

WXES 3182/SEMESTER 2

Session 2001/2002

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

The purpose of this project is to develop a School Management System that utilized information technology to help to manage the school. As a school can be considered as an organization that was head by a principal, a well-managed school is important to make a school a conducive learning environment.

School Management System is a client/server application that is develops mainly for secondary school. The aims of developing this system are to help the teachers manage resources and processes. It is also aims to reduce the workload of teachers by automating some of the processes.

This School Management System will be developing using Visual Basic and SQL Server 2000. It is hoped that this system will be beneficial to teachers as well as students.

ACKNOWLEDGEMENT

I would like to take this opportunity to express my gratitude and thanks to my supervisor Mr. Chiew Thiam Kian for his advice and guidance in completing this project proposal. I would also like to thank my moderator, Dr.Lee Sai Peck for her valuable comments and suggestions.

I would like to extend my gratitude to Puan Malligah Sivapalan, Penolong Kanan Hal Ehwal Murid of Sekolah Menengah Convent Bukit Nanas, Kuala Lumpur and Puan Selvanathan, a former teacher of Sekolah Menengah Bangsar, Kuala Lumpur. They had provided valuable information and suggestion for this project. I am most grateful for their time and considerations.

CHAPTER 1
INTRODUCTION

1.1 Introduction

The government is placing great importance on the use of Information Technology (IT) in education. This can be seen as a lot of schools concentrate on using IT as teaching aids and learning tools [19]. But incorporating IT in school management is still not encouraging as many schools in Malaysia still use manual system in their management.

The purpose of this project is to develop a School Management System that helps the staffs and teachers to do their daily tasks, to keep students records and to have a better attendance management for students. With the increasing number of students each year, management of school will become more complex. Therefore, school need to be more efficient in their management, one of the ways is to implement a computerized system.

School is lead by a principal and is help by a few administrative staffs and teachers. Therefore, the responsibility of managing a school generally lies on the shoulder of the principal. The principal have to manage human resource, financial, monitoring students' performance, check on absentees and et cetera. Effective management will help the principal fully utilized the available resources to create an effective learning environment.

A good school management is important as it helps the principals, teachers and staffs in decision-making. A good decision is a well-informed decision. Obtaining information or data such as students' achievement data easily and correctly can helps teachers to plan their lesson and expectation in order for students to reach their full potentials. Besides that, it helps to save teachers' time and teachers will have more time to concentrate educating the students.

Therefore, the aim of this project is improve the school management and solved the problems faced in the current manual system.

1.2 Problem Definition

A school can have a large number of students, approximately 1000 to 2000 students depending on the size of the school. Therefore, to manage such large number of student is not an easy task. Furthermore, teacher tasks are not just limited to teaching; they have to continuously checking on student performance and discipline.

In school, daily administrative tasks such as calculating examination marks, keeping tracks of student attendance are still done in the manual way. This actually increased workloads on teacher. Longer time is needed to collect and process data and information obtained might be inaccurate due to human-made mistake. Teacher will have insufficient time to analyse data and do planning [10].

1.3 Objective

This project will develop a school management system with the following objectives.

1) Develop a system that utilizes information technology to increase efficiency of school management, administration and communication.
2) To reduce workload of teachers so that they can concentrate on teaching.
3) Help to manage resources and processes.
4) Support decision making for principal and teachers so that they can effectively plan their lesson.

1.4 Project Scope

The project mainly focused on the development of a School Management System for secondary schools in Malaysia. This system will concentrate on the management of students. This project will concentrate on the following areas:

Students Information System

- Students' records can be view, modify and update.
- An option to generate report on students.
- Provide archiving and retrieving where data can be save to local file system for backup purpose and data can retrieve when needed.

Teachers and Staffs Information System

- Teachers and staffs record can be view, modify and update.
- An option to generate report on teachers or staffs.
- Provide archiving and retrieving where data can be save to local file system for backup purpose and data can retrieve when needed.

Document and Data Processing

- Process students' test and exam results and display in the form of report card.
- Analysis of students' exam result.

Attendance Management

- Keep track of students' attendance in each class.
- Generate attendance reports.

The system will also have a login system for authorised access of teacher and staff.

1.5 Methodology

System Development Life Cycle (SDLC) will be use to develop this School Management System. The SDLC is consist of the following phases:

- System Planning
- System Analysis
- System Design
- Construction and Implementation Phase
- System Evaluation and Maintenance

1.5.1 System Planning Phase

At this phase, the school management problem is defined. Preliminary investigation is done to define the project scope, constraints and project schedule. With the initial scope of the project, a project schedule is prepared using a Gantt Chart to monitor the progress of the project.

1.5.2 Analysis Phase

This phase will define what the system must do. A study and analysis of the existing system will be done. By studying the existing system, it will provide a better understanding of the problems that motivate this project.

Next, requirement analysis is done. Information collected from the analysis of the current existing system, will be analyse to determine the requirements and priorities.

With the requirement statement, there are numbers of ways to design the School Management System to fulfill those requirements. Therefore, a system solution will be identified and the solution will be analysed for feasibility at this phase [9].

1.5.3 Design Phase

This phase will transform the requirement specification from the analysis phase into design specification for developing the school management system.

At this phase, Entity Relationship Modeling will be use to design the database of the system. Data Flow Diagram will be use to design the process of the system.

1.5.4 Construction and Implementation Phase

At this phase, database, application programs, and system interface will be construct using programming language. Individual components and overall system is tested at this phase. Once tested, the system will be ready for implementation.

1.5.5 System Evaluation and Maintenance

Once the system is implemented, the system needs to be evaluated and maintained to make sure the system is efficient and meet the requirements that been defined.

1.6 Research Methods

I have interview Puan Malligah Sivapalan, Penolong Kanan Hal Ehwal Murid of Sekolah Menengah Convent Bukit Nanas, Kuala Lumpur. Puan Malligah Sivapalan had explained to me some important aspect of management of the school. Some forms and documents were collected during the session.

I have also interview Puan Selvanathan, a former teacher from Sekolah Menengah Bangsar, Kuala Lumpur.

Another method of gathering information is through research and site visits. Information regarding problems faced in the management of school and other related

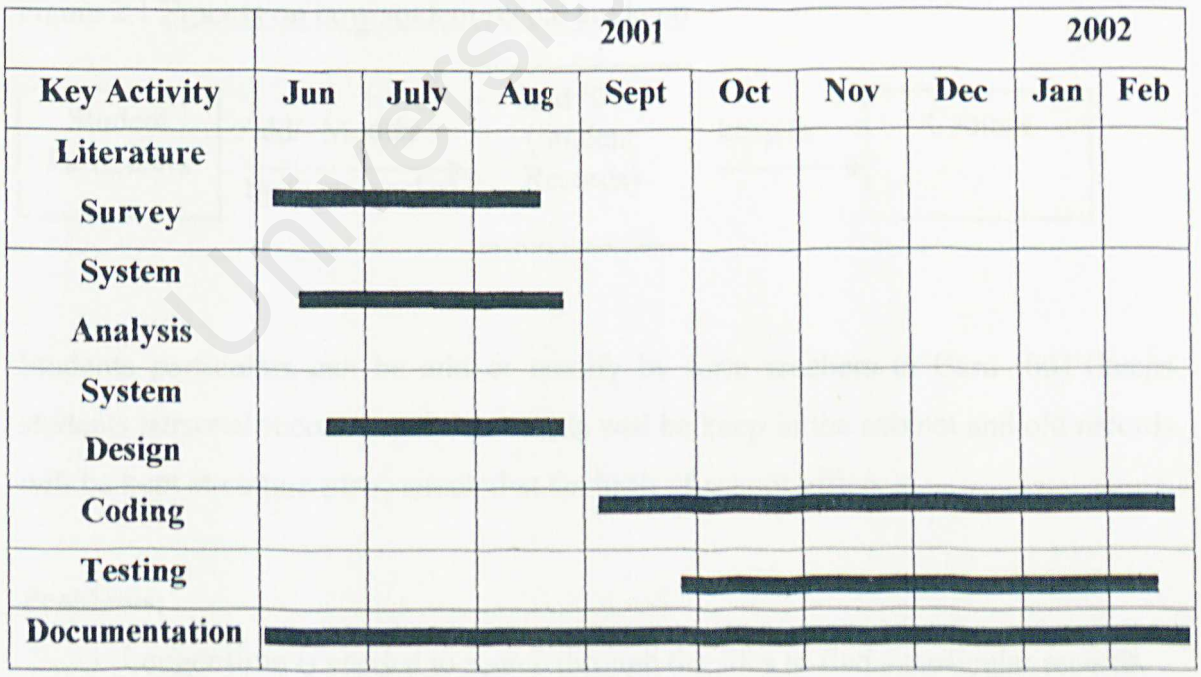
information are gathered from on-line journal, books, and reference books and by exploring the Internet.

1.7 Expected Outcomes

Below are the expected outcomes to be achieved:

1) All the students, teachers and staffs' records can be keep in a central database.
2) Analysis of students result can be done faster. Students' performance can be access easier.
3) Better management of attendance of student. Analysis of attendance is easier and faster.
4) Repetitive tasks can be reduced. Users can concentrate on more important tasks.

1.8 Proposed Project Schedule



Legend Task

Table1.1 Project Schedule

CHAPTER 2
LITERATURE REVIEW

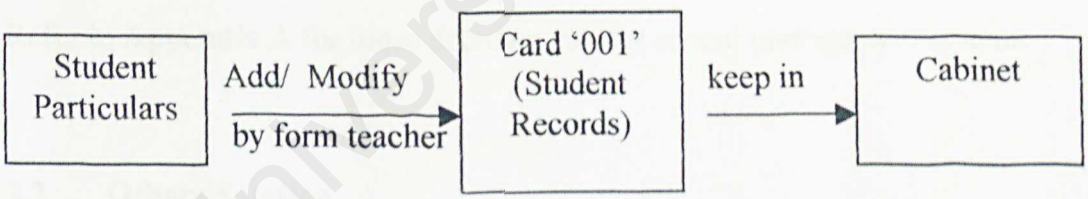
As most of the secondary schools in Malaysia still use manual system in their management, this chapter will review the current manual system and solution that are available in other countries.

2.1 Background Survey

School is an entity or organisation, which include the process of managing, learning and educating. Managing and educating are two interrelated process that are needed to realize the objectives of a school [1].

By reviewing the current school management system, most schools still use manual file system to keep their students, teachers and staffs records.
For example, the Figure 2.1 below shows how the students' records are kept in the schools.

Figure 2.1 Process on how student record are keep



Students particulars can be add or modify by form teachers in Card '001'(keeps students personal records). All the records will be keep in the cabinet and old records will be kept at certain place situated at the back of school office.

Problems:

1) Longer time is needed to search through the files to find a particular records.
2) Retrieval of old records as the old records are kept at the back of the office.
3) School office will eventually have insufficient space to keep the files as the numbers of files increase each year.

4) Data and information might be inconsistent and outdated.
5) Lacks of security as the confidential information are kept in the cabinet. Even if the cabinet is locked, a lock can be easily tampered.

Students' test and examination marks as well as their attendance are recorded in a book called Record Book. Every teacher will have a Record Book. The test or examination marks will written to report card where grading will be assigned. The attendance will also be written to the report card.

Problems:

1) Repetition of tasks. For example, tasks are repeated when teachers have to keep the marks of their students in the Record Book and then they have to write it again in the academic reports.
2) Longer time to process data and analyse students result as the task is done manually.
3) Does not provide a quick overview of attendance. Longer time is needed to analyse and process the attendance.

Refer to **Appendix A** for more detail on current school management system.

2.2 Others Solution

2.2.1 PowerSchool

PowerSchool is a web-based school management system, which built in the thin-client technology. Powerschool enables the schools to record, access and manage student information at real time [20].

Among the features of the system are:

1) **Administrative Tool** – teachers or administrators can manage students' records, class scheduling, and reporting.

2) **Communication Tool** - parents and students can communicate with teachers in real time and students can check on their grade, assignment and attendance.
3) **Class Management Tool** – teachers can take attendance and generate report card.

2.2.2 SchoolSpace

SchoolSpace is another browser-based school management system. Every teacher can customized the homepage to their needs whenever they log-in. The homepage have electronic bulletin board, e-mail, classes and upcoming assignment [21].

Others feature include:

1. **Attendance** - attendance of the students can be easily tracked as picture and the name of students are arrange according to the seating of the students in the class.
2. **Student Information** – students' information such as parent information, medical record and emergency contact can be view.
3. **Grade book** – students' achievement and academic result can be view.
4. **Lesson Plans** – provide a template for teachers that are easy to create and share lesson plans.
5. **Administrative** – set the grading policy required by school and generate reports and graphs for student performance.

2.3 Conclusion

By reviewing the current manual system, a better understanding of the problems and needs of the management of schools have been achieved. Although there are other solution that are being promoted in the Internet, the systems are develop by others countries. In fact, the tasks of school management in Malaysia such as keeping students information, checking on attendance and generate reports on students performance are provided by the available systems.

But, the available systems might not meet the requirement and the needs of schools in Malaysia, as Malaysia Education System is different from other countries. There is some required information that is not captured by the available system.

Therefore, the proposed school management system is a system that will be able to provide the basic needs of stakeholders in school and solved problems in current school management (review in Section 2.1).

The proposed system will be a client-server application because this networking approach has proven to be cost-effective way to share data between multiple users. The school main stakeholders, who include the principal, administrative staffs and teachers will need to access data at the same time to perform their tasks. Therefore, this approach will provide a more effective school management system.

CHAPTER 3

SYSTEM ANALYSIS

3.1 Requirement Analysis

3.1.1 Functional Requirements

Functional requirements are set of functions that are required to be included in the system. Functional requirement include statements of services the system should provide and how the system should react to particular input [8].

The school management system will have the following functional requirement.

Authorisation and Authentication Function

The major function in this module is allows authorise user to access the system. It will provide the following function:

a) Login Function

This function will allows users to login and authenticates user identity.

b) Change Password

This function allows users to change their password.

Teacher and Staff Information System

In this module, the teacher and staff record will be maintain separately. There are

a) Teacher Record Maintenance

b) Staff Record Maintenance

Teachers and staffs information can be saved, view and modify. Every teachers and staffs is allowed to maintain and view their own records. Modification can be done only for certain information such as address and telephone number. Only system administrator can add and delete teachers and staff information.

Students Information System

a) Students Record Maintenance

The student records will keep the following information.

1) Personal Particulars
2) Parents Information
3) Guardian Information
4) Emergency Contact
5) Medical Record

In this module, form teacher and administrator can add, view and modify their student records. Modification can be done only for certain information such as address and telephone number.

Class Management

a) Attendance Management

Provide function that allows attendance entry by form teacher according to class. Form teacher is also allowed to view and modify student attendance.

b) Activity

Provide function that allows form teacher to define the activity join by each student in the class.

c) Subject

Provide function that allows form teacher to define the subject taken by each student in the class.

d) Report Book

Provide function that allows students result entry by form teacher according to subjects. Teachers can view the analysis of student result that, include percentage of student fail, percentage of student pass, average mark for each subject according to each class.

Form teacher can view and modify student result.

Subject Module

Provide function that allows subject teacher to enter student result. Teacher can view and modify student result. Provide analysis of student result according to the subject and class teaches by the subject teacher.

Analysis of Result According to Each Form

Analysis of student result for each subject can be view according to each form. Data that should be display include total of student who take the subject, total student

absent for the exam, percentage of failure, percentage of passes, highest mark, average mark. Percentage of failure for each subject in a form should also be present in graph. This module should also include a search function where users can search for student who obtained the highest mark for each subject in each form.

Reporting Tool

The reports to be generated will be in pre-defined format and can be generated in regular or ad-hoc basic.

a) Reports

Among the reports can be generate are:

1) Report on student information, staff information and teacher information.
2) Marks and grade obtained by students
3) Number and percentage of students passed or failed per subjects according to each class.
4) Number and percentage of students passed or failed per subjects according to each form.
5) Total days of absent for each student in each class.

Reports can be view and print.

Old Records Maintenance

All the old student, teacher and staff record will be maintain by system administrator.

a) Old Teacher and Staff Record Maintenance

Provide archiving and retrieving of old staff and teacher record when needed.

b) Old Student Record Maintenance

Provide archiving and retrieving of old student record when needed. Archiving and retrieving can be done according to student academic year.

System Administration

a) System Setting and Configuration

The administrator can set and configure the system according to the school need. They can set and define:

1) How many class in each form and the name of the class.
2) The extra co-curriculum activities that are available in their school.

3) The subjects teach in the school.
4) The range of mark for each grade depending on each form. For example, they can set grade A for the mark of 75 to 100, grade B for 65 to 74 and so on.
5) Type of Exam.

b) Teacher

The administrator can set and define:

1) The form teacher for each class.
2) Subject taught by teacher.
3) Teacher advisor for each activity
4) Class that teach by teacher for each subject.

c) Create Login Function

This function will allow administrator to add and remove users. It also allows the administrator to reset the user password in case that the users forget their password.

3.1.2 Non-Functional Requirements

The following are the non-functional requirements for the school management system.

1) **User-friendliness** – The system need to have a user-friendly interfaces so that the users will find it easy to use the system for their daily tasks. The screen design needs to be consistent and have the appropriate error handling.

2) **Reliability** – Reliability is defined as the ability of the system to behave consistently in a user-acceptable manner when operating within the environment. The system needs to be reliable so that the integrity of information is maintained and the system components are available when needed.

3) **Security** – As some of the information and data in the system are confidential, the system need to have good security by allowing only authenticate and authorise users to use the system.

4) **Maintainability and Expandability**– The system developed need to be easy to maintain so that the maintenance cost can be reduced in long term. The system also needs to have ability to expand for future enhancement.

5) **Efficiency** – The system need to be efficient to provide a good respond time and throughput so that it does not cause any delay in processing user request. The system also needs to be able to support the workloads.

6) **Robustness** - Under the unexpected circumstances such as given an improper data, the system need to be robust to handle and avoid system error

3.2 Consideration of Development Tools

3.2.1 Microsoft Visual Basic 6.0

Visual Basic is the powerful programming tool for developing Windows based application. The "Visual" part refers to the method used to create the graphical user interface. The "Basic" part refers to the programming code part, stands for Beginner's All-Purpose Symbolic Instruction Code [2]. The main features of Visual Basic are:

1) Visual design

Visual Basic provides the built-in capability to create user interface with the click of a mouse. Therefore, the graphical interface can be created easily and more time can be spent on the features and capabilities of the application to be develop [16].

2) Event-driven programming

Visual Basic provides event-driven programming where programmer can program how each control reacts to user actions, such as click of the mouse, keystrokes and so on [5].

3) Database Access features

Visual Basic supports database access tools such as DAO and ADO. DAO allowed programmer to access Microsoft Access databases only.

With ADO, Visual Basic application sees three objects:

- Connection Object, which establishes a connection to the database.
- Command object, which executes commands against the database.
- RecordSet object, which holds the records retrieved from the database or the records to be updates on the database [5].

4) ActiveX Technologies

ActiveX technologies allow the functionality provided by other applications to be use in own program. Besides that, applications and objects created can be automated using Visual Basic.

3.2.2 Microsoft Visual C++

The visual part of Visual C++ is virtually the same as the Visual Basic in terms of ease of creating graphical user interface. But the programming language use is C++ language, which is more powerful. Visual C++ provides numerous built in classes and support object-oriented programming [7]. The main features of Visual C++ are:

1) Microsoft Foundation Class (MFC) and ActiveX Template Library (ATL)

Wizards such as MFC AppWizard and the ATL COM AppWizard provided in Visual C++ will help to generate the basic source files for a variety of different types of Windows programs. The MFC and ATL libraries provides pre-written classes, as well as supporting code, that can be added to source files to handle the Windows programming tasks specific to the logic of a program. Both can be used to write ActiveX control.

3.3 Consideration of Database Server

3.3.1 Microsoft SQL Server 2000

Microsoft SQL Server is a high-performance client/server relational database management system. It is designed to support high-volume transaction processing as well as data warehousing and decision support application.

Among the advantages of SQL server is:

1) Easy to use as SQL server provides graphical management tools.
2) Support English Query. Instead of using complex SQL queries, user can create applications that accept natural language queries and question written in plain English by using English Query. This feature allows developer to provide easy access to relational database for all types of end users [17].
3) Support Full-text Search. This feature allows developer to provide access to structured and unstructured data in a uniform manner. This means that user can search through text stored in the database even when the text is contained in a formatted document such as Microsoft Word document [17].
4) Support user defined functions that allow code reuse and simplified development. This help to save time as routinely used logic can be incorporate in the user-defined functions [17].

SQL Server 2000 can runs on Microsoft Windows NT 4 or Microsoft Windows NT 4 or Microsoft Windows 2000 Server-based networks.

3.3.2 MySQL

MySQL is a popular Open Source database provided by MySQL AB. MySQL AB is a commercial company that builds its business by providing services around the MySQL database. Open Source means that anyone could study the code and change it according to their needs. MySQL can be downloaded free from the Internet [18].

MySQL is a relational database management system that is very fast, reliable and easy to use. It was developed to handle a very large database much faster and successfully use in highly production environment.

The main features of MySQL are:

1) Fully multithreaded. This means that it can use multiple central processing units if available.
2) Works on different platforms. It supports Linux and Windows platform.
3) SQL functions are implemented through a highly optimized class library so that it can be as fast as possible.
4) Allow mixing of tables from different databases in the same query.

3.4 Client/server Architecture

3.4.1 Overview

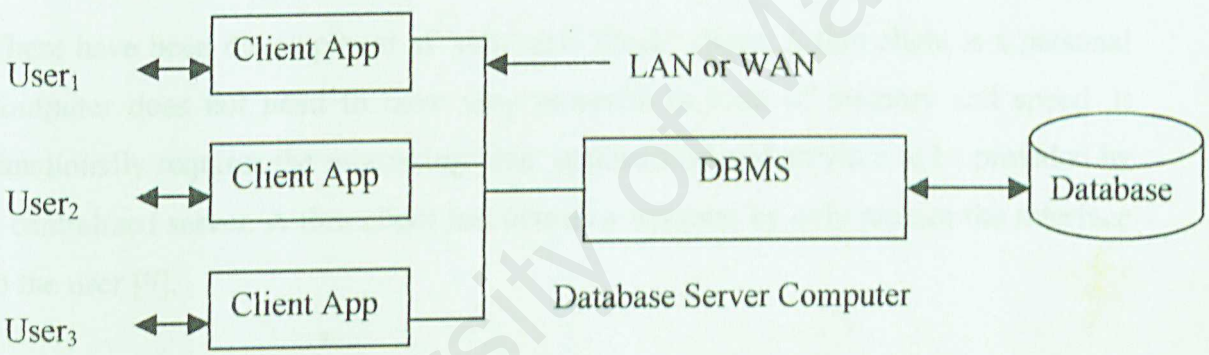


Figure 3.1 Client/Server Architecture

Client/server is a computational architecture that involves client processes requesting services from server processes. Servers manage resources such as files, database and printers. Clients will request for access to these managed resources. Server is said to have serviced the client when it has fulfils the request of a client [12].

Client computer can be any combination of personal computer or workstation that provide user interface to user. The server provides a set of shared user services to users.

Figure 3.1 shows an example in which users have their own client computers. Each user processes own application on each computer. Another computer is the database

server. The user interacts with the database through a program called Database Management System (DBMS), informally known as front end [6]. The client and the server can be connected through local area network (LAN) or wide area network (WAN) [4].

Client-server architecture is important to the development of multi-user databases [14]. For instance, when database users are accessing data or information from a database, they will affect other concurrent processes and activities in the same databases. Client server database systems are design to prevent this problem. The sever will process the request from client accordingly and make sure that the referential integrity is maintained.

3.4.2 Thin and Thick Client

There have been development of 'thin' and 'thick' client. A thin client is a personal computer does not need to have very powerful in term of memory and speed. It functionally requires the processing time, applications and services to be provided by a centralized server. A thin client just acts as a terminal by only present the interface to the user [9].

A thick client is a personal computer or workstation that is more powerful in term of speed, memory and storage capacity. Thick clients are capable of storing and execute their own applications [9].

3.4.3 Two-Tier Client-Server Architecture

In the 2-tier client-server architecture, the actual database is store on the server. All the data manipulation commands such as SQL statements to create, read, update and delete records are execute on the server. The server can be programmed to implement business rules that are better suited to run on the server than on the client.

The client that still must be use is the fat client. All the logic and processing to support the actual application and rules need to process on the client. Examples include data analysis, calculations and so on. This logic is usually written in programming language such as Microsoft Visual Basic and Visual C++. Those program must compiled and execute on the client.

Advantages:

1) Reduced network traffic as only the database request and database that are needed is transported between the client and server.
2) Database integrity is easier to maintain. Only the records required by the client is locked. Other clients can still concurrently work on other records in the same table or database.

Disadvantage:

1) Application logic must be duplicated and maintained on all the clients. Whenever there is a need of upgrade, upgrading must be done on all the clients to ensure all of them get the latest release.
2) Provide no flexibility of partitioning. Once the application is develop, it is not easy to distribute some of the functionality of the program from one server to another.
3) Two-tiered systems will suffer performance problems associated with inefficiency of executing all the application logic on the clients, when the number of clients grows.

However, the 2-tier client-server approach has proven to be effective in solving workgroup (12 to 100 users interacting on the local area network) problems [22].

3.4.4 Three-tier Client /Server Architecture

Three–tier Client/Server approach is more suitable for large organization where there are a large number of users.

In the three-tiered client/server architecture, a middle tiered is added between the system user interface in the client environment and the database management in server environment.

In this approach, most of the application logic is move from the client to a shared or host server such as the transaction server, application server or message server. By moving the application logic to the host server, that logic needs only to be maintained in that host server instead of all the clients. The client will be used basically for presentation services.

Three-tiered client/server logic can be written and partitioned across multiple servers using languages such as Microsoft Visual Basic and Visual C++ in combination of a transaction monitor .

Advantages:

2) Simplifies client configuration and management, as clients only need to execute user interfaces and some personal application logic.
3) More security as the important software is on the server in a more controlled environment.

Disadvantages:

1) Complexity in design and development. This technology is more difficult to implement than the two-tier client/server architecture.

3.5 Data Access Interface

Universal Data Access is Microsoft's strategy for providing high-performance access to variety of information sources, including relational and non-relational data [15].

Universal Data Access provides three types of interface; there are Open Database Connectivity (ODBC), OLE DB and ActiveX Data Objects (ADO).

3.5.1 Open Database Connectivity (ODBC)

The open database connectivity (ODBC) standard is an interface by which application program can access and process SQL databases in a DBMS-independent manner [4]. This means that an application that been created can access databases supported by different DBMS products as long as the database is ODBC-compliant without any program changes.

3.5.2 OLE DB

OLE DB is an object-oriented interface that is suited to object oriented languages like C++ and Java. OLE DB is an implementation of the Microsoft OLE object standard. Therefore, OLE DB objects are COM objects that support all required interfaces for such objects.

OLE DB is designed to provide universal access to several relational and non-relational data sources. OLE DB breaks the features and functions of a Database Management System (DBMS) up into objects. There can be objects that support query operations, perform updates, support the creation of database schema and perform transaction management [4].

3.5.3 ActiveX Data Objects (ADO)

Since OLE DB is an object-oriented interface, it is not as accessible to Visual Basic and scripting developers. Therefore, ADO is defined as a cover over OLE DB objects. ADO enables programmers to access the functionality of OLE DB in almost any languages and it is closely integrated with Remote Date Services (RDS). See figure

This means that by using ADO in conjunction with OLE DB, communication to Access, Oracle, and SQL Server can be done by simply using ADO object model. Therefore, ADO provides a consistent interface that can access any source of data in all environments.

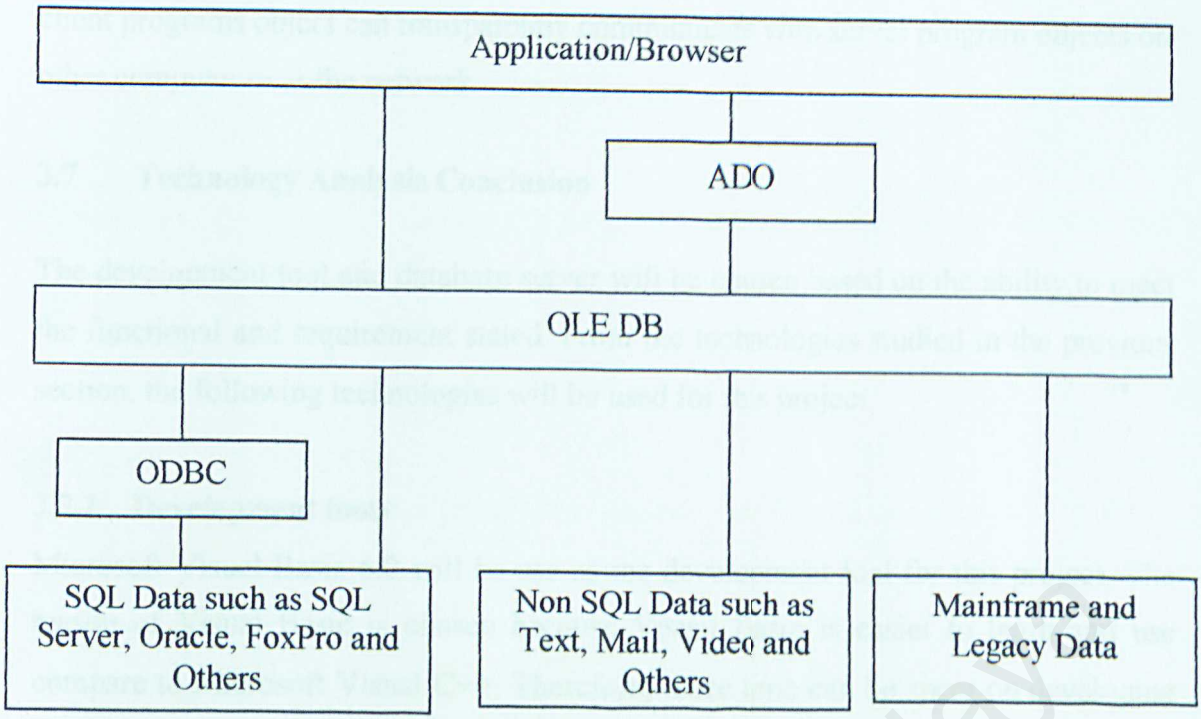


Figure 3.2 Role of ADO, OLE DB and ODBC

3.6 COM (Component Object Model)/DCOM (Distributed COM)

The Component Object Model (COM) is Microsoft's standard for object interaction. The COM specification defines a set of interfaces. Any COM object can communicate and work with other COM object regardless of the language in which the components were created [23].

Microsoft defined two notable specifications based on the COM specification. They are OLE (Object Linking and Embedding) and ActiveX objects. OLE and ActiveX use COM to facilitate all the interaction between objects. OLE uses COM to communicate between applications, allowing users to link and access the functionality of other applications in the Windows environment.

ActiveX uses COM to support communication between controls. An ActiveX Control is an ActiveX object that supports additional interfaces that allow the control's properties and methods to be accessed in many different development environments.

Interfaces COM components can be easily called remotely via Distributed COM (DCOM), which makes it easy to construct distributed applications [11]. Therefore, DCOM extends the communication capabilities across the network. With DCOM,

client programs object can transparently communicate with server program objects on other computer over the network.

3.7 Technology Analysis Conclusion

The development tool and database server will be chosen based on the ability to meet the functional and requirement stated. From the technologies studied in the previous section, the following technologies will be used for this project.

3.7.1 Development tools

Microsoft Visual Basic 6.0 will be use as the development tool for this project. The reason of Visual Basic is chosen because Visual Basic is easier to learn and use compare to Microsoft Visual C++. Therefore, more time can be spent on developing the features and functions of this system.

3.7.2 Database Server

Microsoft SQL Server will be use as the database server for this School Management System. Since the system to be developed will be Windows-based system, Microsoft SQL Server is suitable to use as a database server for this system. Furthermore, SQL Server is easier to use as it has graphical management tool compares to MySQL, which is command-based. SQL Server support bigger database and user-defined functions, which are important for the system future expandability.

CHAPTER 4
SYSTEM DESIGN

This chapter described the process of designing the School Management System. The requirements statements in the system analysis stage will be translated into design specification.

4.1 Database Design

The following section lists the database structure of each table needed in this system.

4.1.1 Data Dictionary

Table: User_T

Description: User Login Table, which keeps user ID and password

Field Name	Data Type	Description
User_ID	Char (30) primary key	User ID to login
Password	Char (50)	User Password
Access_Level	Int (4)	Access_Level
User_Role	Varchar(50)	User Role

Table 4.1 User_T

Table: Student_T

Description: Student Information Table, which keeps student personal information

Field Name	Data Type	Description
Student_ID	Char (15) primary key	Student ID
Name	Varchar (50)	Student Name
Ic_No	Char (20)	Identity card number
Gender	Char (20)	Gender
Date_Birth	DateTime (8)	Date of Birth
Street1	Varchar (50)	Street1 of the address
Street2	Varchar (50)	Street2 of the address
Street3	Varchar (50)	Street3 of the address
City	Varchar (30)	City
Postcode	Char (5)	Postcode

Place_Birth	Varchar (50)	Place of Birth
Telephone	Varchar (15)	Telephone number
Race	Char (20)	Race
Religion	Char (20)	Religion
Birth_No	Char (20)	Birth Certificate number
Status	Char (1)	To determine current students/ former students
Date_Enroll	DateTime (8)	Date of enrollment
Date_Leave	DateTime (8)	Date of leaving school
Last_update	DateTime (8)	Date and time of last update
Guardian_IcNo	Varchar (20) foreign Key	Guardian Identity card number
Guardian_Name	Varchar(50)	Guardian Name

Table 4.2 Student_T

Table: Parent_T

Description: Parent Information Table, which include student's father and mother information

Field Name	Data Type	Description
Parent_IcNo	Char (15) primary key	Identity Card Number
Parent_Name	Varchar (50)	Parent name
Occupation	Varchar (50)	Occupation
Race	Char (20)	Race
Religion	Char (20)	Religion
Office_Address	Varchar (80)	Office address
Tel_House	Varchar (15)	House telephone number
Tel_Office	Varchar (15)	Office telephone number
Mobile_Phone	Varchar (15)	Mobile Phone number
Street 1	Varchar (50)	Street1 of house address
Street 2	Varchar (50)	Street2 of house address
Street 3	Varchar (50)	Street3 of house address

City	Varchar (30)	City
Postcode	Char (5)	Postcode

Table 4.3 Parent_T

Table: Student_Parent_Relationship_T

Description: This table keeps the relationship between the student and parents.

Field Name	Data Type	Description
Student_ID	Char (15)	Student ID
Parent_IcNo	Char (15)`	Parent ID number
Relationship	Varchar(50)	Relationship(Father/Mother)

Table 4.4 Student_Parent_Relationship_T

Table: Staff_T

Description: Staffs Information Table, which keeps staffs personal information

Field Name	Data Type	Description
Staff_ID	Char (15) primary key	Staff ID
Name	Varchar (50)	Staff Name
Ic_No	Char (20)	Identity card number
Date_Birth	DateTime (8)	Date of birth
Race	Varchar (20)	Race
Religion	Varchar (20)	Religion
Gender	Char (1)	Gender
Marital_Status	Varchar (20)	Marital Status
School_Position	Varchar (50)	Position in school
Street1	Varchar (50)	Street1 of address
Street2	Varchar (50)	Street2 of address
Street3	Varchar (50)	Street3 of address
City	Varchar (30)	City of address
Postcode	Char (5)	Postcode of address
Telephone	Varchar (15)	Telephone number
Mobile_Phone	Varchar (15)	Mobile phone number
Last_Update	DateTime (8)	Last update date and time
Date_Leave	DateTime (8)	Date of leaving school

Staff_Status	Char (1)	Staff status
Date_Enroll	DateTime (8)	Date of entered school

Table 4.5 Staff_T

Table: Teacher_T

Description: Teachers Information Table, which keeps teacher personal information

Field Name	Data Type	Description
Teacher_ID	Char (15) primary key	Teacher ID
Name	Varchar (50)	Teacher Name
Ic_No	Char (20)	Identity card number
Date_Birth	DateTime (8)	Date of birth
Race	Char (20)	Race
Religion	Char (20)	Religion
Gender	Char (1)	Gender
Marital_status	Char (20)	Marital Status
Category	Char	Category
Position	Varchar (50)	Position in school
Street1	Varchar (50)	Street1 of address
Street2	Varchar (50)	Street2 of address
Street3	Varchar (50)	Street3 of address
City	Varchar (50)	City of address
Postcode	Char (5)	Postcode of address
Tel	Varchar (15)	Telephone number
Mobile_phone	Varchar (15)	Mobile phone number
Status	Char (1)	To determine current teacher or former teacher
Date_Teach	DateTime (8)	Date start teaching
Date_Leave	DateTime (8)	Date leave school
Last_Update	DateTime (8)	Last update date and time

Table 4.6 Teacher_T

Table: Guardian_T

Description: Student Guardian Information Table

Field Name	Data Type	Description
Guardian_IcNo	Varchar (20) primary key	Identity Card number
Guardian_Name	Varchar (50)	Guardian name
Occupation	Varchar (50)	Occupation
Office_Address	Varchar (80)	Office address
Tel_House	Varchar (15)	House telephone number
Tel_Office	Varchar (15)	Office telephone number
Street 1	Varchar (50)	Street1 of house address
Street 2	Varchar (50)	Street2 of house address
Street 3	Varchar (50)	Street3 of house address
City	Varchar (30)	City
Postcode	Char (5)	Postcode

Table 4.7 Guardian_T

Table: Contact_T

Description: Student Emergency Contact Information Table

Field Name	Data Type	Description
Contact_ID	Int (4) primary key	Contact ID number
Name	Varchar (50)	Contact person name
Tel	Varchar (15)	Telephone number
Tel_1	Varchar (15)	Telephone number
Street 1	Varchar (50)	Street1 of house address
Street 2	Varchar (50)	Street2 of house address
Street 3	Varchar (50)	Street3 of house address
City	Varchar (30)	City
Postcode	Char (5)	Postcode
Relation	Varchar (50)	Relationship with student
Student_ID	Int (4) foreign key	Student ID

Table 4.8 Contact_T

Table: Teacher_Subject_Relationship_T

Description: This table keeps the subjects teaches by each teacher according to year

Field	Data Type	Description
Teacher_ID	Char (15)	Teacher ID
Subject_ID	Int (4)	Subject ID
Year	Int (4)	Academic Year

Table 4.9 Teacher_Subject_Relationship_T

Table: Student_Subject_Relationship_T

Description: This table keeps the subject taken by each student according to academic year.

Field	Data Type	Description
Student_ID	Char (15)	Student ID
Subject_ID	Int (4)	Subject ID
Year	Int (4)	Academic year

Table 4.10 Student_Subject_Relationship_T

Table: Subject_T

Description: Subject Table

Field Name	Data Type	Description
Subject_ID	Int (4) primary key	Subject ID
Subject_Name	Varchar (80)	Subject name
Form	Int (4) foreign key	Form

Table 4.11 Subject_T

Table: Class_T

Description: Class Table

Field Name	Data Type	Description
Class_ID	Int (4) primary key	Class ID
Class_Name	Varchar (50)	Class name
Form	Int (4) foreign key	Form

Table 4.12 Class_T

Table: Student_Class_Relationship_T

Description: This table keeps what class that the student is in according to academic year

Field Name	Data Type	Description
Year	Int (4)	Academic year
Class_ID	Int (4) Primary Key	Class ID
Student_ID	Char (15) Primary Key	Student ID

Table 4.13 Student_Class_Relationship_T

Table: Teacher_Class_Relationship_T

Description: This table keeps who is the form teacher for each class according to year

Field Name	Data Type	Description
Year	Int (4)	Academic year
Class_ID	Int (4)	Class ID
Student_ID	Char (15)	Student ID

Table 4.14 Teacher_Class_Relationship_T

Table: Activity_T

Description: Extra Co-curriculum Activities Table

Field Name	Data Type	Description
Activity_ID	Int (4) primary key	Activity ID
Activity_Name	Varchar (80)	Name of the activity
Activity_Type	Int (4) foreign key	Type of activity

Table 4.15 Activity_T

Table: ActivityType_T

Description: Types of extra-curriculum Activities Table

Field Name	Data Type	Description
Activity_Type	Int (4) primary key	Type of Activity
Description	Varchar (80)	Description of type of activity

Table 4.16: ActivityType_T

Table: Teacher_Activity_Relationship_T

Description: This table keeps activity advice by each teacher each year

Field Name	Data Type	Description
Teacher_ID	Char (15)	Teacher ID (teacher advisor)
Activity_ID	Int (4)	Activity ID
Year	Int (4)	Academic year

Table 4.17: Teacher_Activity_Relationship_T

Table: Student_Activity_Relationship_T

Description: This table keeps activities join by each students each year.

Field Name	Data Type	Description
Student_ID	Char (15)	Student ID
Activity_ID	Int (4)	Activity ID
Year	Int (4)	Year

Table 4.18 Student_Activity_Relationship_T

Table: Test_T

Description: This table keeps type of test or exam

Field Name	Data Type	Description
Test_ID	Char (15) primary key	Test ID
Test_Type	Varchar (80)	Type of Test or Exam

Table 4.19 Test_T

Table: StudentResult_T

Description: This table keeps the result of student for each subjects based on a test or exam.

Field Name	Data Type	Description
Student_ID	Char (15)	Student ID
Test ID	Char (15)	Test ID

Year	Int (4)	Academic year
Subject_ID	Int (4)	Subject_ ID
Total_Mark	Real (4)	Total Marks for the subject (Test1+Test2+Assignment+Exam_Mark)
Test1	Real (4)	Marks for test1
Test2	Real (4)	Marks for test2
Assignment	Real (4)	Marks for assignment
Grade	Char (5)	Grade for the subject
Exam_Mark	Real (4)	Marks for exam
Note	Char (5)	Indicate if student absent or drop the subject ("TH" Absent, "G' Drop)
Pass	Char (1)	Indicate student Fail or Pass (F for Fail, P for Pass)

Table 4.20 StudentResult_T

Table: Medical_T

Description: Student Medical Record Table

Field Name	Data Type	Description
Student_ID	Char (15) foreign key	Student ID
Date	DateTime (8)	Date
Year	Int (4)	Academic Year
Hearing	Bit	True if require attention/ False if not
Eyesight	Bit	True if require attention/ False if not
Speaking	Bit	True if require attention/ False if not
Disable	Bit	True if require attention/ False if not
Serious_Sickness	Text	Indicate if there is any serious sickness or illness
Height	Real (4)	Height in meter
Weight	Real (4)	Weight in kilogram

Table 4.21 Medical_T

Table: Attendance_T

Description: This table keeps student attendance according to academic year

Field Name	Data Type	Description
Year	Int (4) Primary key	Academic year
Student_ID	Char (15) Primary key	Student ID
Day_Attend	Int (4)	Number of days student attend school
Day_Miss	Int (4)	Number of days student absent from school

Table 4.22 Attendance_T

Table: Absent_T

Description: This table keeps student absent date and reason.

Field Name	Data Type	Description
Year	Int (4) Primary key	Academic year
Student_ID	Char (15) Primary key	Student ID
Date_Absent	DateTime (8)	Date absent from school
Reason	Bit (1)	True - With Reason, False- without reason

Table 4.23 Absent_T

Table: Grade_T

Description: This table keeps the grade set for the range of exam marks for each form.

Field Name	Data Type	Description
ID	Int (4) Primary Key	ID
Form	Char (5)	Form
Lower_Mark	Int (4)	Lower range of the marks
Upper_Mark	Int (4)	Upper range of the marks
Grade	Char (2)	Grade for the range of the marks
Description	Varchar (50)	Description on grade

Table 4.24 Grade_T

Table: Form_T

Description: This table keeps the form (level) and description

Field Name	Data Type	Description
Form	Int (4) Primary Key	Form
Description	Varchar (50)	Description

Table 4.25 Form_T

Table: Teacher_SubjectClass_Relationship_T

Description: To keeps information on which class the teacher teach for a particular subject.

Field Name	Data Type	Description
Teacher_ID	Char (15)	Teacher ID
Subject_ID	Int (4)	Subject ID
Class_ID	Int (4)	Class ID
Year	Int (4)	Academic Year

Table 4.26 Teacher SubjectClass_Relationship_T

Table: tmp_Student

Description: This table is a temporary table that keeps student information that been archive.

Field Name	Data Type	Description
Student_ID	Int (4) primary key	Student ID number
Name	Varchar (50)	Student Name
Ic_No	Char (20)	Identity card number
Gender	Char (20)	Gender
Date_Birth	DateTime (8)	Date of Birth
Street1	Varchar (50)	Street1 of the address
Street2	Varchar (50)	Street2 of the address
Street3	Varchar (50)	Street3 of the address
City	Varchar (30)	City
Postcode	Char (5)	Postcode

Place_Birth	Varchar (50)	Place of Birth
Telephone	Varchar (15)	Telephone number
Race	Char (20)	Race
Religion	Char (20)	Religion
Birth_No	Char (20)	Birth Certificate number
Status	Char (1)	To determine current students/ former students
Date_Enroll	DateTime (8)	Date of enrollment
Date_Leave	DateTime (8)	Date of leaving school
Last_update	DateTime (8)	Date and time of last update
Guardian_IcNo	Varchar (20) foreign Key	Guardian Identity card number
Guardian_Name	Varchar(50)	Guardian Name

Table 4.27 tmp_student

Table: tmp_Parent

Description: This table is a temporary table that keeps student parent information that been archive.

Field Name	Data Type	Description
Parent_IcNo	Char(15) primary key	Identity Card Number
Parent_Name	Varchar (50)	Parent name
Occupation	Varchar (50)	Occupation
Race	Char (20)	Race
Religion	Char (20)	Religion
Office_Address	Varchar (80)	Office address
Tel_House	Varchar (15)	House telephone number
Tel_Office	Varchar (15)	Office telephone number
Mobile_Phone	Varchar (15)	Mobile Phone number
Street 1	Varchar (50)	Street1 of house address
Street 2	Varchar (50)	Street2 of house address
Street 3	Varchar (50)	Street3 of house address
City	Varchar (30)	City

Postcode	Char (5)	Postcode
Student_ID	Char (15)	Student ID
Date_Leave	DateTime (8)	Date of Student Leave School

Table 4.28 tmp_Parent

Table: tmp_Medical

Description: This table is a temporary table that keeps Student Medical information that has been archive.

Field Name	Data Type	Description
Student_ID	Int (4) foreign key	Student ID number
Date	DateTime (8)	Date
Year	Int (4)	Academic Year
Hearing	Bit	True if require attention/ False if not
Eyesight	Bit	True if require attention/ False if not
Speaking	Bit	True if require attention/ False if not
Disable	Bit	True if require attention/ False if not
Serious_Sickness	Text	Indicate if there is any serious sickness or illness
Height	Real (4)	Height in meter
Weight	Real (4)	Weight in kilogram
Date_Leave	DateTime (8)	Date of student leave school

Table 4.29 tmp_Medical

Table: tmp_Guardian

Description: This table is a temporary table that keeps Student Guardian information that has been archive.

Field Name	Data Type	Description
Guardian_IcNo	Varchar (20) primary key	Identity Card number
Guardian_Name	Varchar (50)	Guardian name
Occupation	Varchar (50)	Occupation
Office_Address	Varchar (80)	Office address
Tel_House	Varchar (15)	House telephone number
Tel_Office	Varchar (15)	Office telephone number
Street 1	Varchar (50)	Street1 of house address
Street 2	Varchar (50)	Street2 of house address
Street 3	Varchar (50)	Street3 of house address
City	Varchar (30)	City
Postcode	Char (5)	Postcode
Student_ID	Char (15)	Student ID
Date_Leave	DateTime (8)	Date of student Leave School

Table 4.30 tmp_Guardian

Table: tmp_Teacher

Description: This is temporary table that keeps the teachers information that have been archived.

Field Name	Data Type	Description
Teacher_ID	Char (15) primary key	Teacher ID
Name	Varchar (50)	Teacher Name
Ic_No	Char (20)	Identity card number
Date_Birth	DateTime (8)	Date of birth
Race	Char (20)	Race
Religion	Char (20)	Religion
Gender	Char (1)	Gender
Marital_status	Char (20)	Marital Status
Category	Char	Category

Position	Varchar (50)	Position in school
Street1	Varchar (50)	Street1 of address
Street2	Varchar (50)	Street2 of address
Street3	Varchar (50)	Street3 of address
City	Varchar (50)	City of address
Postcode	Char (5)	Postcode of address
Tel	Varchar (15)	Telephone number
Mobile_phone	Varchar (15)	Mobile phone number
Status	Char (1)	To determine current teacher or former teacher
Date_Teach	DateTime (8)	Date start teaching
Date_Leave	DateTime (8)	Date leave school
Last_Update	DateTime (8)	Last update date and time

Table 4.31 tmp_Teacher

Table: tmp_Staff

Description: This is a temporary table that keeps staffs information, which has been archived.

Field Name	Data Type	Description
Staff_ID	Char (15) primary key	Staff ID
Name	Varchar (50)	Staff Name
Ic_No	Char (20)	Identity card number
Date_Birth	DateTime (8)	Date of birth
Race	Varchar (20)	Race
Religion	Varchar (20)	Religion
Gender	Char (1)	Gender
Marital_Status	Varchar (20)	Marital Status
School_Position	Varchar (50)	Position in school
Street1	Varchar (50)	Street1 of address
Street2	Varchar (50)	Street2 of address
Street3	Varchar (50)	Street3 of address
City	Varchar (30)	City of address
Postcode	Char (5)	Postcode of address

Telephone	Varchar (15)	Telephone number
Mobile_Phone	Varchar (15)	Mobile phone number
Last_Update	DateTime (8)	Last update date and time
Date_Leave	DateTime (8)	Date of leaving school
Staff_Status	Char (1)	Staff status
Date_Enroll	DateTime (8)	Date of entered school

Table 4.32 tmp_Staff

4.1.2 Entity-Relationship Diagram

Figure below shows the Entity-Relationship Diagram of the school Management System.

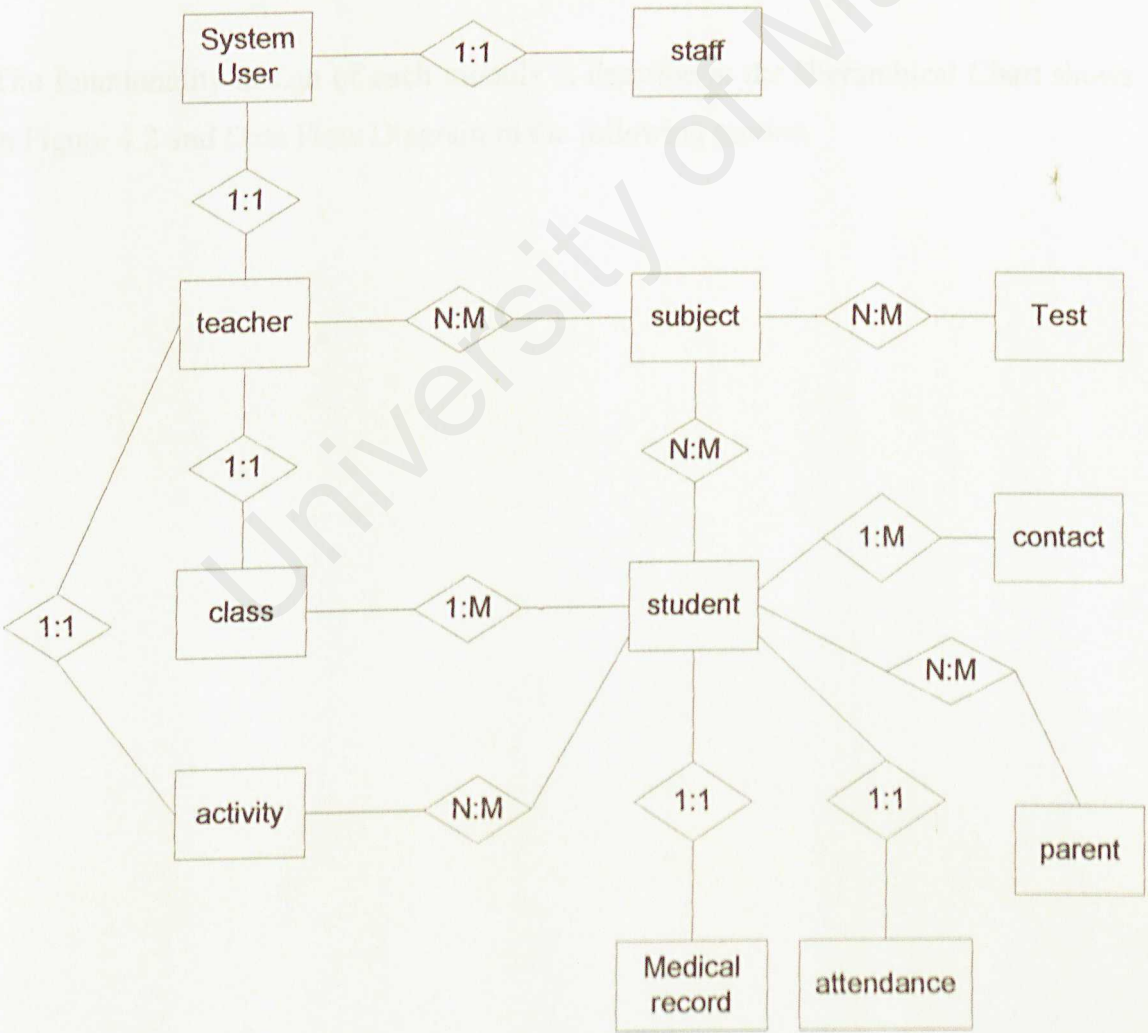


Figure 4.1 Entity-Relationship Diagram of School Management System

4.2 Functionality Design

4.2.1 Hierarchical chart

The School Management System generally has the following modules, which are:

1) Authorisation and Authentication Function
2) Staff Information System
3) Teacher Information System
4) Student Information System
5) Class Management
6) Subject Module
7) Analysis of Result
8) Old Records maintenance
9) System Administration

The functionality design of each module is describe in the Hierarchical Chart shows in Figure 4.2 and Data Flow Diagram in the following section.

Figure 4.2 Hierarchical Chart for School Management System

4.2.2 Data Flow Diagram (DFD)

4.2.2.1 DFD of Authorisation and Authentication Function

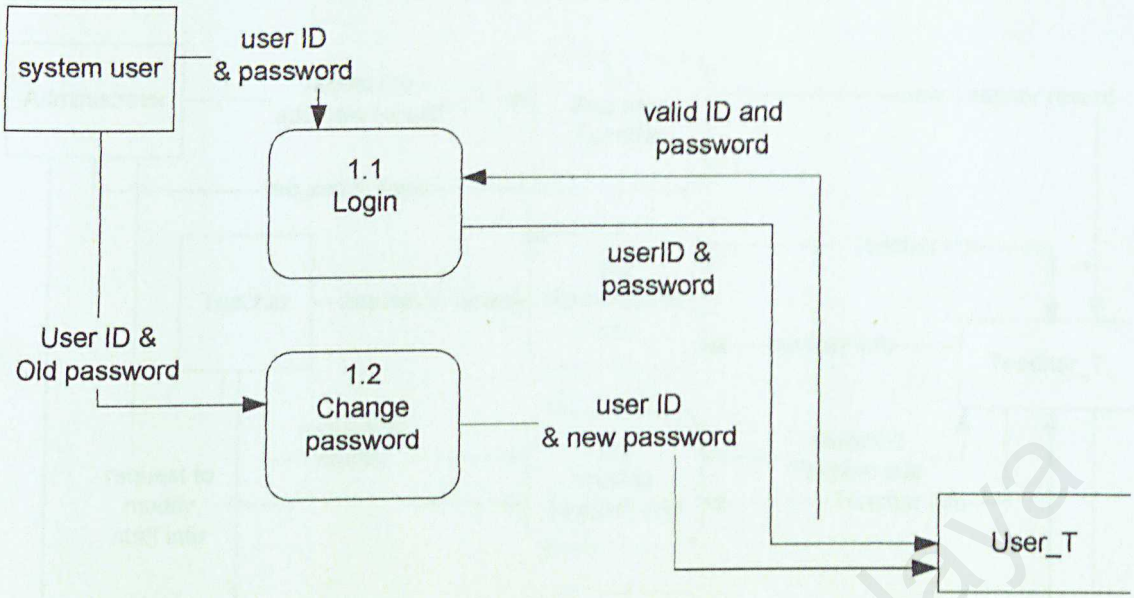


Figure 4.3 Data Flow Diagram for Authorisation and Authentication Function

4.2.2.2 DFD of Staff Information System

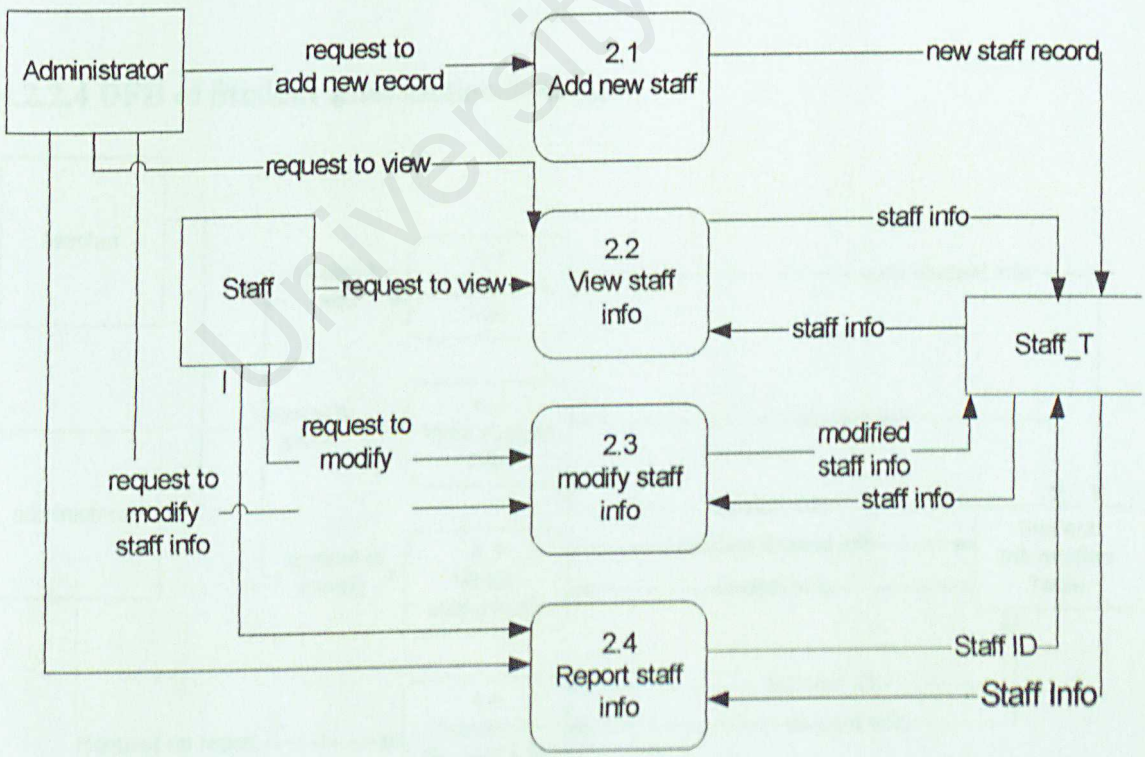


Figure 4.4 Data Flow Diagram for Staff Information System

4.2.2.3 DFD of Teacher Information System

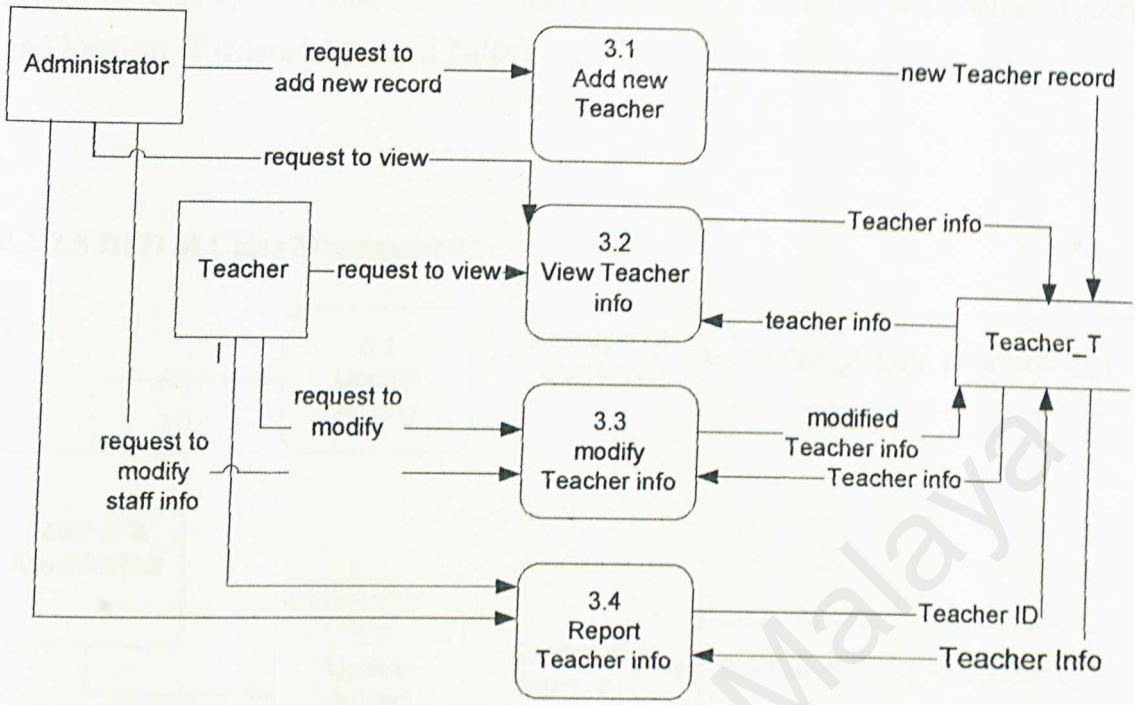


Figure 4.5 Data Flow Diagram for Teacher Information System

4.2.2.4 DFD of Student Information System

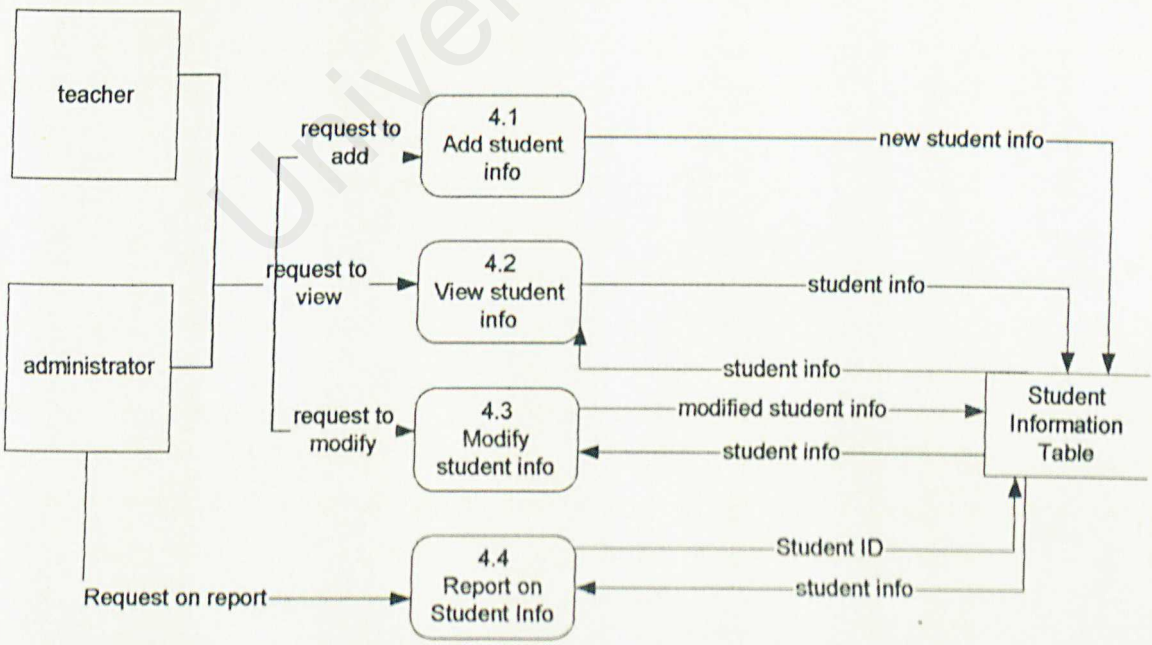
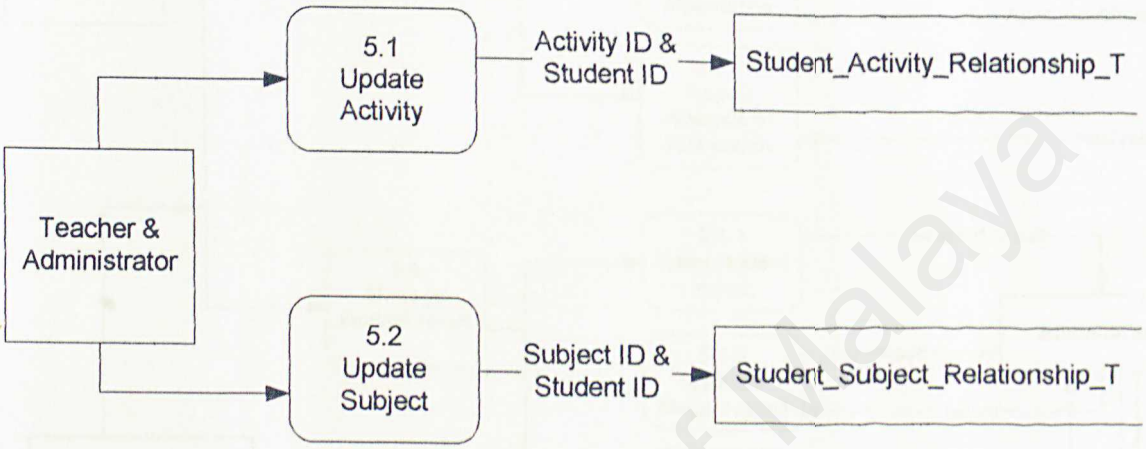


Figure4.6 Data Flow Diagram for Student Information System

In this Data Flow Diagram, Student Information Table is use to represent the tables Student_T (Student Information Table), Parent_T (Parents Information Table), Guardian_T (Guardian Information Table), Contact_T (Emergency Contact Table) and Medical_T (Medical Record Table).

4.2.2.5 DFD of Class Management



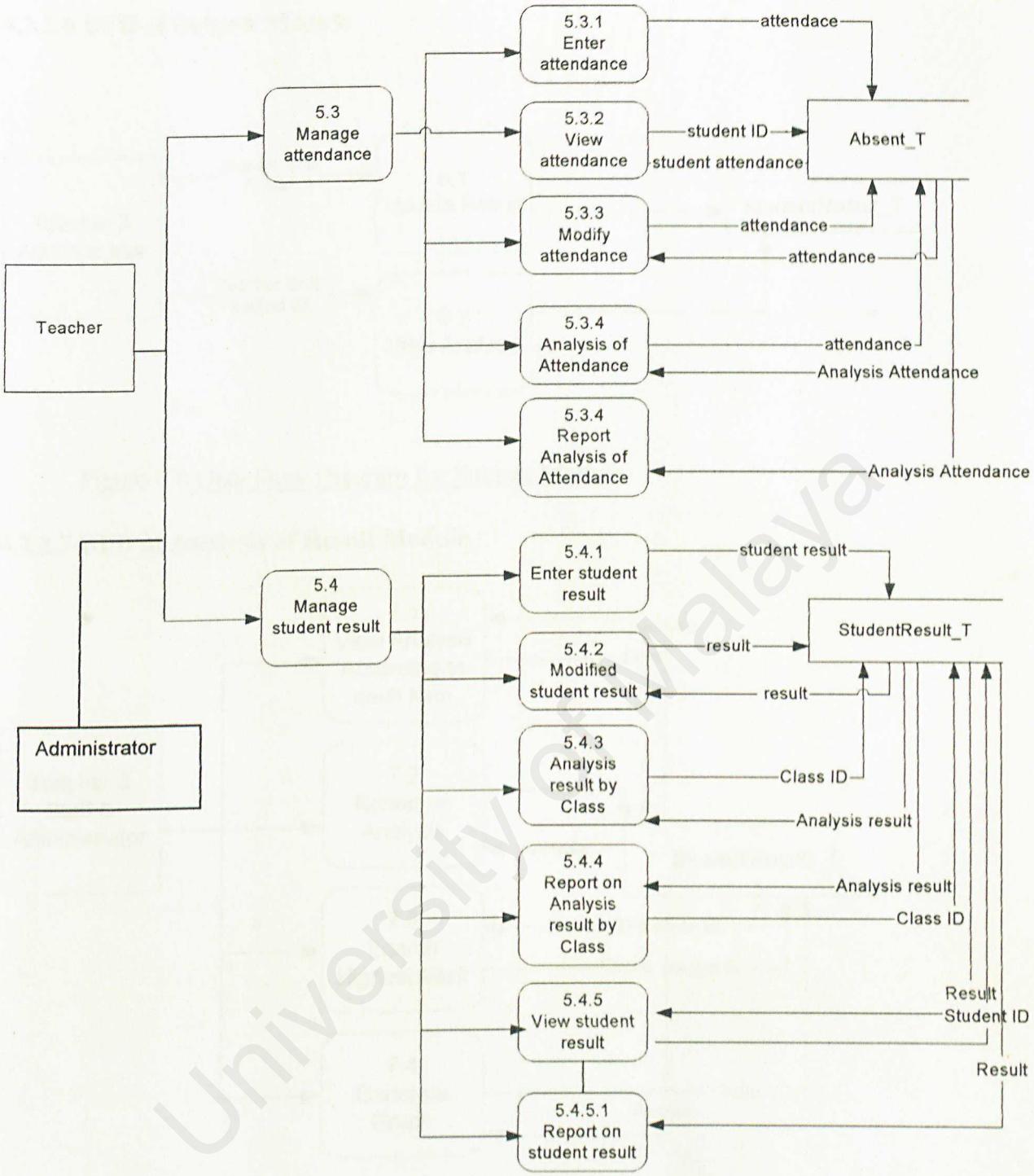


Figure 4.7 Data Flow Diagram for Class Management

4.2.2.6 DFD of Subject Module

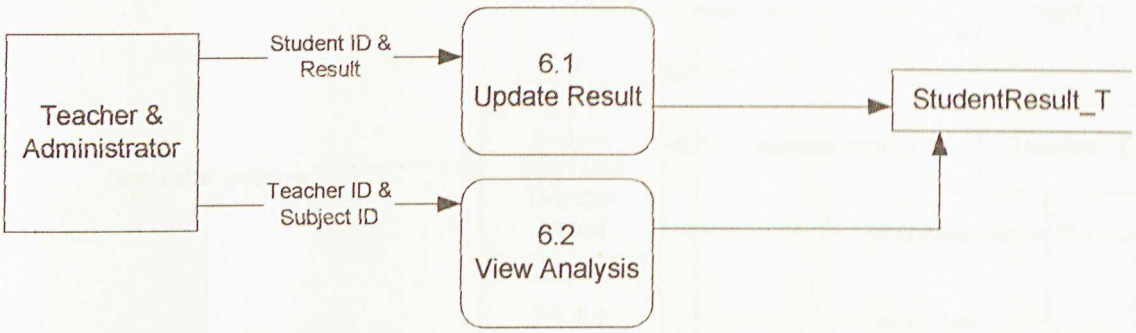


Figure 4.8 Data Flow Diagram for Subject Module

4.2.2.7 DFD of Analysis of Result Module

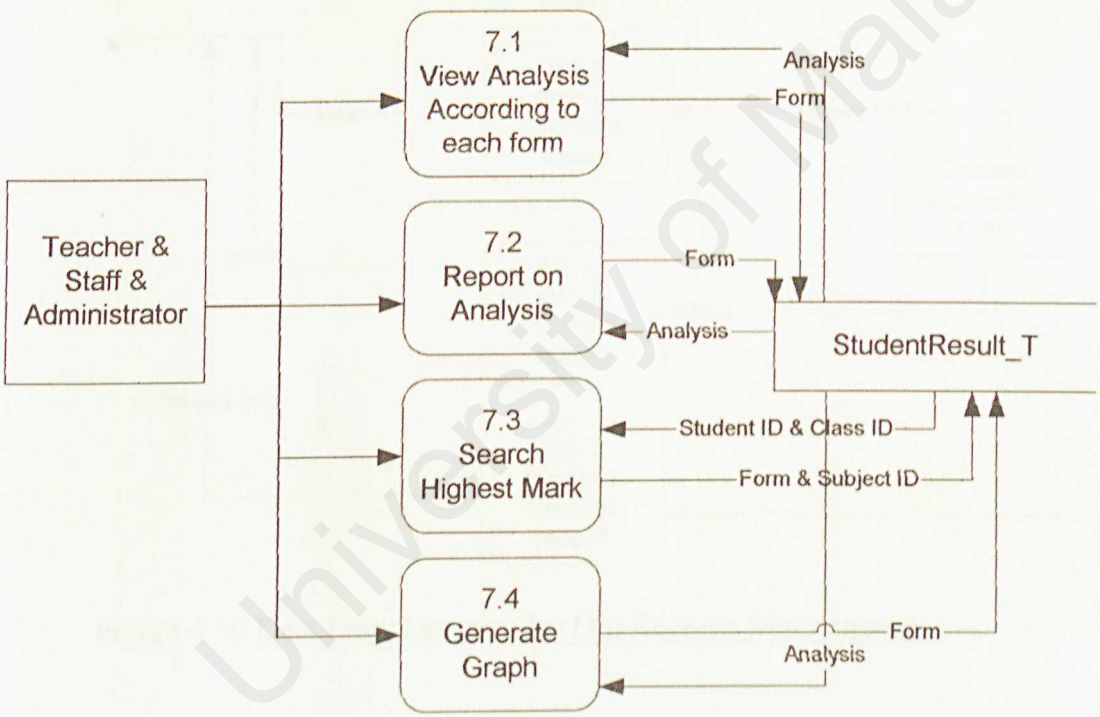


Figure 4.9 Data Flow Diagram for Analysis of Result Module

4.2.2.8 DFD of Old Records Maintenance

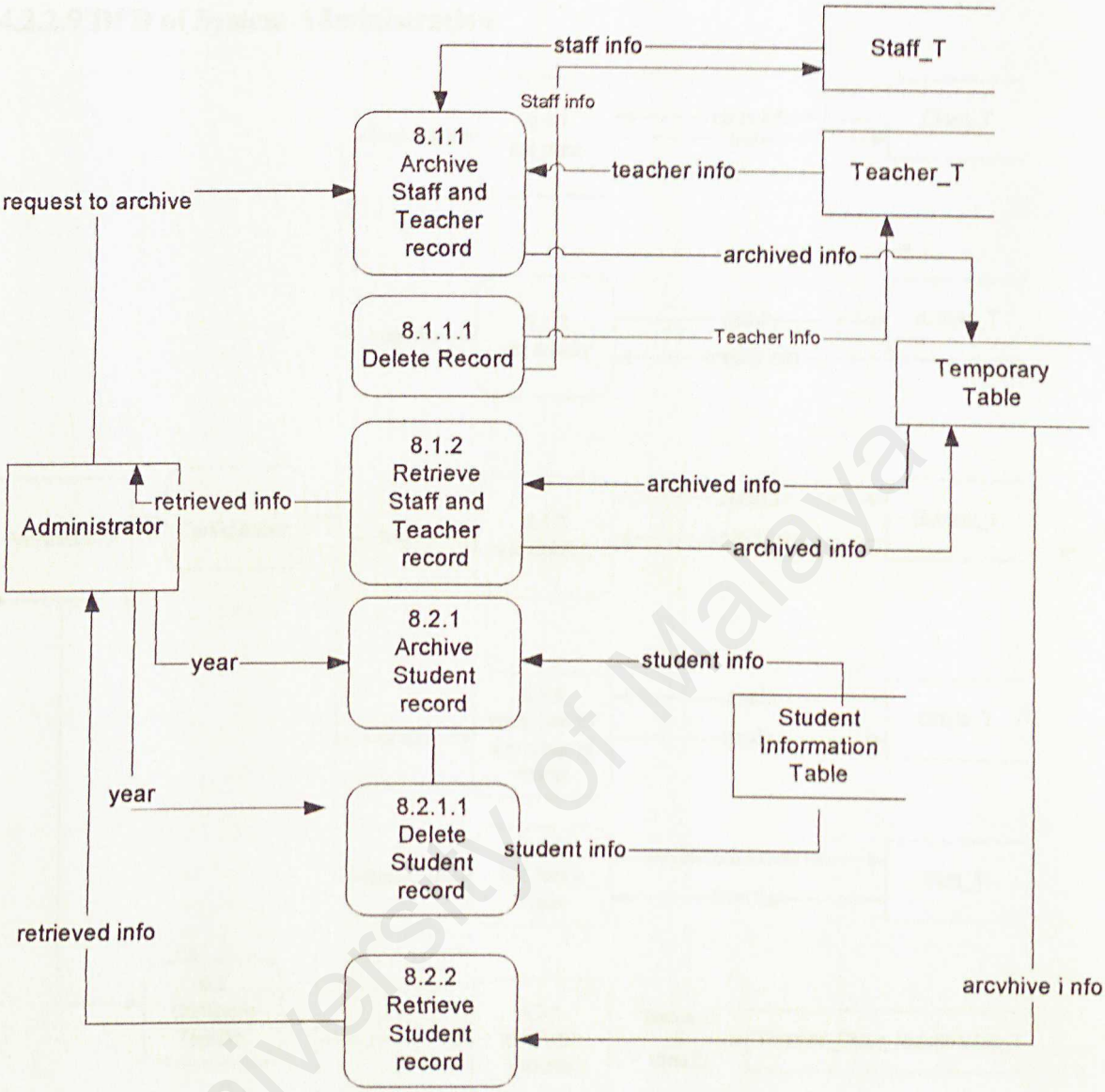
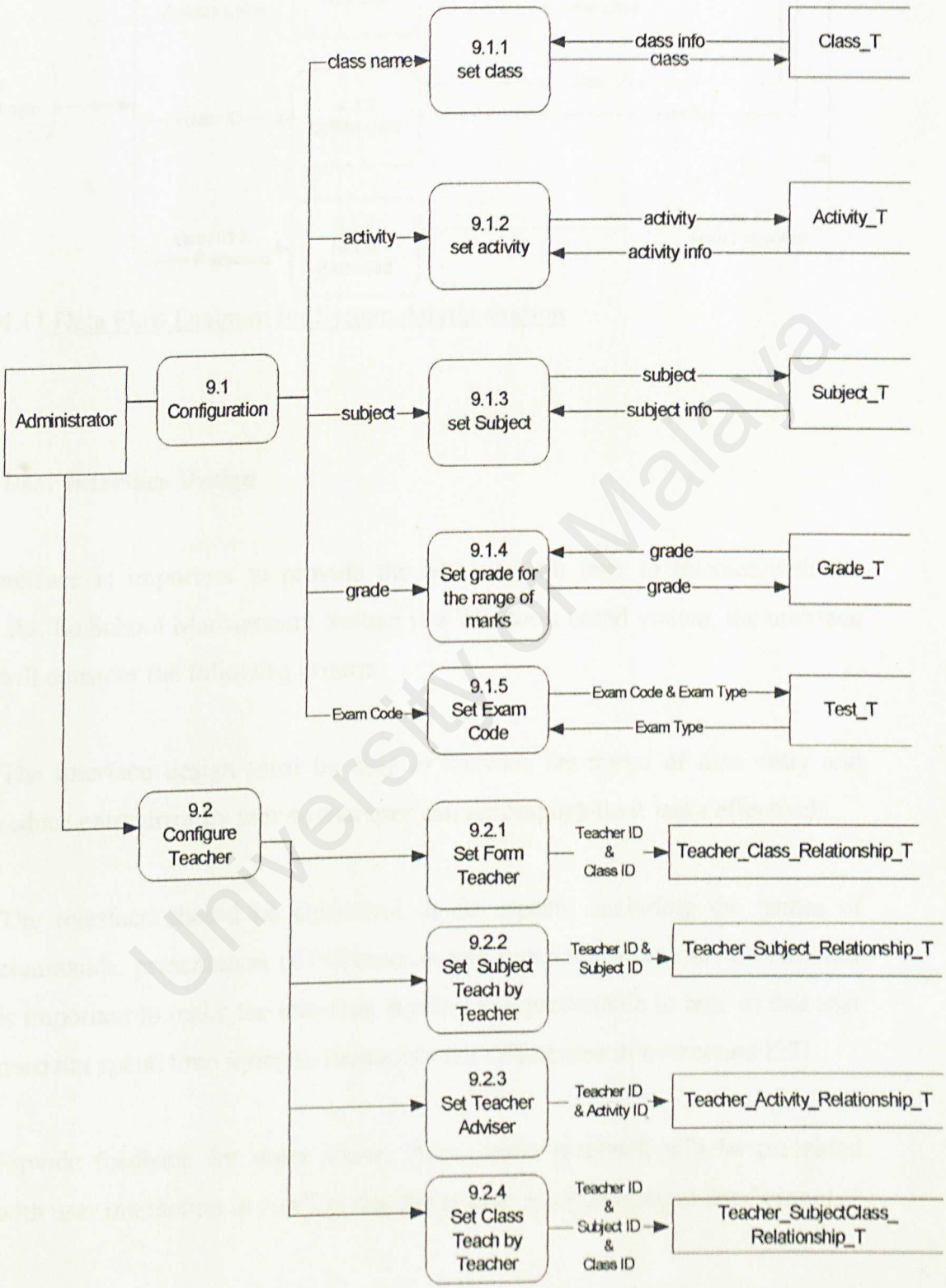


Figure 4.10 Data Flow Diagram for Old Records Maintenance

4.2.2.9 DFD of System Administration



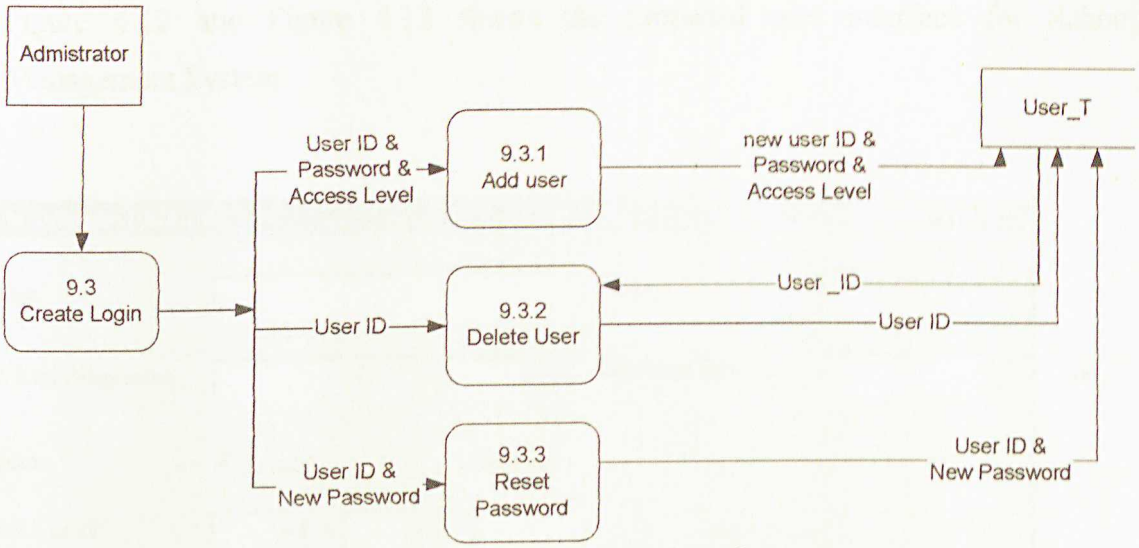


Figure 4.11 Data Flow Diagram for System Administration

4.3 User Interface Design

User Interface is important to provide the best way for user to interact with the system. As the School Management System is a Windows based system, the interface design will consider the following criteria:

1) The interface design must be able to increase the speed of data entry and reduce entry error by user so that user can accomplish their tasks effectively.

2) The interface should be consistent in all aspects, including the names of commands, presentation of information and functional behavior. Consistency is important to make the interface familiar and predictable to user so that user need not spend time trying to remember the differences in interaction [23].

3) Provide feedback for users action. Appropriate feedback will be presented with user interaction to confirm that the system is responding to user's input.

4) The system should be able to accommodate potential user error and make user action reversible and recoverable.

Figure 4.12 and Figure 4.13 shows the proposed user interface for School Management System.

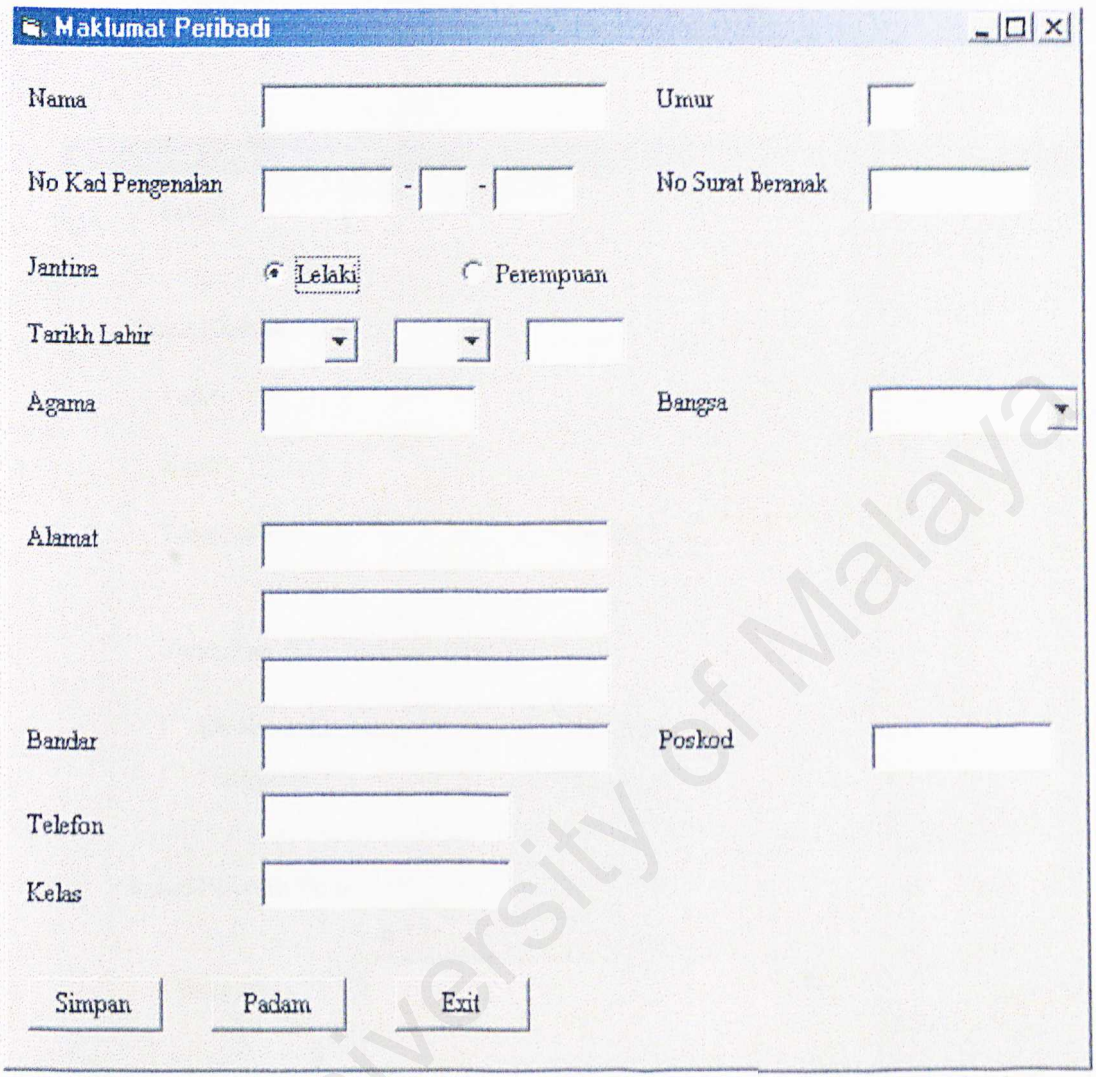


Figure 4.12 Propose Screen for teacher to enter student personal particulars

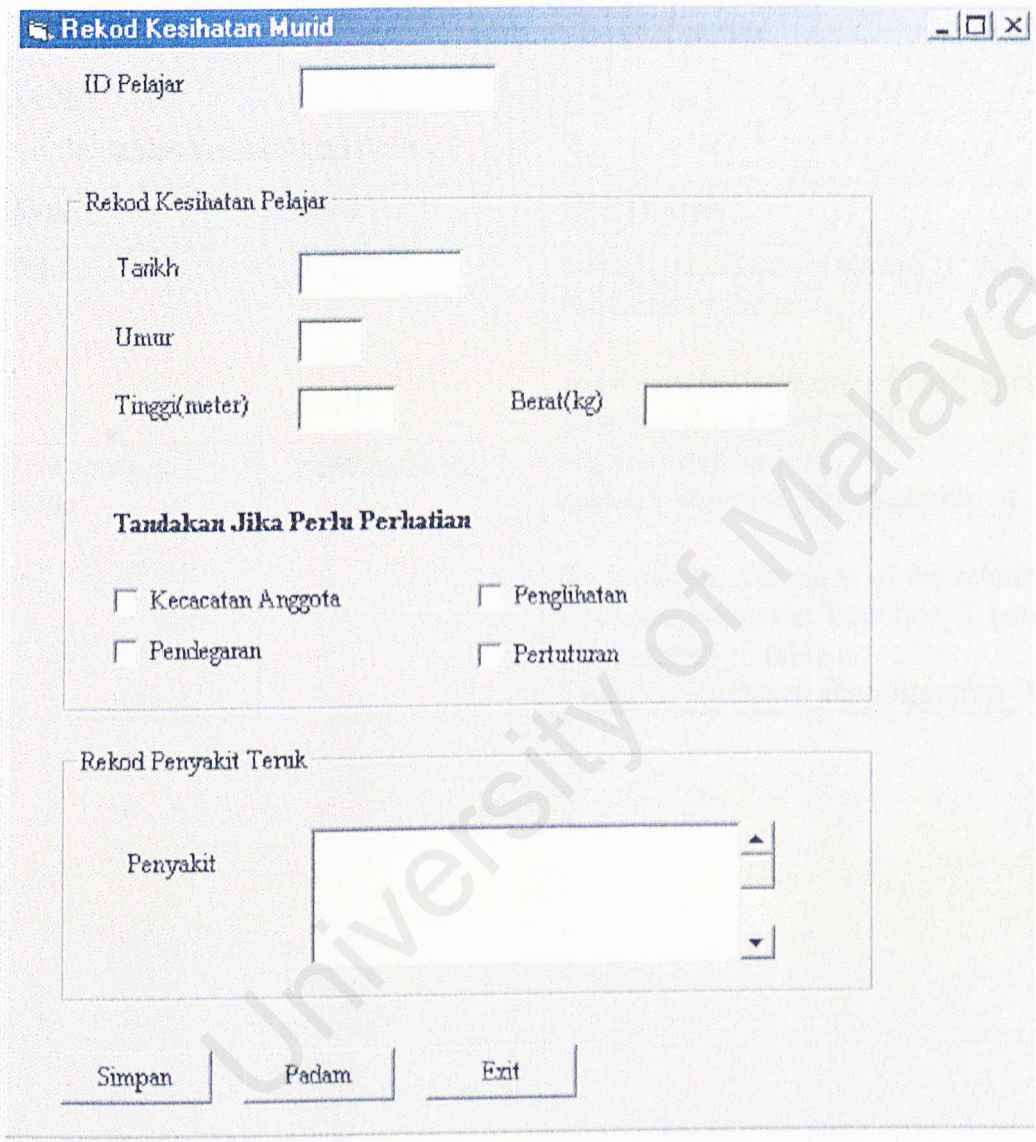


Figure 4.13 Propose Screen for teacher to enter student medical record

4.4 Naming Conventions

Naming standards are used to help to control the look of the system as well as the maintenance the system. The following are the naming standards will be used when developing the system using the objects in the SQL server.

Tables

All the tables are suffixed with _T.

Type	Suffix	Description
Table	_T	Naming standards is Name_T. Name is the name of the table. For example, the name of the Teacher table name is Teacher_T
Relationship Table	_Relationship_T	Naming standards is Name1_Name2_Relationship_T. For example, the name of the relationship table that relate the Teacher_T table and the Subject_T table is Teacher_Subject_Relationship_T.

Indexes

Naming Standards	Description
PK_TableName	SQL server will automatically place an index on the primary key column using this name whenever a table with a primary key is created. TableName is name of the table. For example, an index created by the SQL server on the Teacher_T table is PK_Teacher_T.
IX_TableName	Additional index created will be prefixed with IX_ followed by the table name. For example, additional index created on the Teacher_T table is IX_Teacher_T

Stored Procedures

All the procedures created are will be prefixed with sp_

Prefix	Description
sp_parmins_name	This procedure accepts parameters and insert data.
sp_parmselect_name	This procedure accepts parameters and select data.
sp_parmdel_name	This procedure accepts parameters and deletes data.
sp_parmupd_name	This procedure accepts parameters and update data.
sp_select_name	This procedure only selects data and does not accepts parameters.
sp_Delete_name	This procedure only delete data and does not accepts parameters.

CHAPTER 5
SYSTEM IMPLEMENTATION

5.1 System Implementation

This chapter will discuss about the environment, tools and approach used to implementing School Management System.

5.2 Development Environment

The development environment is the system condition that is use to develop the School Management System.

Hardware

- 450MHz AMD K62 Processor
- 256MB RAM
- 10 GB Hard Disk

Software

Operating System

- Windows 2000 Professional

Database Server

- Microsoft SQL Server 2000

System Development

- Microsoft Visual Basic 6.0

5.3 Coding

5.3.1 Programming Language

School Management System front-end application is developed using Visual Basic. Visual Basic uses the BASIC programming language and has an Integrated Development Environment (IDE) where the system is develop, run, test and debug.

5.3.2 Database

The database in School Management System is created using SQL server database. There are also several store procedures created in the SQL server database. Store procedure which contains T-SQL statements is use to perform a unit of work such as select, update, insert or delete data in the database. These store procedures can be accessed through Visual Basic application. Store procedures are created to allow code reuse as multiple users can execute the same store procedures at the same time and it can help to ease the maintenance. Views are also created to simplify the process of retrieving data

5.3.3 Approach

The approach use to connect the system to the SQL Server database is by using ADO (ActiveX Data Objects) together with an Application Programming Interface (API) OLE DB. For School Management System, a DSN-less connection is use to connect to the database, where a connection to the SQL Server can be establish by specify the parameters required in the connection string without using a DSN (Data Source Name).

Generally, most of the SQL statements to select, insert, update and delete data are created in the store procedures. Some of the store procedure accepts inputs parameters and return parameters. Each of these stored procedures will be execute in first in the SQL Server. Then these store procedure will be accessed through Visual Basic. To provide data access to the store procedures, several class modules was created in Visual Basic.

These class modules contains method that will pass parameter from variables in Visual Basic to the store procedures and execute them, and then return value from store procedure to variable in Visual Basic if the store procedure contains return values. The class modules created are:

- clsInsertStudent and clsUpdateStudent for Student Information System Module.
- ClsChangePassword, clsManageLogin, clsInsertData, clsUpdateData, clsDeleteData and clsDateFuction that are for all the modules in the system.

To generate report in the system, DataReport ActiveX Designer in Visual Basic is use. The reports generated in the system are in pre-defined format and they can be print or export into HTML format.

To handle error in the system, each procedure in the system has its own error handler. Generally, error handler in the system will handle two types of errors; there are ADO Error and VB Error. If errors occur, it first checks whether it is an ADO Error, if it is not an ADO Error, then it will checks for VB Error. After that, the error message will be display in a message box to inform user that errors have occurs. This is to ensure that the system did not end abruptly when errors occur.

The final outcome of interface for the School Management System is different from the proposed interface, as the proposed interface needs to be enhanced.

5.3.4 Documentation

Internal documentation is written in the coding to describe the program data structures, algorithms, and control flow. This internal documentation is important to provide information to someone who might need to read through the source code for modification and maintenance. The internal documentation includes program comments to help readers understand how a particular function is implemented in the code.

CHAPTER 6
SYSTEM TESTING

6.1 Testing Strategies

Testing is important to verify that the School Management System has meet the proposed requirement. It is aims to discover errors and faults in the system and remove them from the system. Unit testing, integration testing and system testing are the testing strategies use for testing the School Management System.

6.1.1 Unit Testing

Unit Testing is done where each program component is tested on its own and is done concurrently during the coding phase. Unit testing make sure that the program component function perform properly with the expected input. In unit testing, white box testing which is based on knowledge of the internal logic of an application's code is use. Tests are based on coverage of code statements, branches, paths, and conditions.

Generally, there are three types of fault that need to be identify in School Management System and need to be eliminate if exist. There are:

- Algorithmic fault, which occurs when a component's algorithm or logic does not produce the proper output for a given output because something is wrong with the processing steps.
- Syntax fault, where the constructs of programming language is not use properly.
- Computation and precision faults, where a formula's implementation is wrong and does not compute the result to the required degree of accuracy. This is because School Management System has to perform analysis of student result.

The major steps in unit testing are:

1) Examining the program code to identify the faults.

2) Testing a component by choosing test cases, where input data and condition is choose to determine that the program component produced the desired output.

6.1.2 Integration Testing

When each program component is unit-tested, integration testing is done to verify that the system components work together. The approach use for integration testing of School Management System is the Bottom-up integration approach. In this approach, each program component is tested individually, and then the next components to be tested are those that call the previously tested one.

6.1.3 System Testing

Once the integration testing is done, system testing is done to ensure that the system has the desired functionality. System testing ignores system structure and focus on whether the system has fulfilled functional requirements and non-functional requirements. In system testing, black box testing, which is not based on any knowledge of internal design or code, is use.

Steps taken in system testing are:

1) Function test, which is based on the functional requirements. Function testing compares the system's actual performance with the requirements.

2) Performance testing, which test the non-functional requirements. The types of performance tests done are:
    a) Human factor tests, which test the requirement dealing with the user interface. Testing is done to determine whether the system is user friendly.
    b) Security test to ensure that the security requirements are met. Testing is done by login as different user roles, and test whether each module has met the security requirements.

6.2 Analysis of Test Result

1) Functions of system according to the requirement

According to the result of function testing, the School Management System has loaded with functions that are according to functional requirements.

Besides that, non-functional requirement such as security is met as the systems provides an authorisation and authenticate function. The system also fulfilled the robustness requirement, as the system is able to handle errors.

2) Some functions require user memorization

According to the human factors test, there are some functions, which required user to remember the class number and teacher ID especially if the user login as the administrator. For example, if the administrator wants to enter the Student Information System ("Sistem Maklumat Pelajar") module, they have to enter the class number. The same problem also arises when the administrator wants to enter the Class Management ("Pegurusan Kelas") module and Subject ("Matapelajaran") module where they have to enter the teacher ID. Therefore, the administrator needs to remember the class number or teacher ID in order to access these modules.

Enhancement can be done by include a search button which shows the class number and class name or the teacher ID and their name to eliminate this problem.

3) Cannot Directly Enter Student Result

The teacher cannot directly enter the student mark directly, instead the teacher had to enter the marks in the Edit Box (Refer to the figure). This is because the MSFlexGrid control that is use in the system lacks the basic capability of letting the user to edit a single cell. Therefore, a TextBox control is place on top of the cell and the user edit a single cell in the textbox.

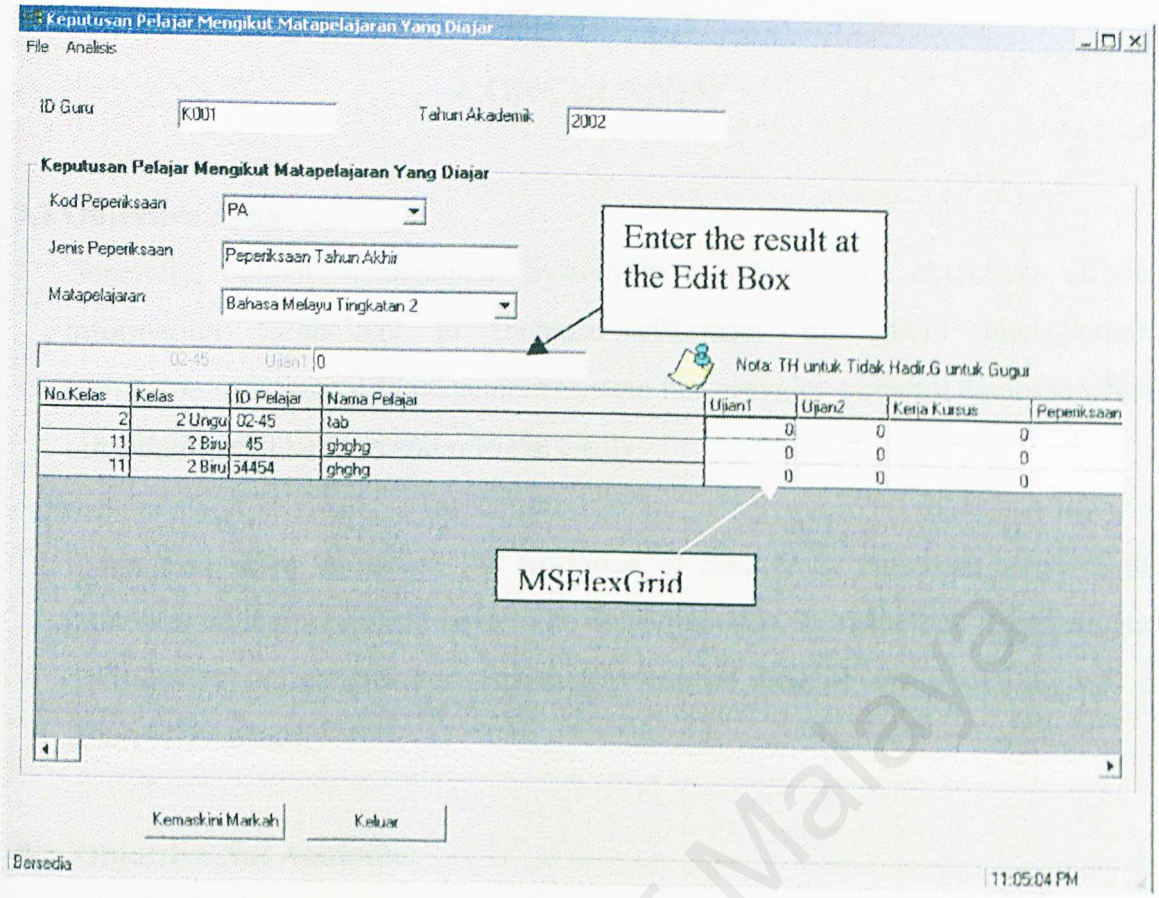


Figure 6.1 Screen of Student Result

CHAPTER 7
CONCLUSION

7.1 Objective Achieved

Generally, School Management System have achieves the objectives utilizes information technology to increase efficiency of school management, administration. School Management System has provides a central database where information can be save and retrieve easily.

It has also helps to reduce the workload of teacher by automate some of the processes which previously have to be done manually. Besides that, it also support decision-making as data and information can be retrieve easily and analysis of result is done easily.

7.2 Objective Not Achieved

The School Management System has not achieved the objective of increasing the efficiency of communication. This is because the system does not provide function that the user can communicate with each other or send notice.

7.3 Application Strength

1) Security and Authentication

Every user will need a login name and password to use the system. The login name and password will be use to authorize the user and to check the access level for the user. This helps to ensure the security of the system and prevents unauthorized access to the system.

2) Provide Analysis of Student Result

The system provides analysis of student result for each subject according to:

- Each form (level).
- Each class
- Subject Teacher

3) Assign Grade Automatically

The system will automatically assign grade to the student result when the marks of the students is entered and saved. This helps to reduce time, as the teacher does not need to enter the grade manually.

4) Old Record Maintenance

Old student, teacher and staff record can be archived. The records that have been archive can be retrieve when needed.

7.4 Application Limitation

1) Platform Dependent

As the School Management System is develop using Microsoft Visual Basic. The system can only be installed on the Windows Operating System.

2) Graph Printing Facility

The system does not have printing facility for the graph. The graph can only be view.

3) Teacher planning

The system does not provide facility that the teacher can include their teaching planning, report on teaching and reminder.

7.5 Future Enhancement

1) User Interface

The system interface should enhance to a more attractive interface, as the current system interface is too plain.

2) Graph printing facility and include more graphs

Graph printing facility should be added. More graphs should be added to the system, as the data presented in graph are easier to read.

3) Include Help function

There is no help function included in the system. Although user manual is available, user feels that if the help function is included in the system, it will be easier for them to refer to it with just a click on the button. Therefore, for future enhancement, helps function should be included in the system.

4) Provide teacher planning

For future enhancement, the system should include teacher planning so that the system, which provide analysis of student result, will be able to help the teacher to plan their lesson more effectively.

5) Include search function

A search button should be included for function that require user to enter ID so that user do not need to remember the ID as the problem already addressed in the analysis of test result section.

6) Enhancement on the Attendance Management Function

Currently, the system only allows teacher to set whether a student is absent with excuse or without excuse, for future enhancement, attendance management should allows teacher to enter reason for why a student is absent. It can also be enhance where a message is generated to inform the teacher if a student had been absent without excuse continuously for a number of days.

7.6 Problems and Solving Method

1) Complexity

As the development of the system involved quite a number of modules, the development of the system sometime become complex. Therefore, the design is review and allows minimal changes once the logic is fixed.

2) Technical Problems

During the development of the system, sometime technical problems are encounter and the solution is gathered from books, online help and online forum.

7.7 Knowledge, Skill and Experience

I have the opportunity to learn Visual Basic and learn to use the Microsoft 's ADO database connection interface for database connection. By using SQL Server as the database server, I have a chance to learn Transact SQL Language and store procedure apart from the administration and configuration of the database server.

Furthermore, I have also learned on various consideration and technology available on building a client server application from the aspect of platform, programming language, and databases.

USER MANUAL

Software and Hardware Requirement

- Intel 486 processor and later
- At least 64Mb RAM
- Microsoft SQL Server 2000 installed

U.1 Getting Started

To install the School Management System, double click on the Setup application file, the installation wizard will guide you on installation of the system.

U.2 Starting School Management System

Click on the School Management System executable file. A login form will appear to enable user to logon to the system.

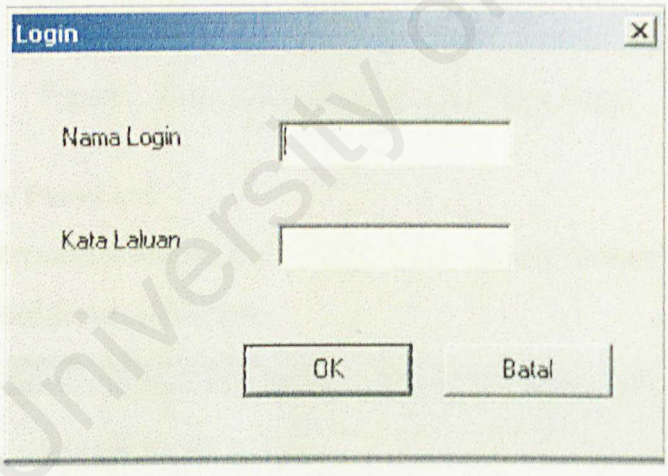


Figure1. User Manual: Screen of Login

Upon logon to the system, the main menu page will be shown. If user login as teacher or staff the administration button on the menu bar will be disabled.

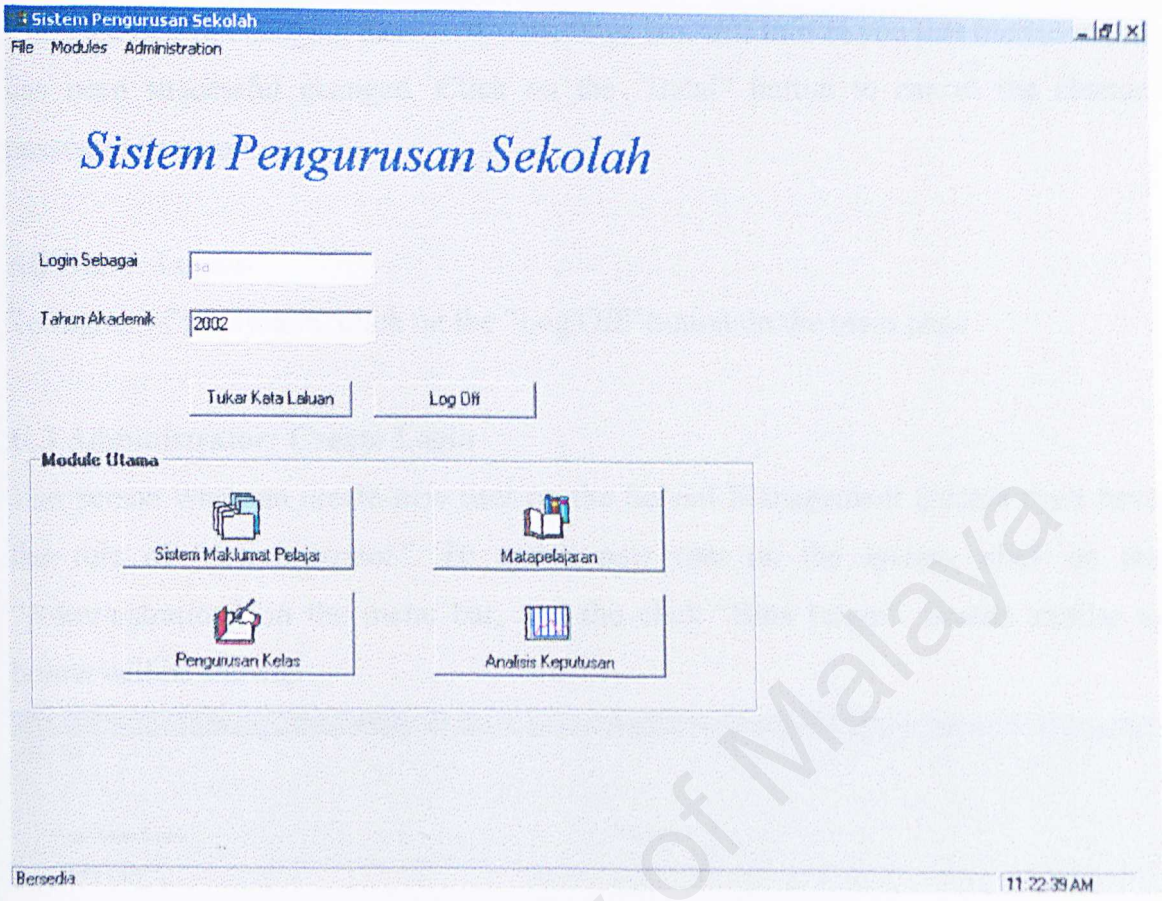


Figure2. User Manual: Screen of Main Page

All Users: Change Password

To change the password, click on the "Tukar Kata Laluan" button at the main page. The change password form will appear.

Figure3. User Manual: Screen of Change Password

Enter old password ("Kata Laluan Lama"), new password ("Kata Laluan Baru") and confirm the new password ("Sahkan Kata Lauan") and click on the "OK" button. Upon success in changing password, a message box will inform you that the password has been successful changed. Click on the "Batal" button to cancel the change password function.

All Users: Logout

To logout of the system, click on the "Log Off" button on the main page.

U.3 Administrator: Create Login

The person who can create new user on the School Management System must have the role of "Administrator". To create new user on the system, click on the "Administration" on the menu bar, and the click "Bina Login". Screen similar to below will be shown.

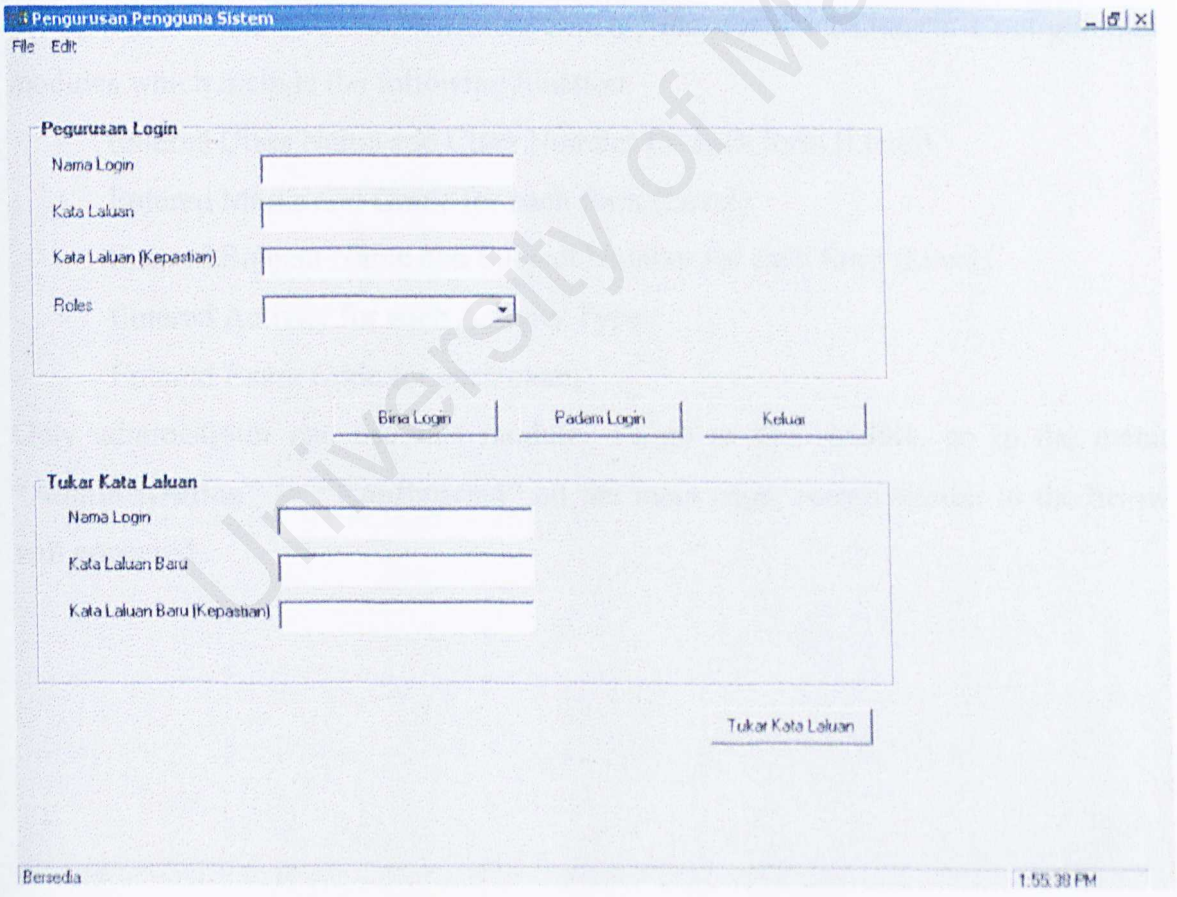


Figure 4. User Manual: Screen of Create Login

In the "Pengurusan Login", entered all the information, select the role of the user, and click on the "Bina Login" button, a new user will be created. The login name ("Nama Login") should be the same with Teacher ID or the Staff ID.

Administrator: Delete Login

On the same page, just entered the login name and click on the "Padam Login" button. A message box will appear to inform you that the login is successfully deleted.

Administrator: Change password

If a user forgets his or her password, the administrator can reset their password.

On the same page, in the Tukar Kata Laluan", just entered the login name of the user, and their new password, and click on the "Tukar Kata Laluan" button.

U.4 Administrator: Configuration

To customized the needs of different school, this system include a Configuration modules which include the following function:

- Entered Class Name and Class Number for each form (Level).
- Entered Marks and Grade for each form (Level).
- Entered Subject Name and Subject Number for each form (Level)
- Entered Activity for each Activity Type.
- Entered Exam Code for each exam.

Only administrator can use this module. To go to this module, go to the menu **"Administration" -> "Konfigurasi"** on the main page, screen similar to the below will appeared.

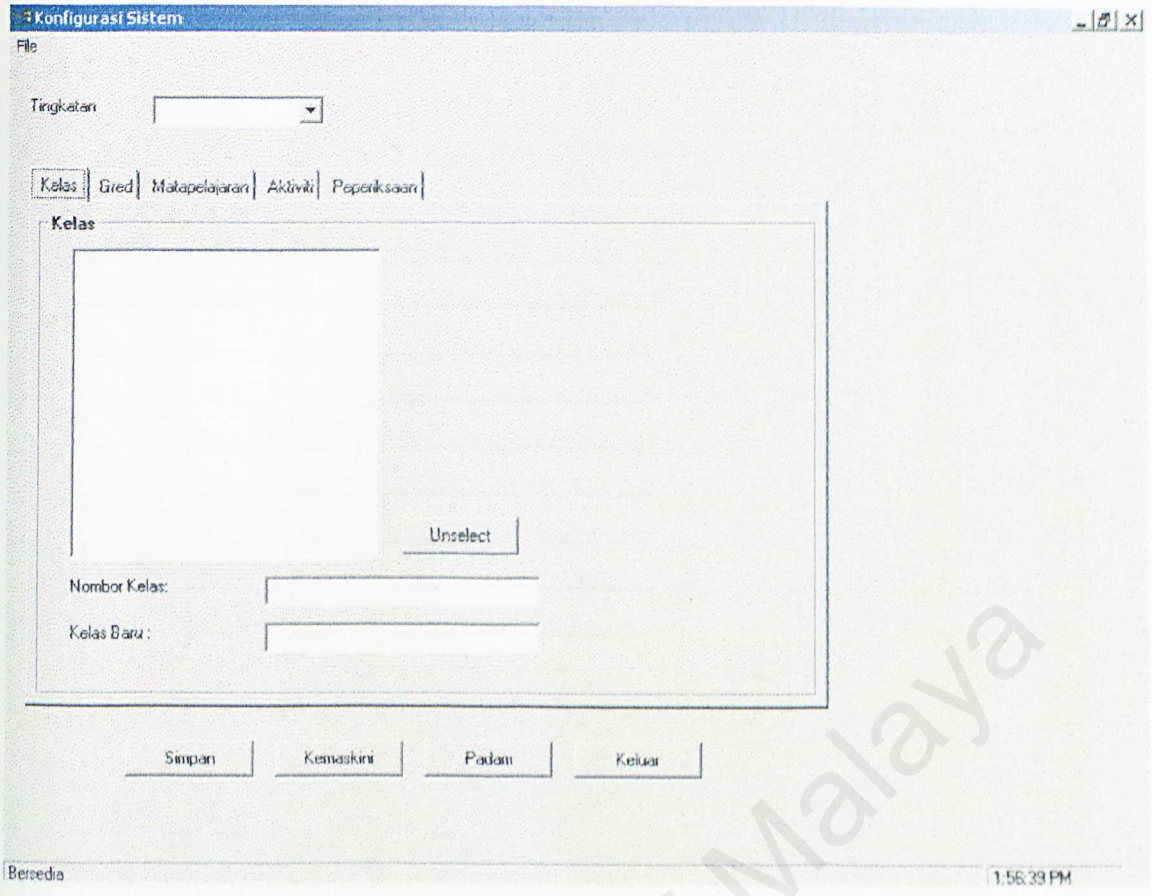


Figure 5. User Manual: Screen of Configuration Main Page

For tab "Kelas":

To create a new form (Level), go to menu "File" -> "Tingkatan Baru", entered the new form, class name and class number for that form and click "Simpan". To create class for an existing form, just select the particular form in the combo box "Tingkatan", and entered the class number and class name and click "Simpan". All the class number has to be a unique number.

To modify the class number or the class name, just select the class that need to be modify and entered new class number or class name, and click on the "Kemaskini" Button. To delete a class, just select a class and click "Padam" button.

For tab "Gred":

Select a form (level) from the "Tingkatan" combo box. Entered the marks, grade and description and click "Simpan". To Update the grade, select the radio button for the row of that you wish to update and click "Kemaskini". To delete a row, just select the radio button and click "Padam".

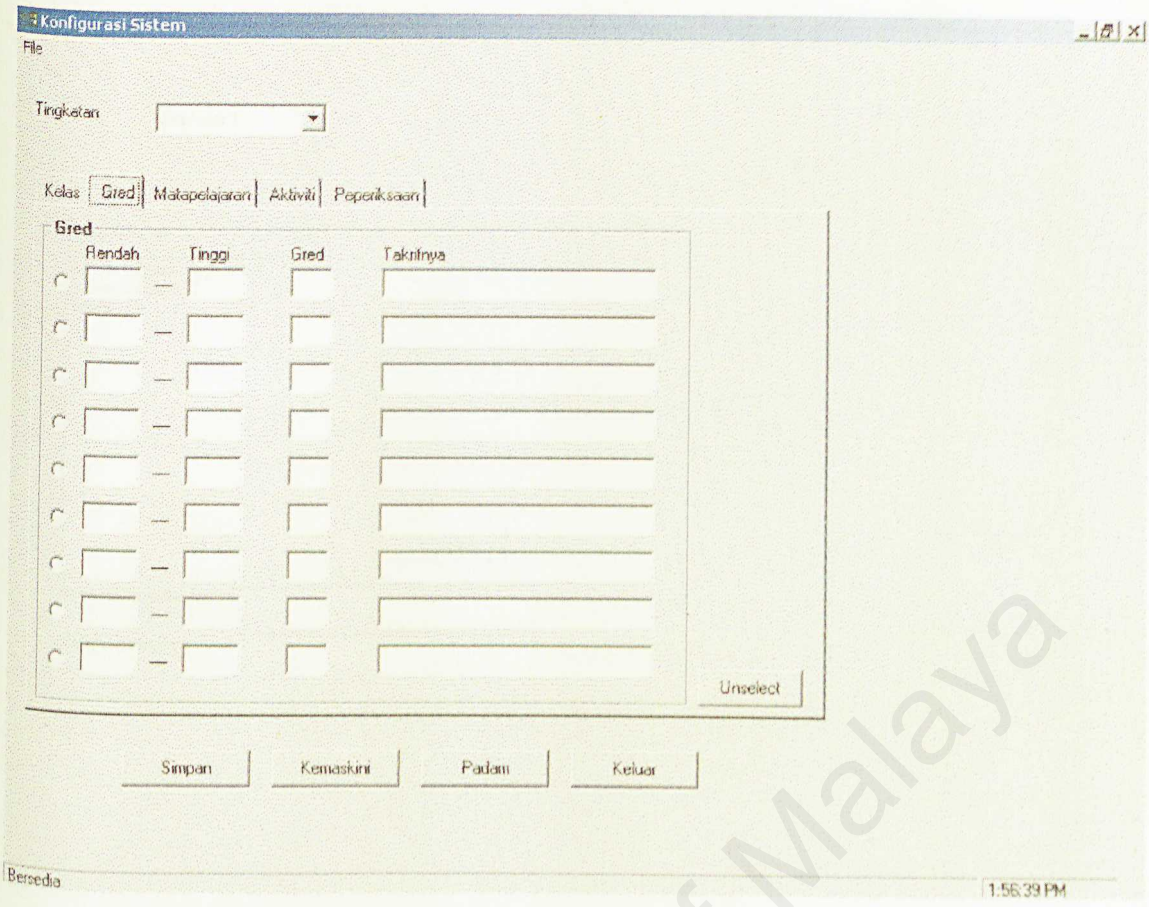


Figure 6. User Manual: Screen of tab Grade for Configuration

For tab "Matapelajaran":

To create a new form (Level), go to menu "File" -> "Tingkatan Baru", entered the new form, class name and class number for that form and click "Simpan". To create class for an existing form, just select the particular form in the combo box "Tingkatan", and entered the class number and class name and click "Simpan". All the class number has to be a unique number.

To modify the class number or the class name, just select the class that need to be modify and entered new class number or class name, and click on the "Kemaskini" Button. To delete a class, just select a class and click "Padam" button.

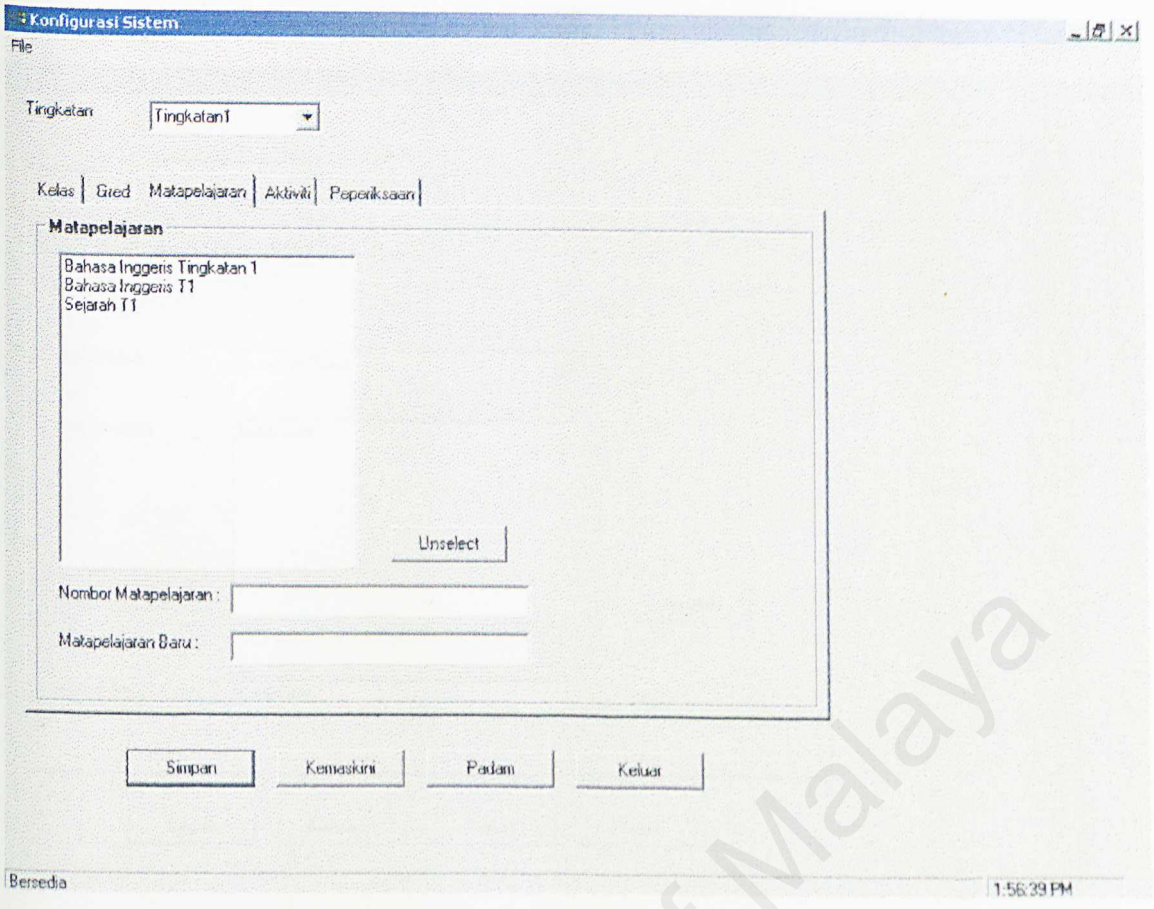


Figure 7. User Manual: Screen of tab Subject in Configuration

For tab "Aktiviti":

To create a new activity type, click menu File -> Jenis Activiti Baru, entered new type of activity and an activity, and click "Simpan". After the Activity type is save, you can enter the new activity and just click "Simpan". To update the activity name, just select an activity and entered the new activity name and click "Kemaskini". To delete an activity, just select the activity and click "Padam".

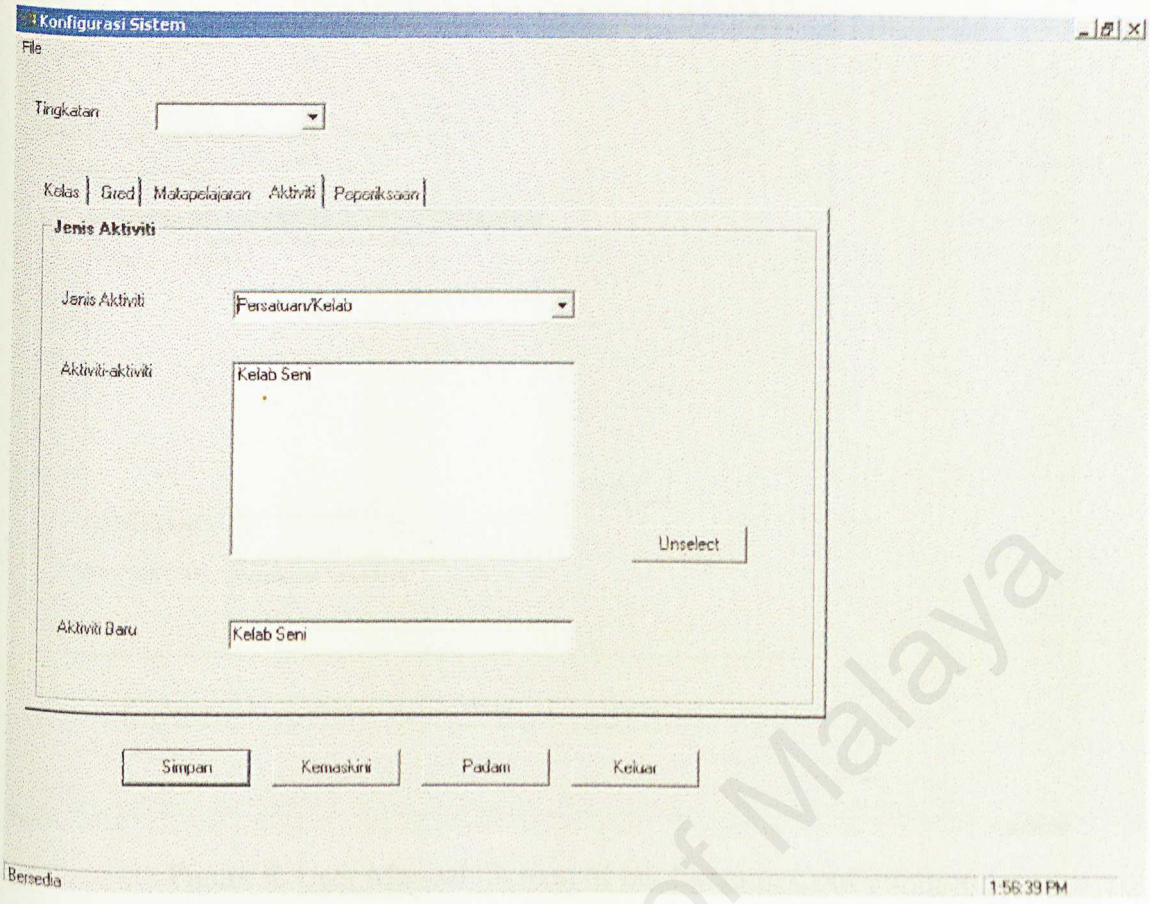


Figure 8. User Manual: Screen of Activity in Configuration

For tab "Peperikasaan":

Just entered the exam code in "Kod Peperiksaan", and the Type of Exam in "Jenis Peperiksaan" and click "Simpan". To update, click on the Type of Exam that you wish to update and entered new code and type of exam and click "Kemaskini". To delete the Type of Exam, just select the type of exam that you want to delete and click "Padam".

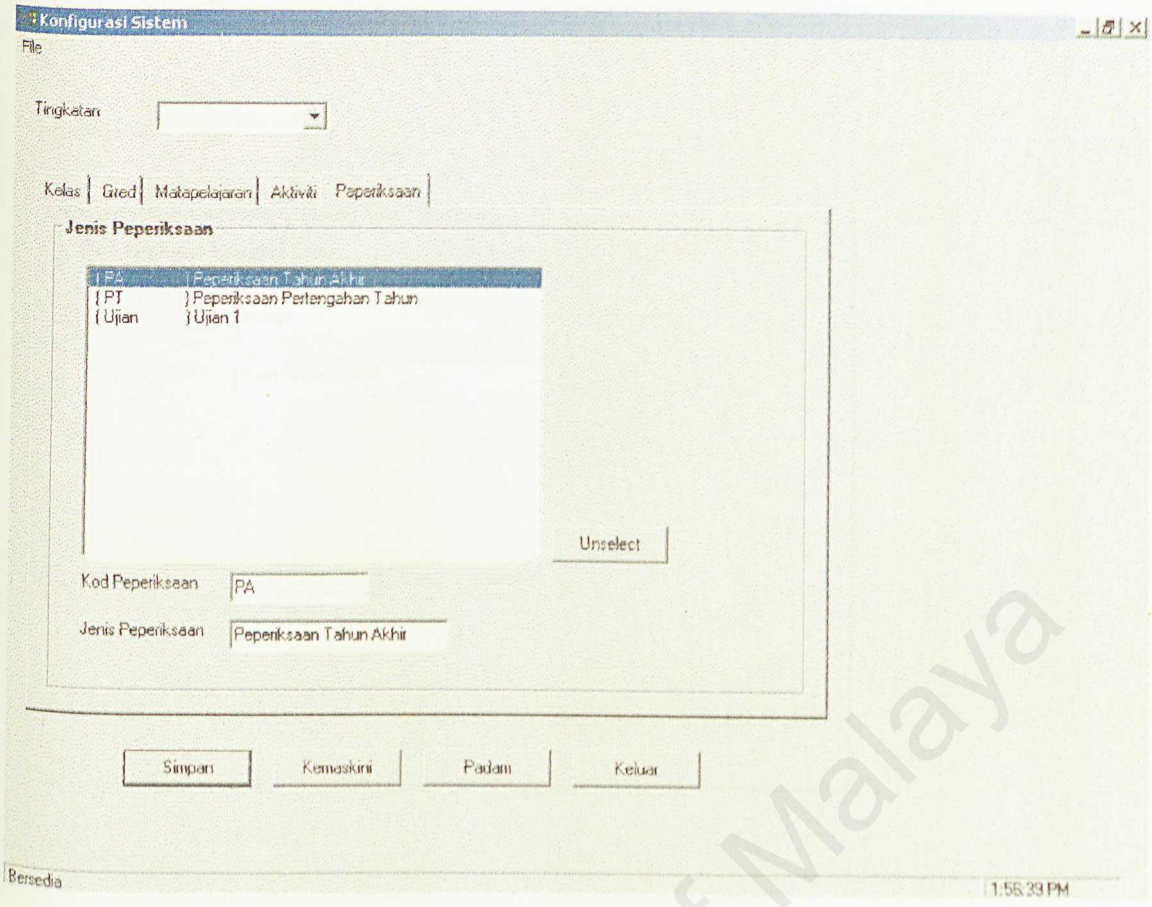


Figure 9. User Manual: Screen of tab Exam in Configuration

U.5 Administrator: "Guru"

Go **to Administration - > Guru** on the main page to enter this module. This module is to let the administrator to enter the information on:

- Form teacher and class
- Subject teach by each teacher
- Teacher Advisor for each Activity
- Teacher teach subject for which class

For Tab "Guru Tingkatan":

Select a class and a teacher and click "Simpan". Form teacher for that class in save. To change the form teacher for a class, just select a new teacher in the "Guru Tingkatan" and click "Kemaskini".

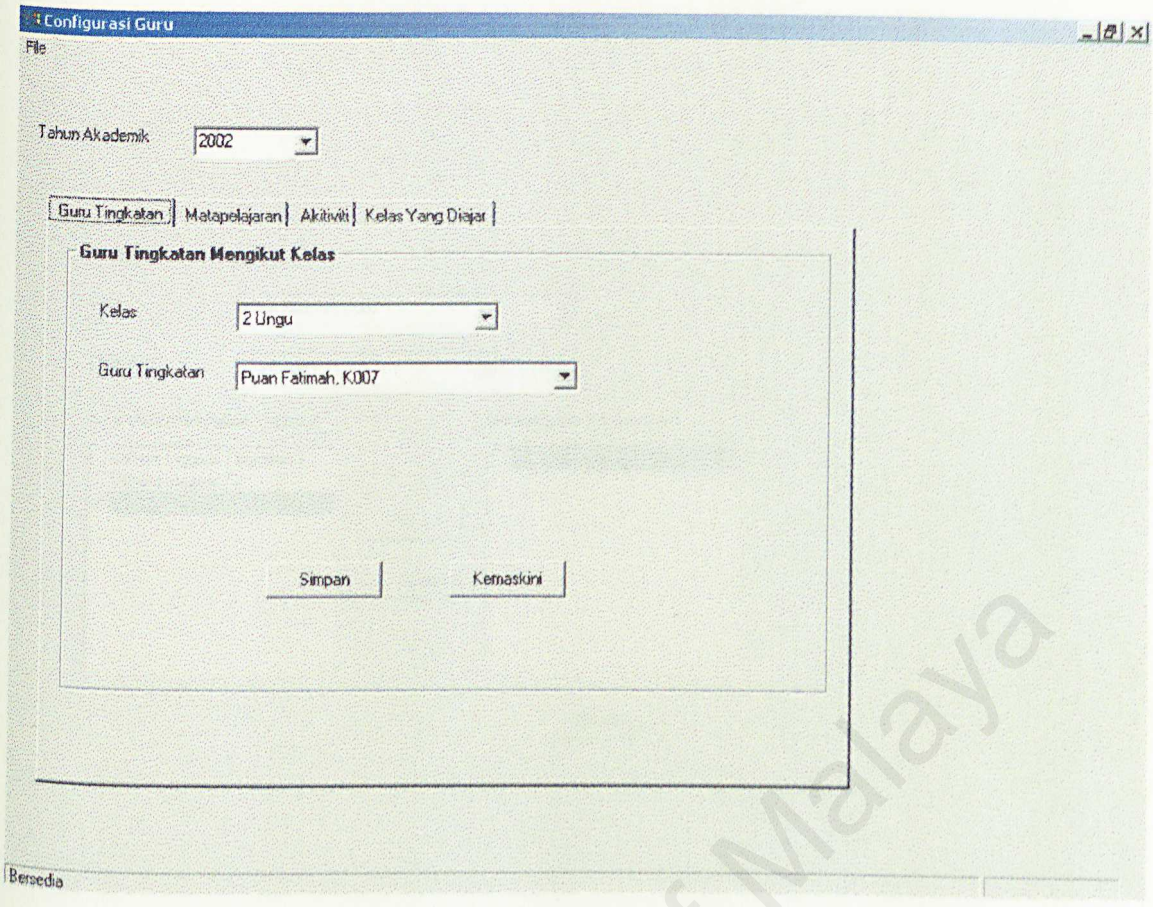


Figure 10. User Manual: Screen of tab Form Teacher in Teacher Configuration

For Tab "Matapelajaran":

Select a teacher, form and subject that a teacher teach and click "Pilih". If the subject belongs to another form, select the form and choose the subject and click "Pilih". After all the subjects are chosen, click "Kemaskini".

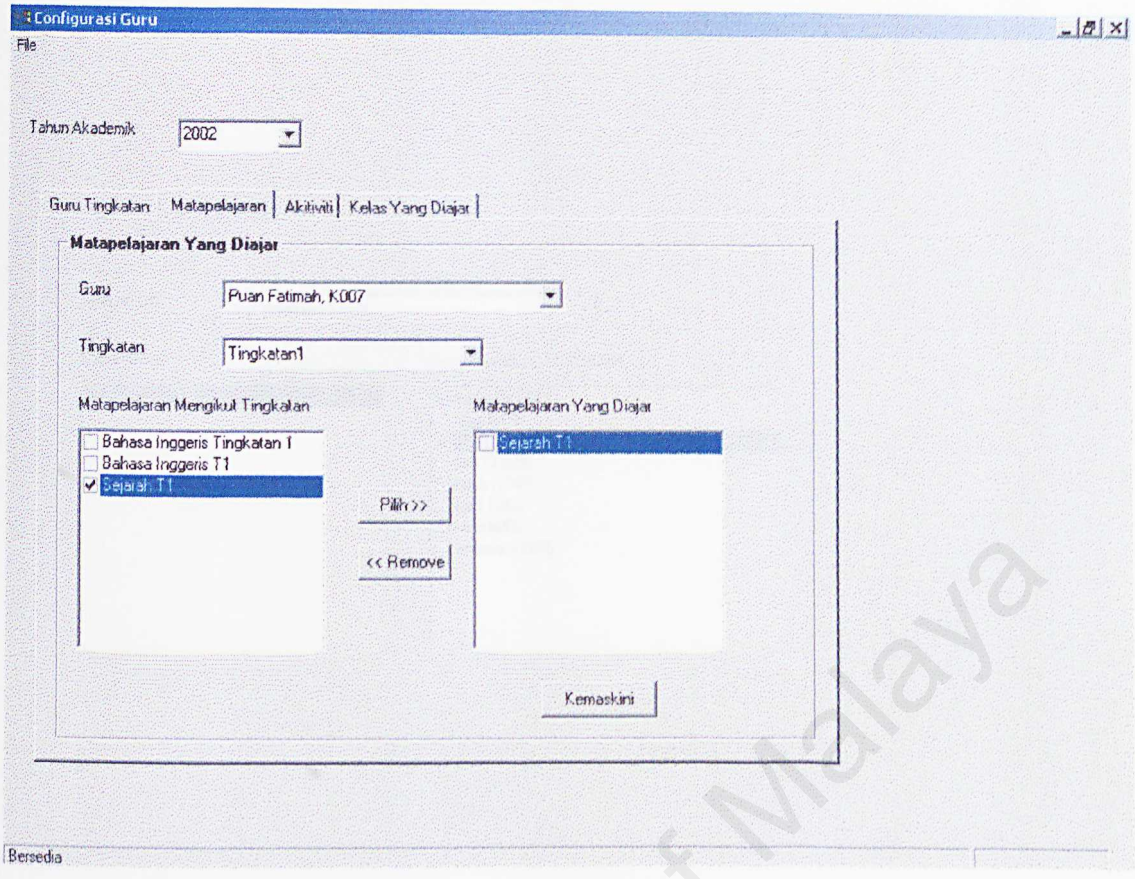


Figure 11. User Manual: Screen for tab Subject in Teacher Configuration

For Tab "Aktiviti":

Choose the type of activity in the "Jenis Aktiviti". Select the activity in the "Nama Aktiviti" and choose the teacher, who are the teacher advisors for this activity. After that, click "Kemaskini".

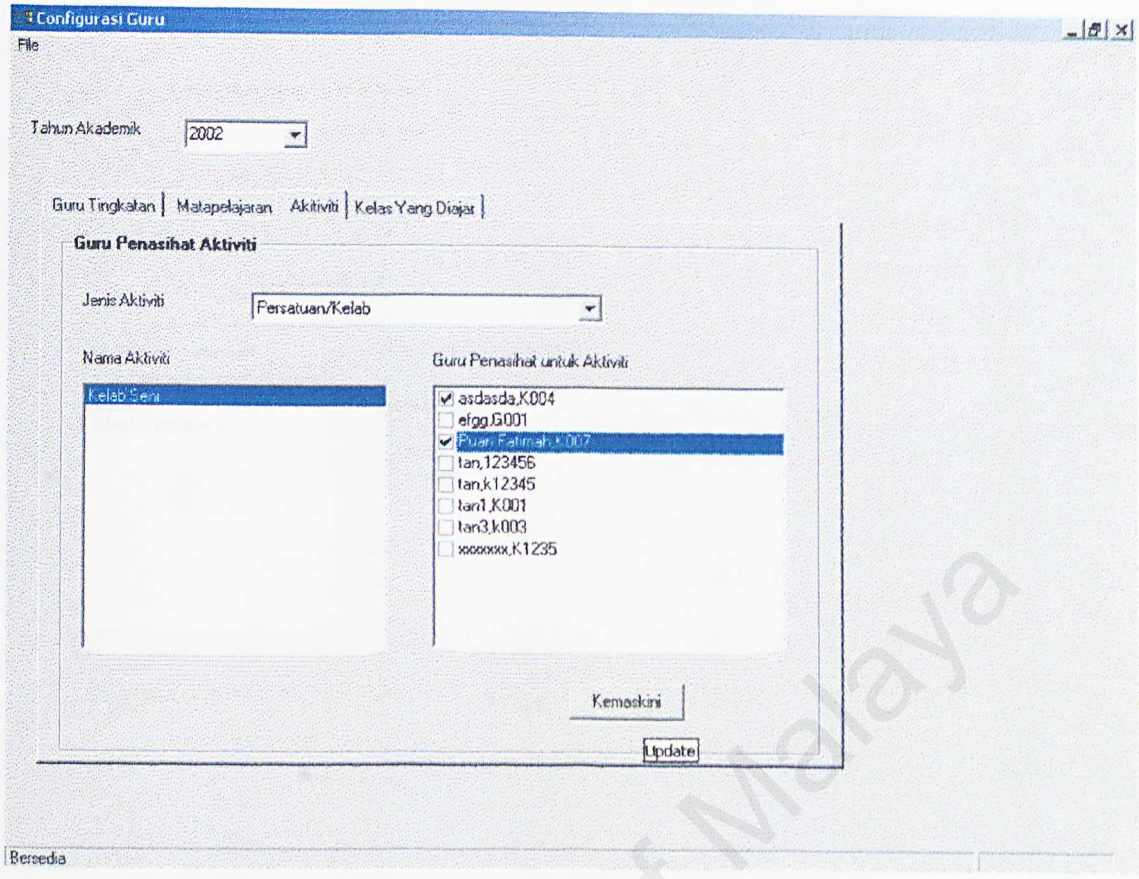


Figure 12. User Manual: Screen of tab Activity in Teacher Configuration

For Tab "Kelas Yang Diajar":

Select a teacher, and subject that the teacher teach. Choose the class that the teacher teaches for that subject and click "Kemaskini".

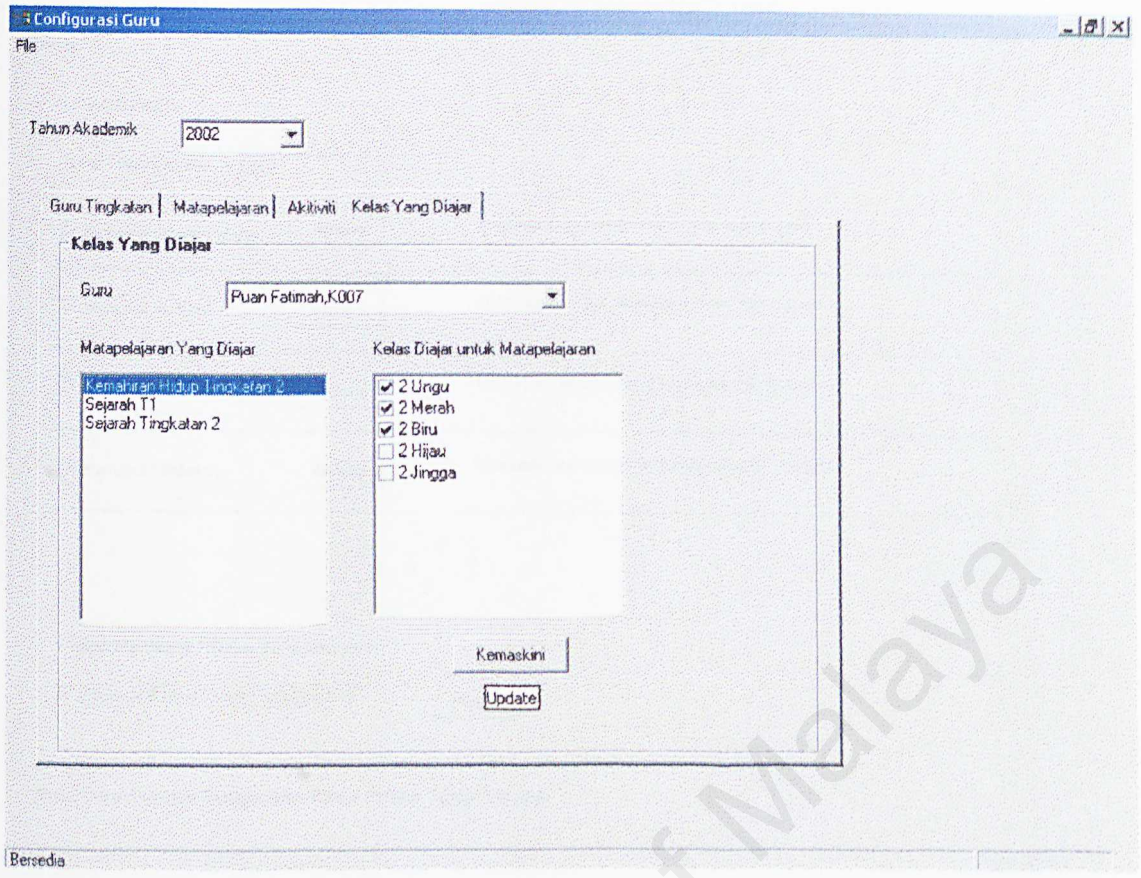


Figure 13. User Manual: Screen of tab Class Teach in Teacher Configuration

U.6 Administrator: Archive

For Archive Student: Go to **Administration->Maintenance->Archiving->Archive Pelajar.**

For this function, only student information such as personal particulars, parent or guardian information, medical information will be copy to temporary tables. Student information to be archive will base on the year that the students leave the school.

Enter the year that the students leave school, select the information that needs to be archive and click on the button "Archive".

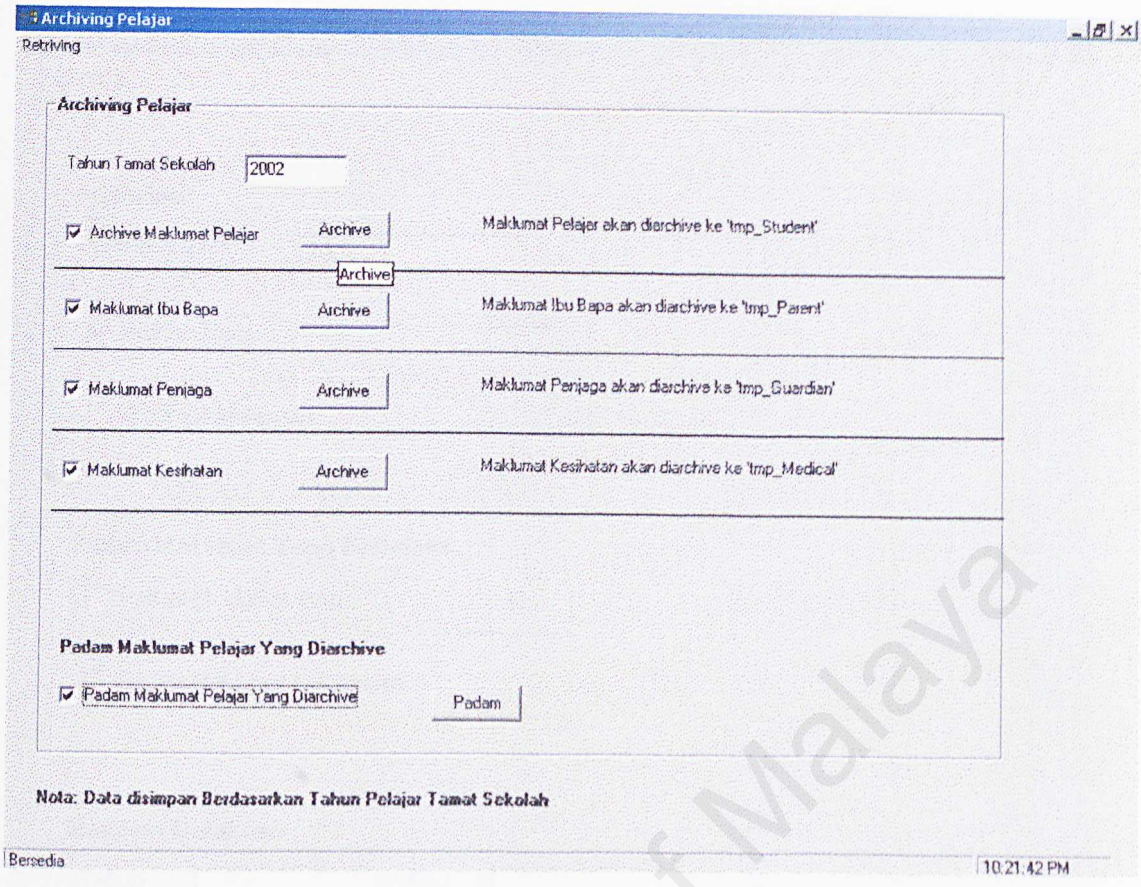


Figure 14. User Manual: Screen of Archive Student

After archiving, you can choose to delete all the information regarding the student such as their personal information, result, class, and subject taken.

For Archive Staff/ Teacher: Go to **Administration->Maintenance->Archiving->Archive Guru/Kakitangan**

For this function, teacher or staff personal information will be archive to a temporary table. Only teacher or staff information that leave school will be archive. After archiving, administrator can choose to delete the teacher/staff information that has been archive. In the "Maintain Database", administrator can choose to delete information such as class teacher, teacher adviser, subject teach according to the year entered by the administrator.

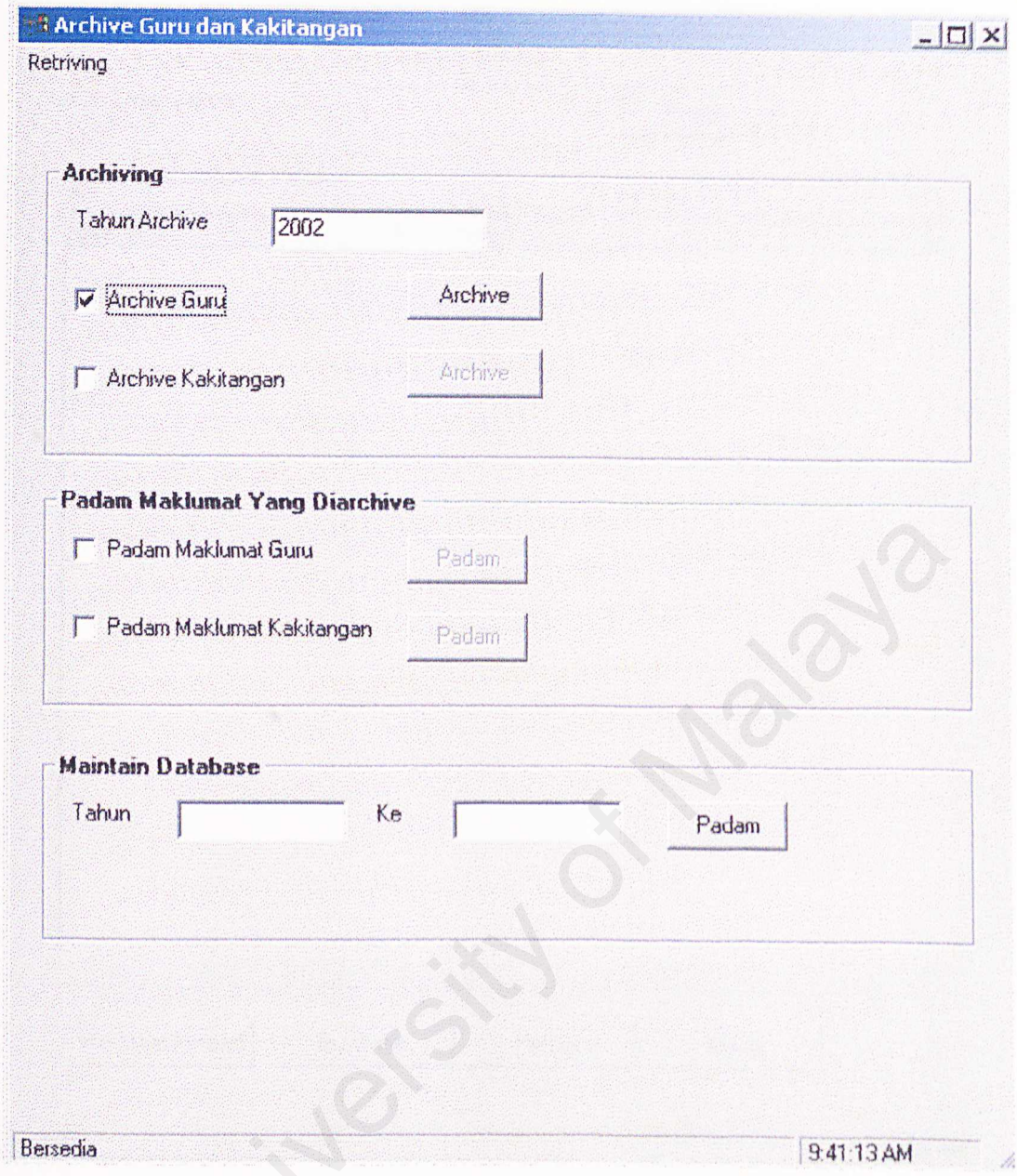


Figure 15. User Manual: Screen of Archive Teacher/Staff

Only administrator should use the archiving function after careful planning to avoid losing information.

U.7 Administrator: Retrieve

For retrieve student information: Go to **Administration->Maintenance-> Retrieving -> Retrieve Pelajar**. A screen similar to below will appeared.

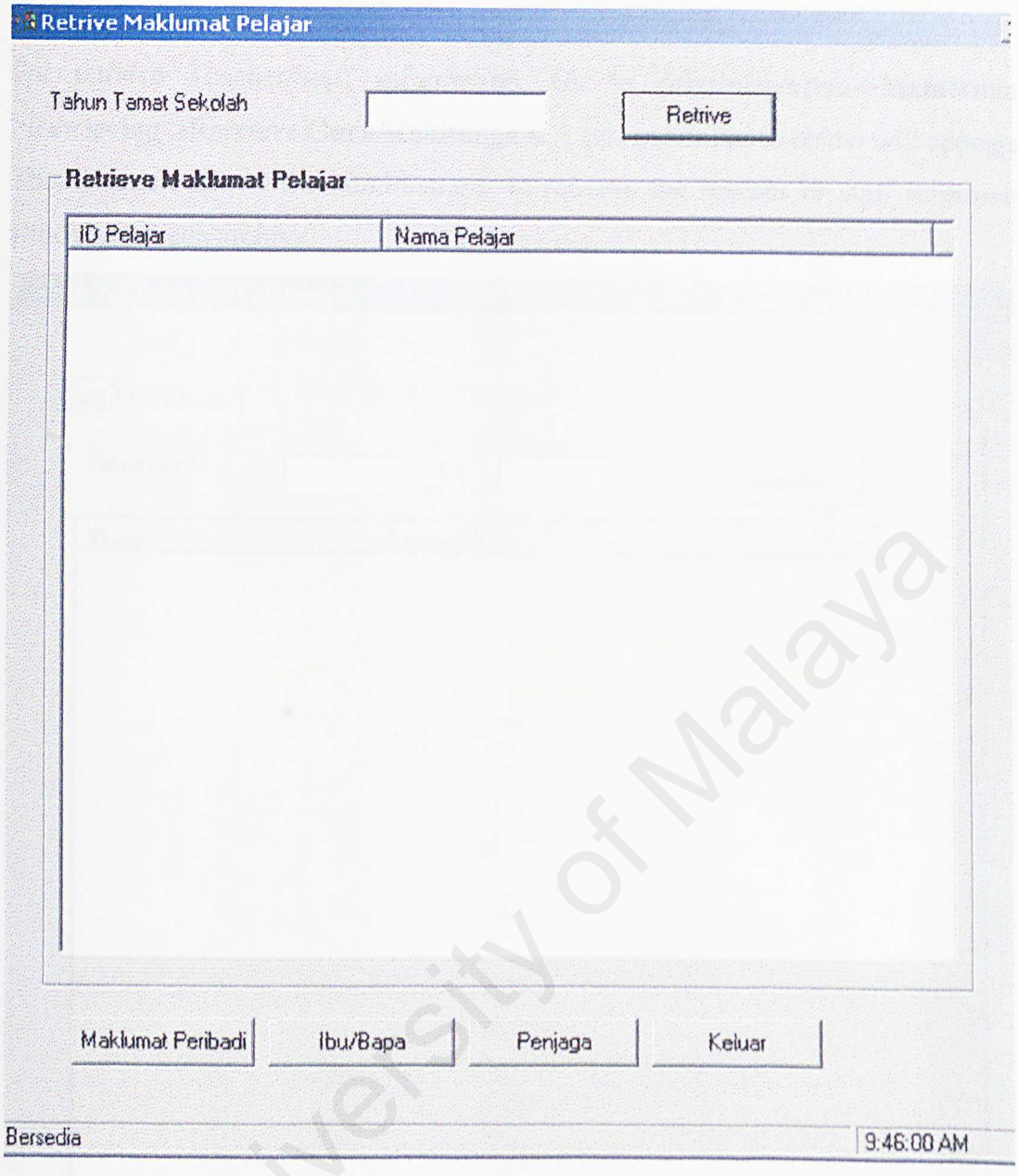


Figure 16. User Manual: Screen of Retrieve Student

This function allows the administrator to retrieve the student information, which have been archived.

Enter the year of student leave school, and click "Retrieve" button. A list of students that left the school at the year specified by the administrator will be display. Select a student, click on the "Maklumat Peribadi" button to view the student personal and medical information. Click on the "Ibu/Bapa" button to view the student parents' information. Click on the "Penjaga" button to view the student guardian information. Click on the "Keluar" button to exit. All the information will be display in the report format.

For retrieve Teacher/Staff information: Go to **Administration->Maintenance->Retrieving->Retrieve Guru/Kakitangan.** A screen similar to below will appeared. This function allows the administrator to retrieve the teacher or staff information, which have been archive.

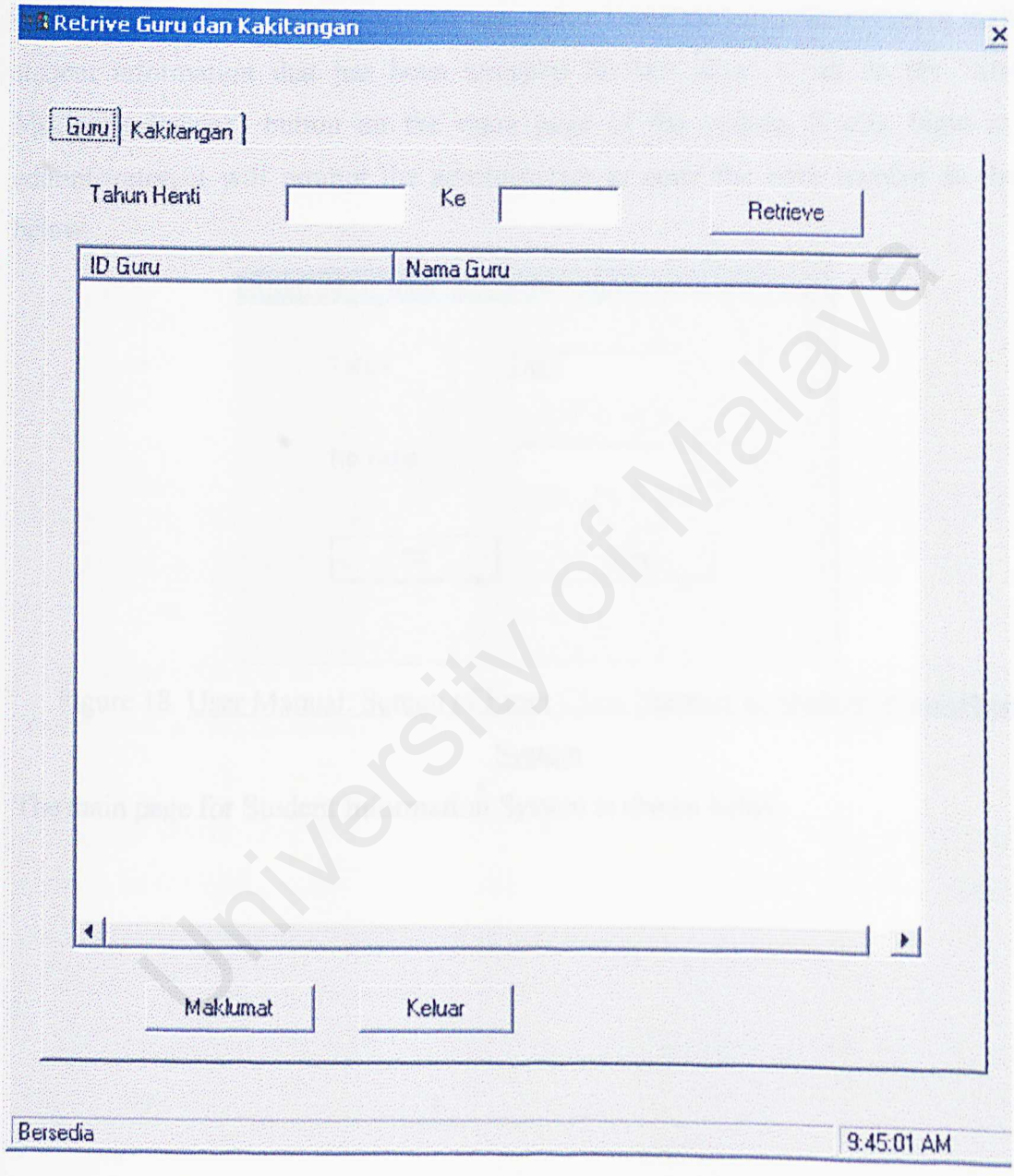


Figure 17. User Manual: Screen of Retrieve Teacher/Staff

To retrieve teacher, enter the year that the teacher left the school and click "Retrieve" button. A list of teacher that left school at the year specified by administrator will be display. Select a teacher; click on the "Maklumat" button, the teacher information will be display in the report format. To retrieve the staff information, click on the

"Kakitangan" tab, and the way to retrieve information is the same as the way to retrieve teacher information. Click on the "Keluar" button to exit.

U.8 Student Information System ("Sistem Maklumat Pelajar")

For this module, only form teacher that have been assigned to class can access this module. In this module, form teacher can entered new student information or updates student information that has been assigned for this class. Click on the "Sistem Maklumat Pelajar" button on the main page of the system. If user login as an administrator, it will prompt the administrator to enter the class number as shown below.

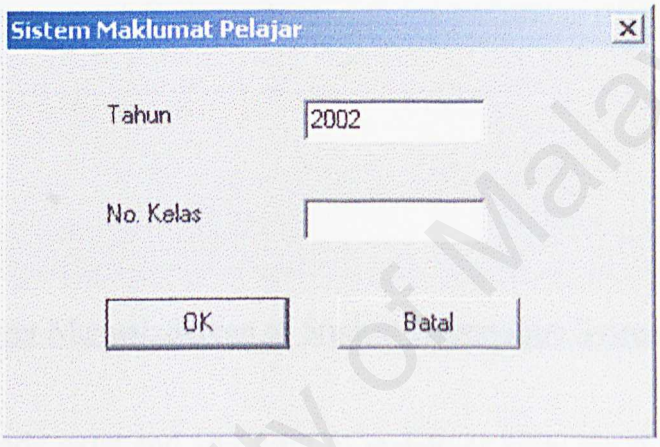


Figure 18. User Manual: Screen of Enter Class Number in Student Information System

The main page for Student Information System is shown below.

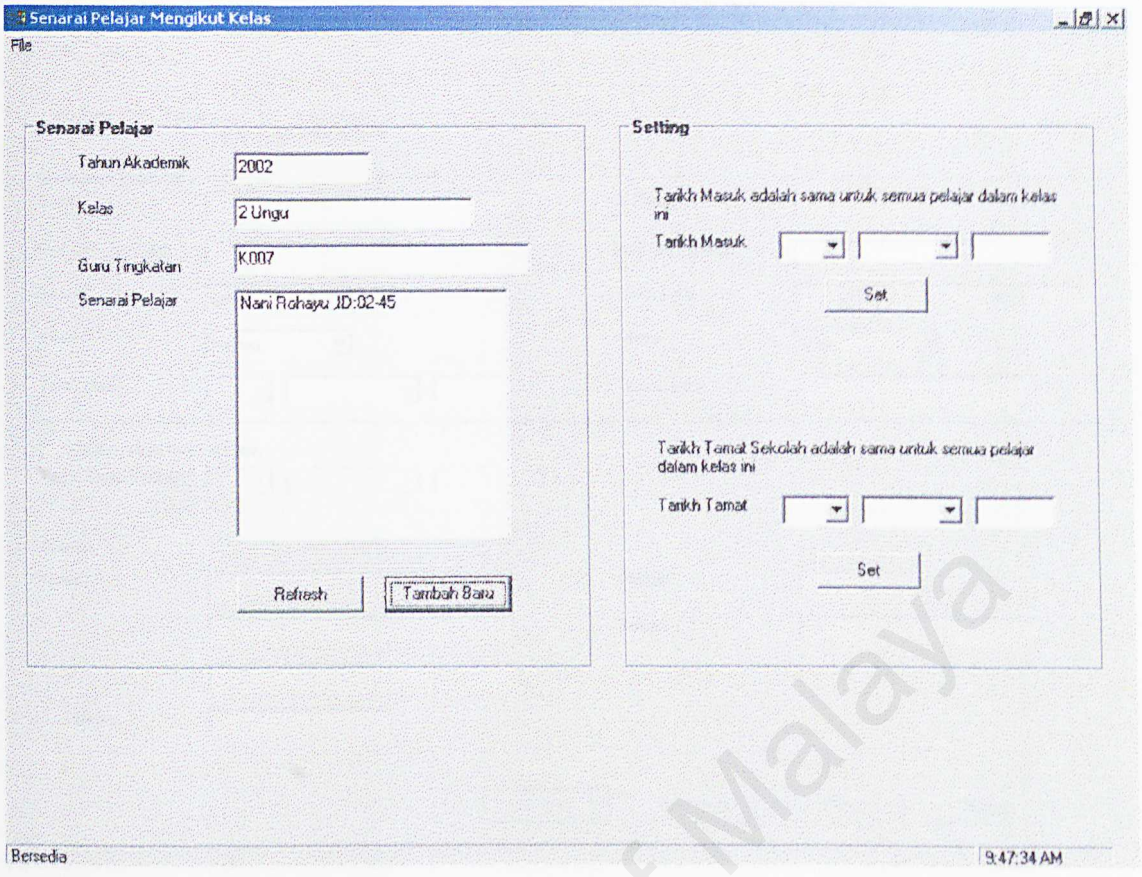


Figure 19. User Manual: Screen of Student Information System Main Page

If the student already exist and you want to update or view the student information, double click on the name of the student in the "Senarai Pelajar" list box.

If you want set the Date of Enrollment ("Tarikh Masuk") which is the same for all the students in the class, select the date and click "Set" button. The steps for setting the Date of Leaving School ("Tarikh Tamat") are same.

To add new student information, click on the "Tambah Baru" button. A screen shown below is shown.

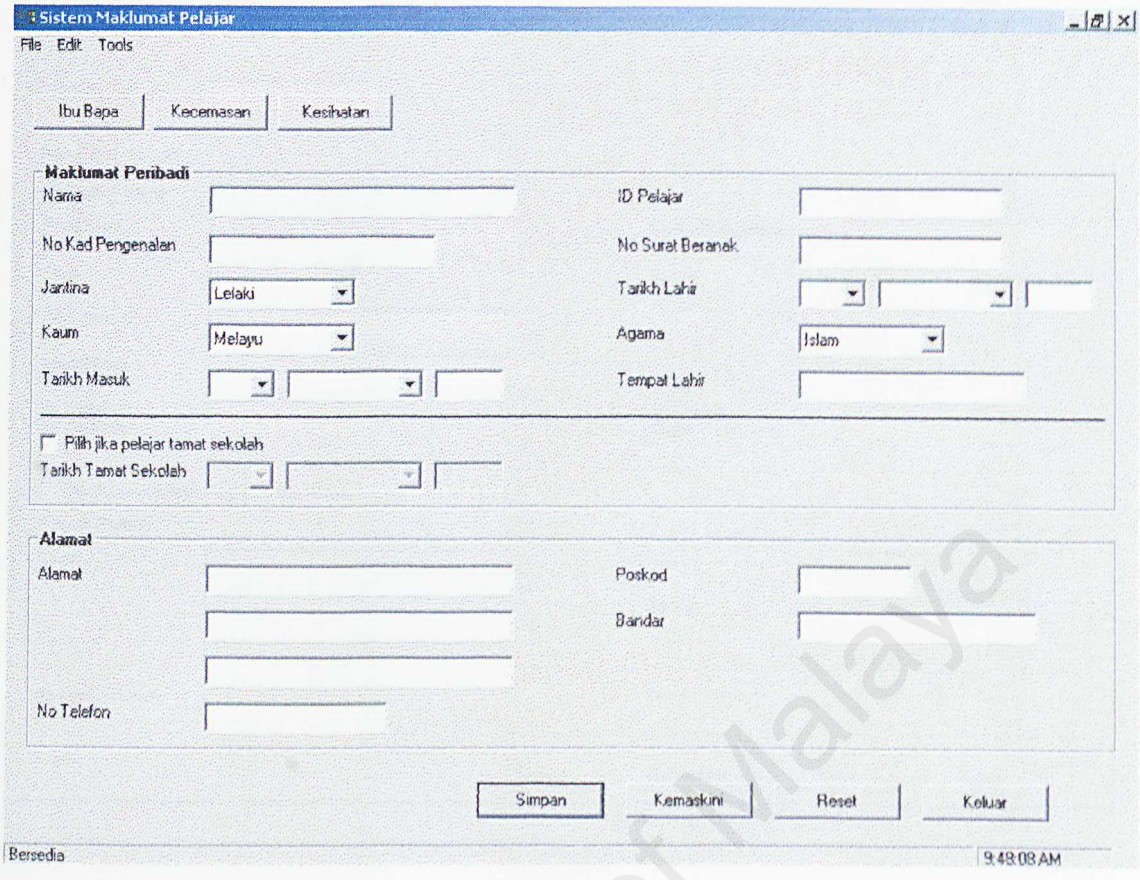


Figure 20. User Manual: Screen of Student Information

To save information, click on "Simpan", to updates information, click on "Kemaskini", to clear all the fields, click on the "Reset" button. To exit from this page, click on the "Keluar" button. To print report on the student personal information, go to **Tools -> Laporan Maklumat Pelajar**. To print report on student medical information, go to **Tools-> Laporan Kesihatan.**

From this page, click on the "IbuBapa" button on top of the page to enter parents and guardian information.

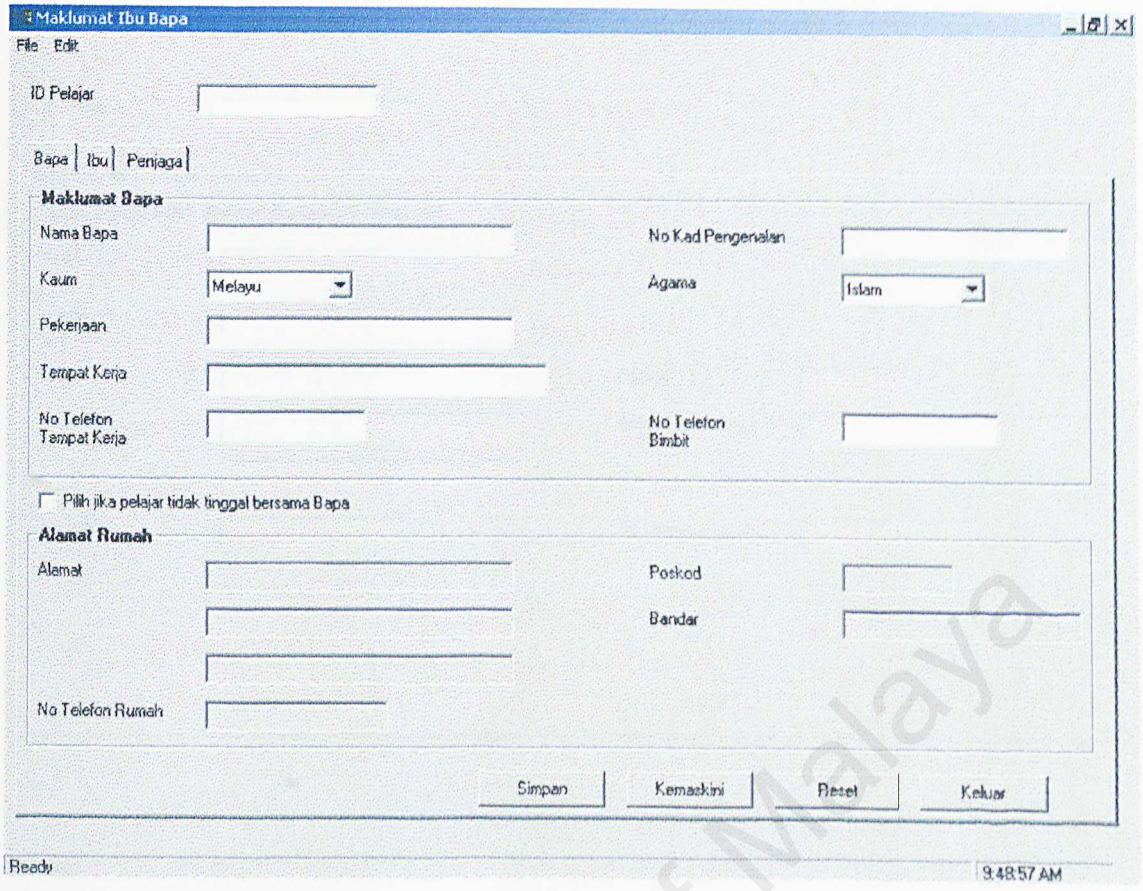


Figure 21. User Manual: Screen of Parents Information

Click on the "Kecemasan" button, if you want to enter information on student emergency contact.

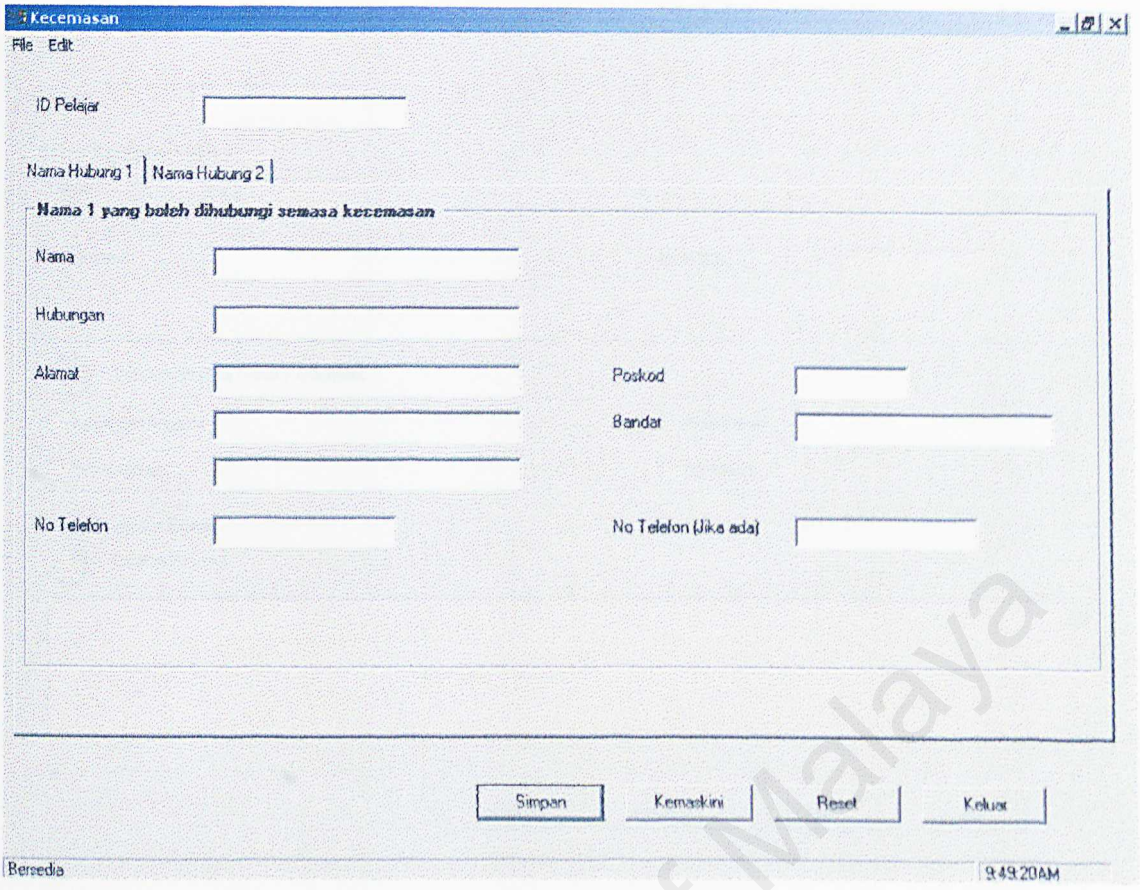


Figure 22. User Manual: Screen of Emergency Contact

Click on the "Kesihatan" button, if you want to enter information on student medical information.

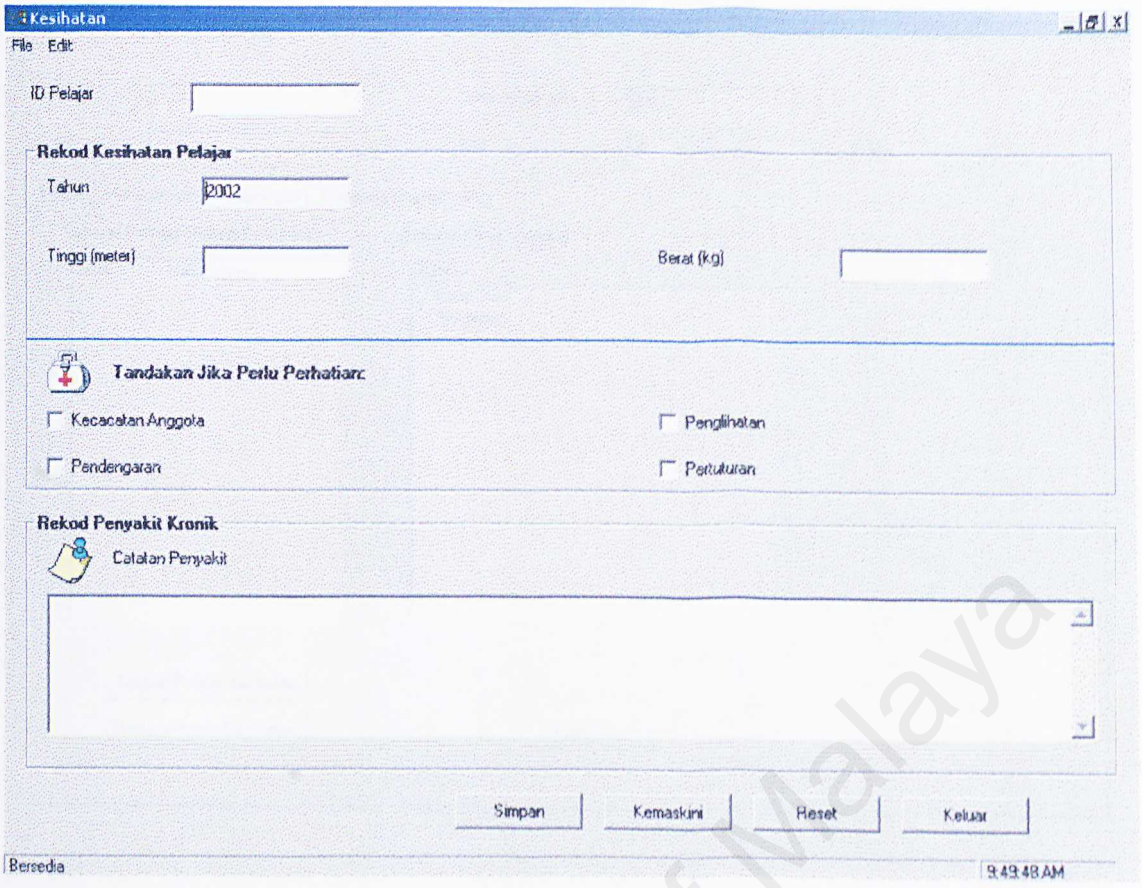


Figure 23. User Manual: Screen of Student Medical Information

U.9 Class Management ("Pengurusan Kelas")

To access this module, click on the "Pengurusan Kelas" button in the main page or go to **Modules->Pengurusan Kelas**. If user login as administrator, the administrator will be prompt to enter the teacher ID as shown below.

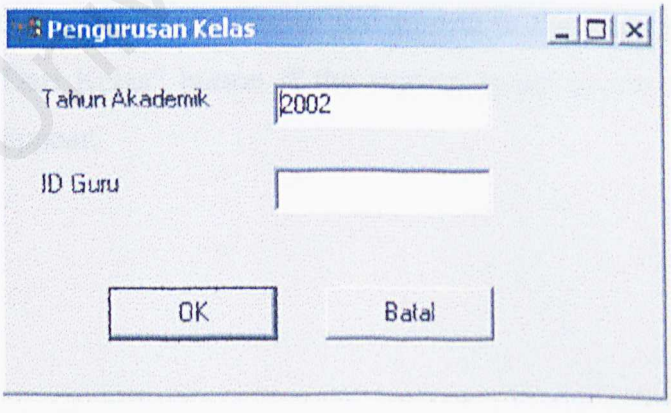


Figure 24. User Manual: Screen of Enter Teacher ID in Class Management

For this module, only form teacher that been assigned to class can access this module. For example, if a teacher with ID "K007" is the form teacher for class "2 Ungu". The screen shown below is show.

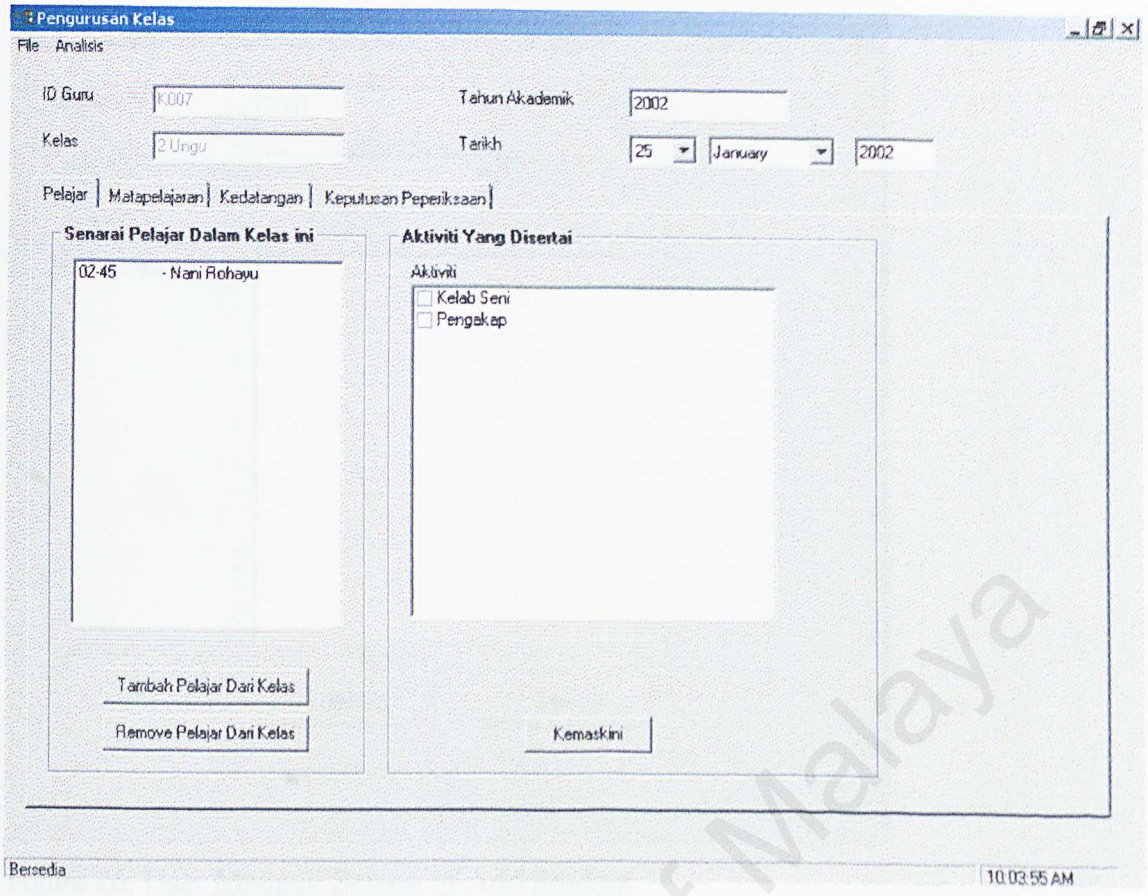


Figure 25. User Manual: Screen of tab Student in Class Management

For tab "Pelajar", it will list all the student name and ID for this class. To set what are the activity join by each student in this class, choose a student and select the activity join by the student in the "Aktiviti Yang Disertai" and click "Kemaskini".

If no student name is display, teacher can add student to this class by clicking on the "Tambah Pelajar Dari Kelas" button if the student is not a new student. A screen shown below will appear.

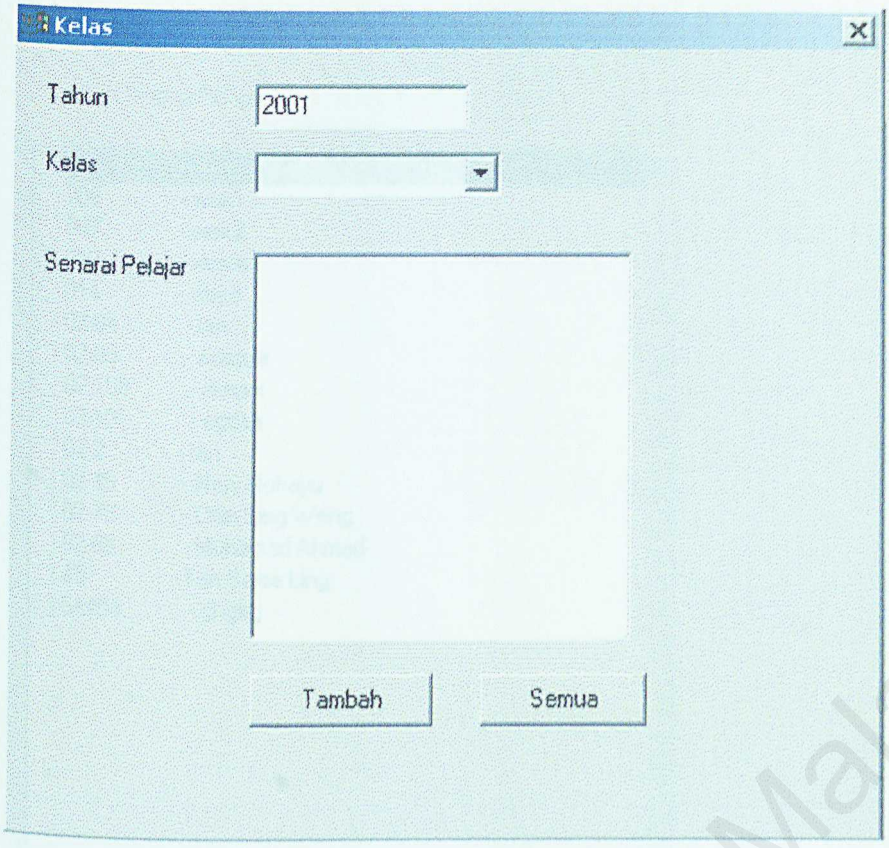


Figure 26. User Manual: Screen of Add Student

If the teacher wants to add student from class "1 Ungu" in the previous year, just choose the class name and select the student that need to be added and click "Tambah". If the teacher clicks on the "Semua" button, a list of the entire student in the school will be display; teacher will have another option of adding student from this list. This option is needed as the teacher might accidentally remove new student from the current class, as new student has not been assigned to any class for the previous year.

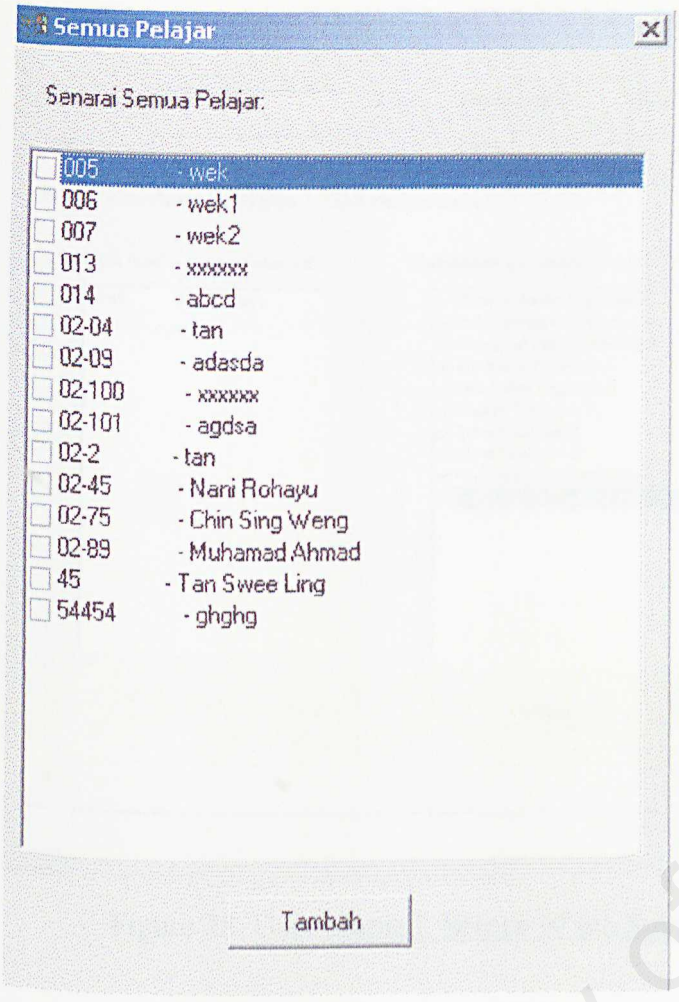


Figure 27. User Manual: Screen of List of All Students

For tab "Matapelajaran", it let the teacher keeps information on subjects taken by each student. Select a student in "Senarai Pelajar Dalam Kelas ini" and select the subjects taken by the student and click "Kemaskini".

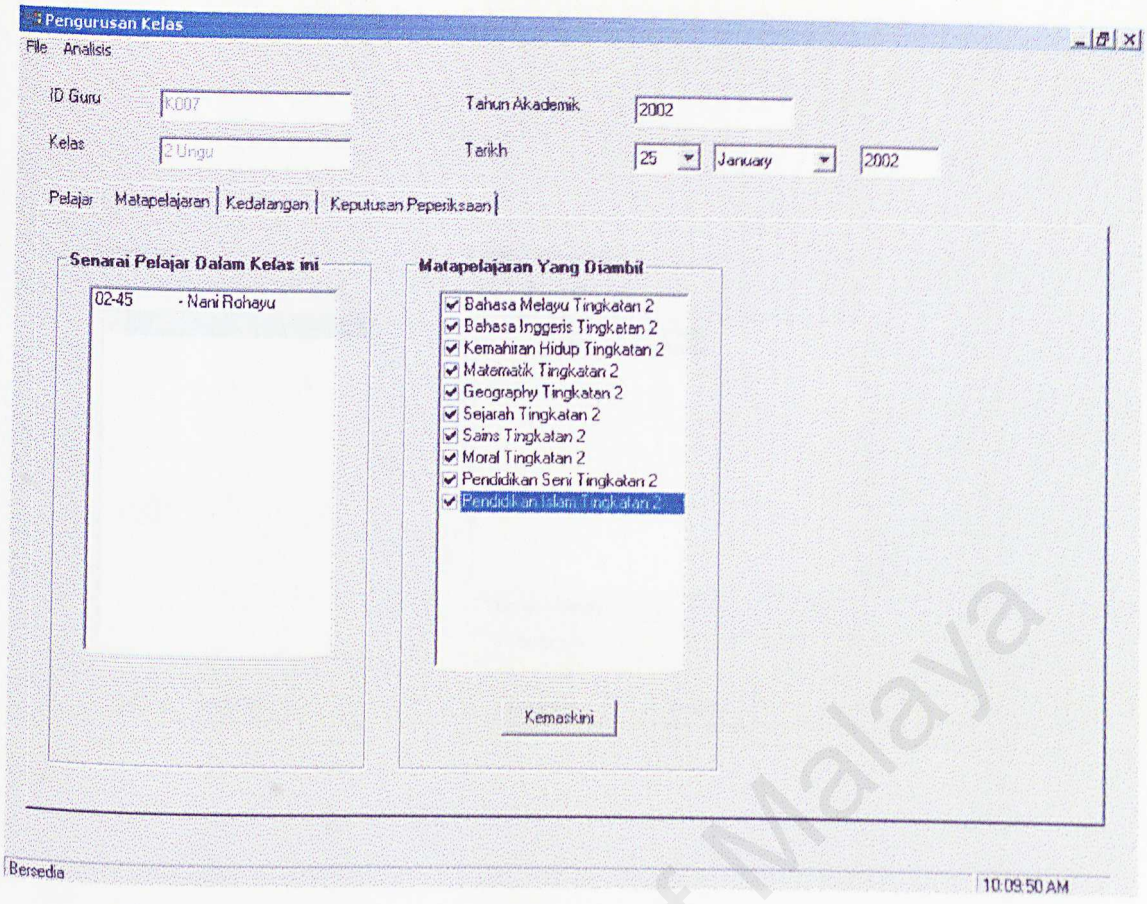


Figure 28. User Manual: Screen of tab Subject in Class Management

For tab "Kedatangan", it let the teacher manage the attendance of students in this class. In "Tandakan Pelajar Yang Tidak Hadir", select the students that are absent on that day and click "Kemaskini".

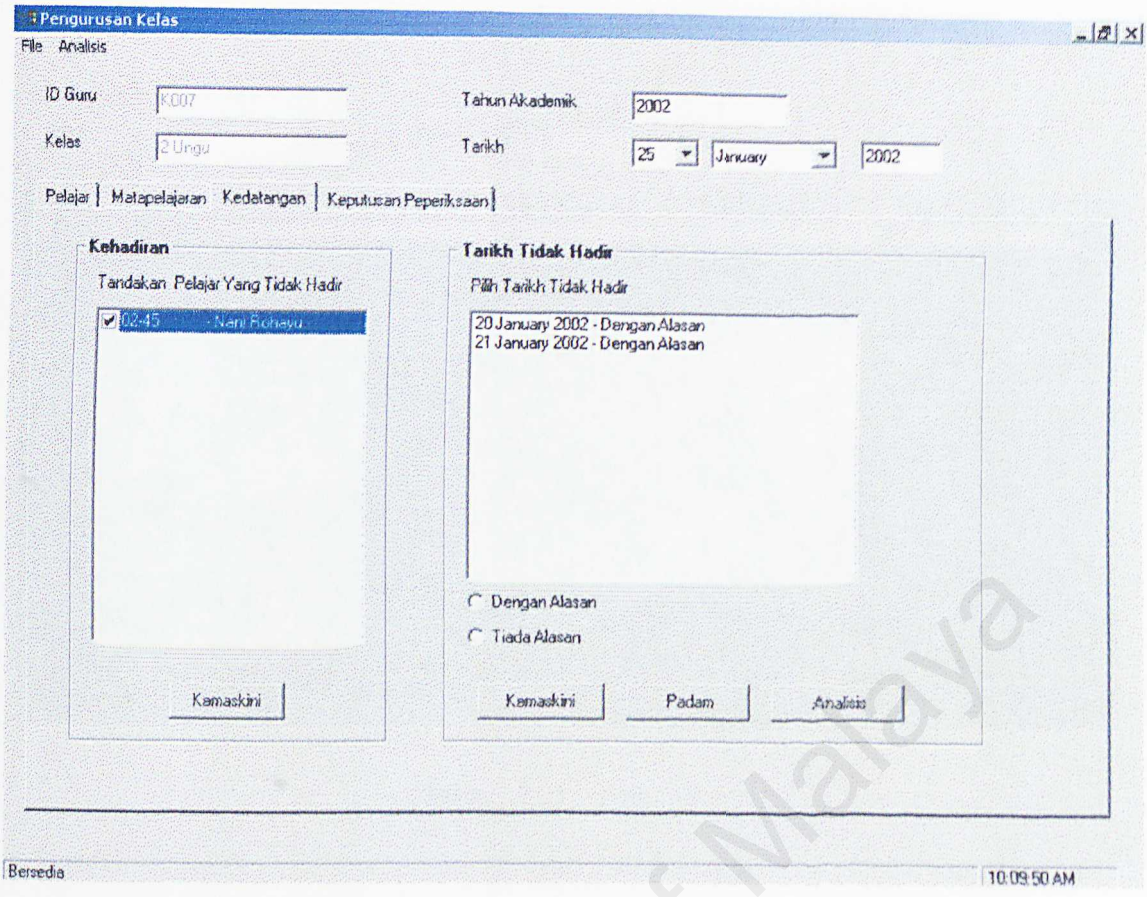


Figure 29. User Manual: Screen of tab Attendance in Class Management

The date of absent for the selected student will be kept. By default, the student will be assumed as absent with excuse. To change it, just select the student, select the date of absent for that student, select the radio button "Tiada Alasan" and click "Kemaskini". To delete the date of absent, just select the student, select the date of absent and click "Padam". To display the analysis of student attendance, click on the "Analisis" button. A screen shown below will be display.

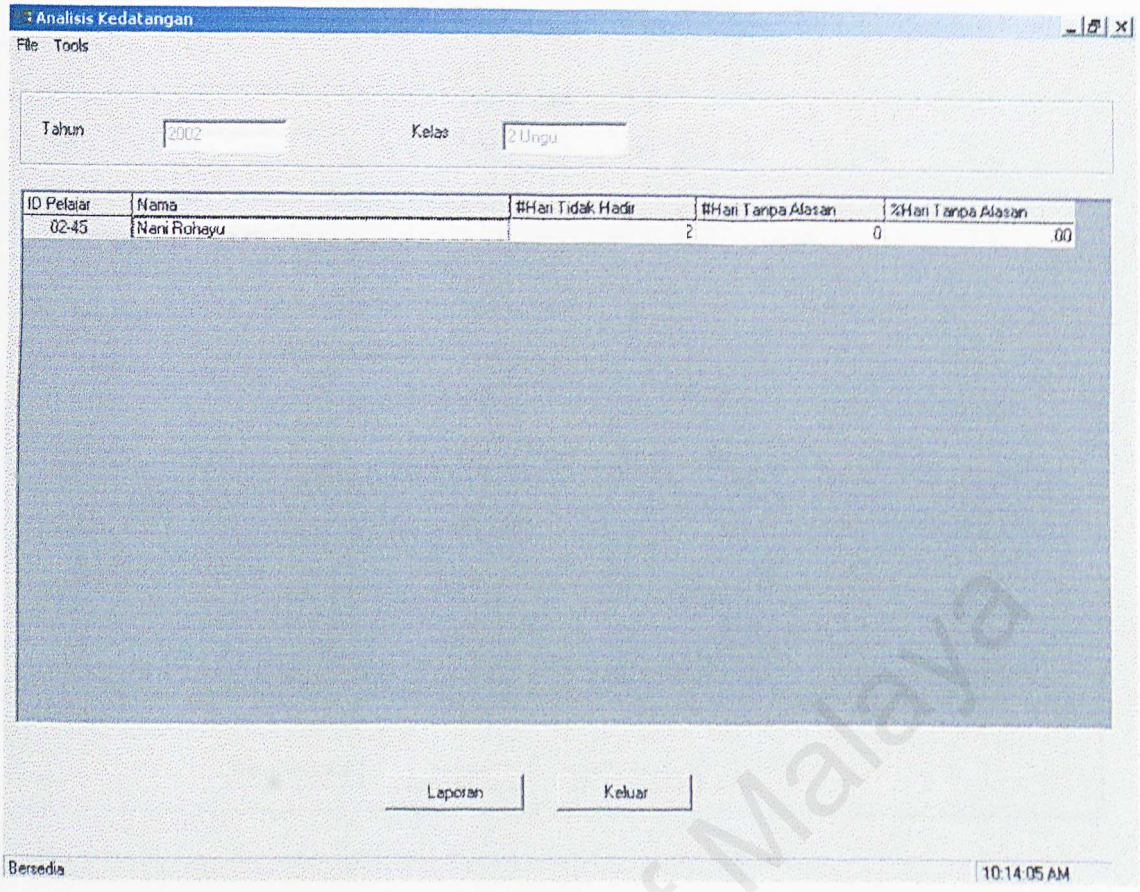


Figure 30. User Manual: Screen of Analysis Attendance

This page will show total days of absent, total days of absent without excuse and percentage of days absent without excuse for each student. To view the report, just click on "Laporan" button. To exit, click on the "Keluar" button.

For tab "Keputusan Peperiksaan", it lets the form teacher to enter their student result.

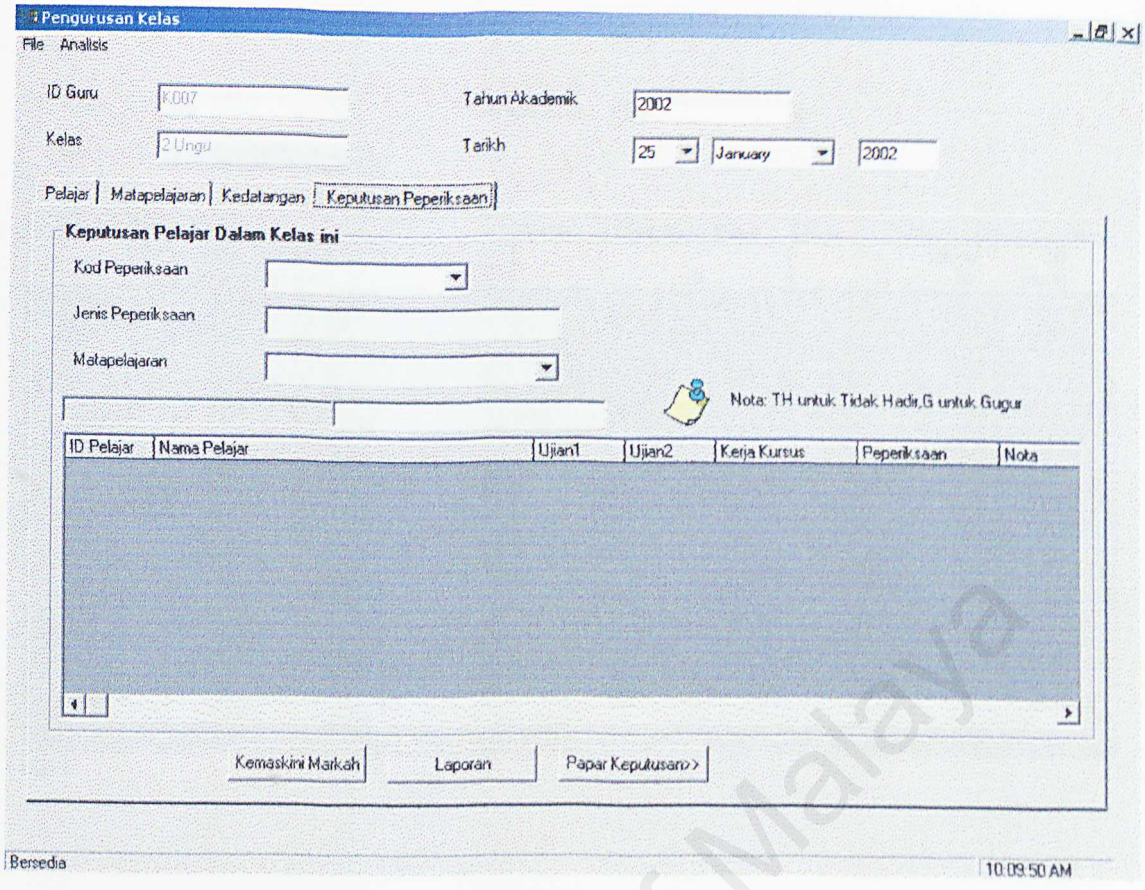


Figure 31. User Manual: Screen of Exam Result in Class Management

Select the "Kod Peperiksaan" (Exam Code) and the "Jenis Peperiksaan" (Type of exam) will be display. Select a subject in "Matapelajaran". The form teacher can entered the mark for "Ujian1" (Test1), "Ujian2"(Test2), "Kerja Kursus" (Assignment), "Peperiksaan" (Exam Mark). The total mark of the exam will be the total of "Ujian1", "Ujian2", "Kerja Kursus" and "Peperiksaan".

If a student did not attend the exam, leave the "Peperiksaan" column blank and entered the "Nota" (Note) column as "TH". If a student drop the subject, leave the "Peperiksaan" column blank and entered the "Nota" column as "G". If there are no "Ujian1", "Ujian2", and "Kerja Kursus", just leave the column as "0" and entered the exam mark in "Peperiksaan" column. After that, just click on the "Kemaskini Markah" button, the total mark and grade for each student will be shown.

Click on the "Laporan" button to view the report. To view the analysis of student result according to class, go to **"Analisis"-> "Analisis mengikut Kelas"**. The screen similar to the below is show.

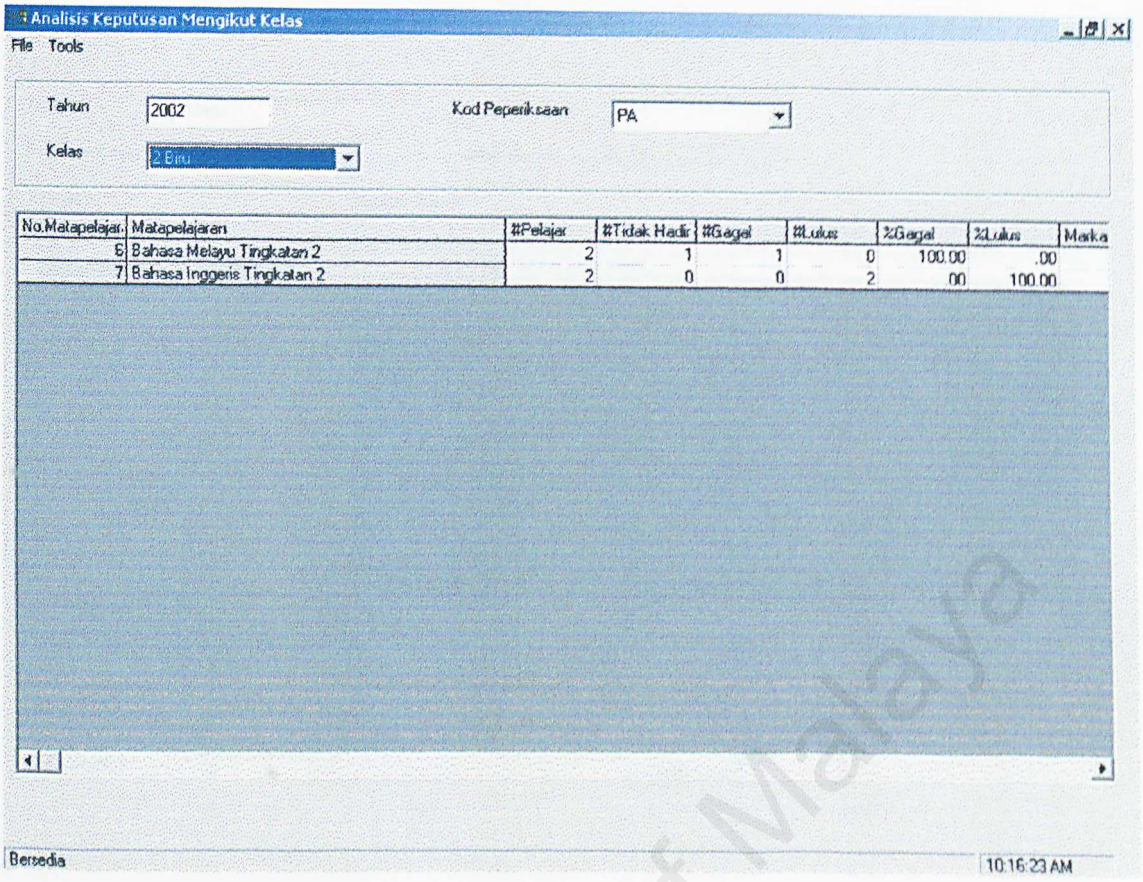


Figure 32. User Manual: Screen of Analysis Result according to Class

Select the exam code, a class, and analysis of student result by subject is shown. Click on the **"Tools"->"Laporan"** to view the report.

Back to the "Keputusan Peperiksaan" tab, click on the "Papar Keputusan>>" button. A screen similar to below is shown.

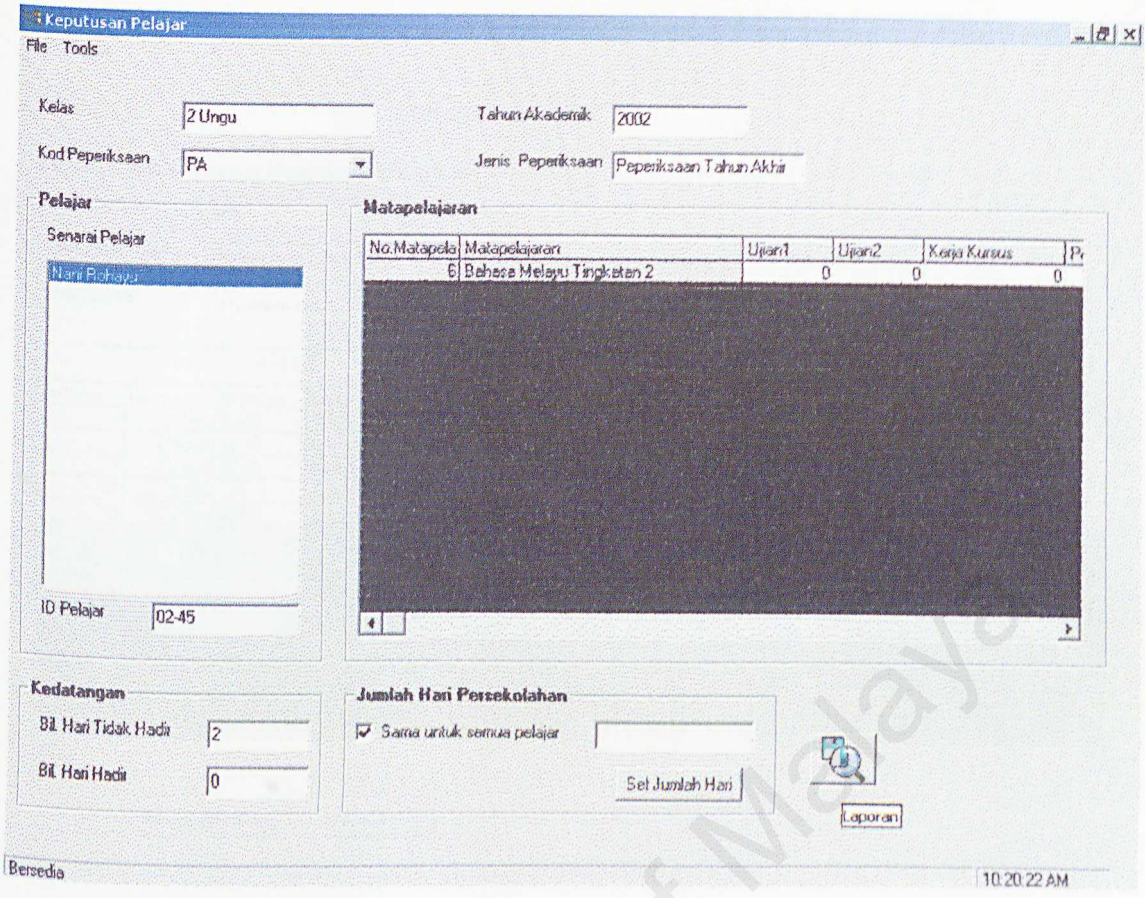


Figure 33. User Manual: Screen of Display Student Result

Select an exam code, select a student to view their result. To calculate the number of day of attendance for all the students in the class, enter the total numbers of days for school in the "Jumlah Hari Persekolahan" and check the check box "Sama untuk semua pelajar", and click "Set Jumlah Hari". If you want to set the total numbers of day of school for one student only, then uncheck the check box "Sama untuk semua pelajar", select the student and click "Set Jumlah Hari".

To print report on the student result, select a student and click on the icon "Laporan" or go to **Tools->Laporan**. To exit from this page, go to **File->Keluar**.

U.10 Subject ("Matapelajaran") Module

This module is for teacher to enter student result for the subject that they (teacher) teach. Therefore, this system provides option to either let the teacher who teaches the subject to enter the result or the form teacher to enter the result. To go to this module, click on the "Matapelajaran" button on the main page or go to Modules->"Matapelajaran".

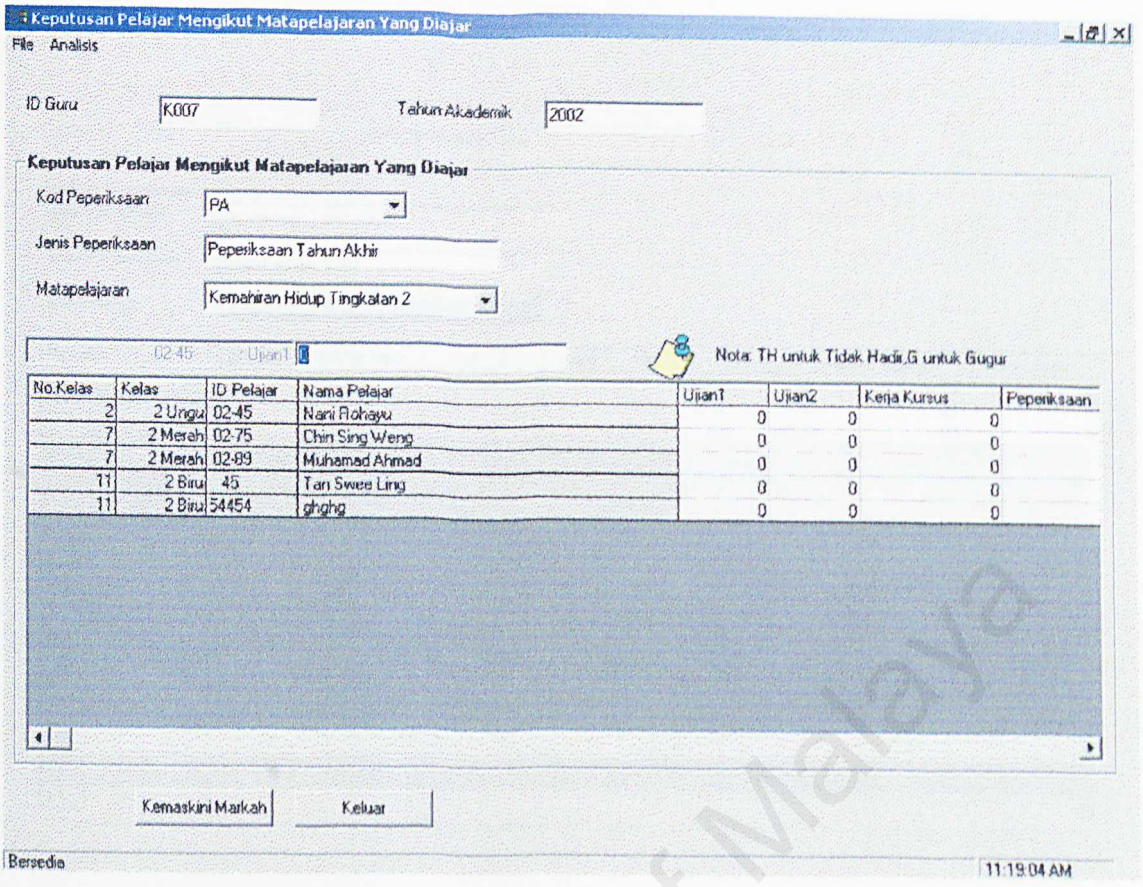


Figure 34. User Manual: Screen of Enter Student Result

Select an exam code ("Kod Peperiksaan") and type of exam will be shown in "Jenis Peperiksaan". In the "Matapelajaran" field, it will be loaded with the subjects teach by the teacher. Just select a subject, list of students who took the subject taught by the teacher will be shown according to their class. The teacher can enter the student result similar to the result entry in the Class Management ("Pengurusan Kelas") module. Click "Kemaskini Markah" button to update the marks.

To view the analysis of student result according to the subject taught by teacher, go to "Analisis" -> "Analisis Keputusan". A page similar to below will appear.

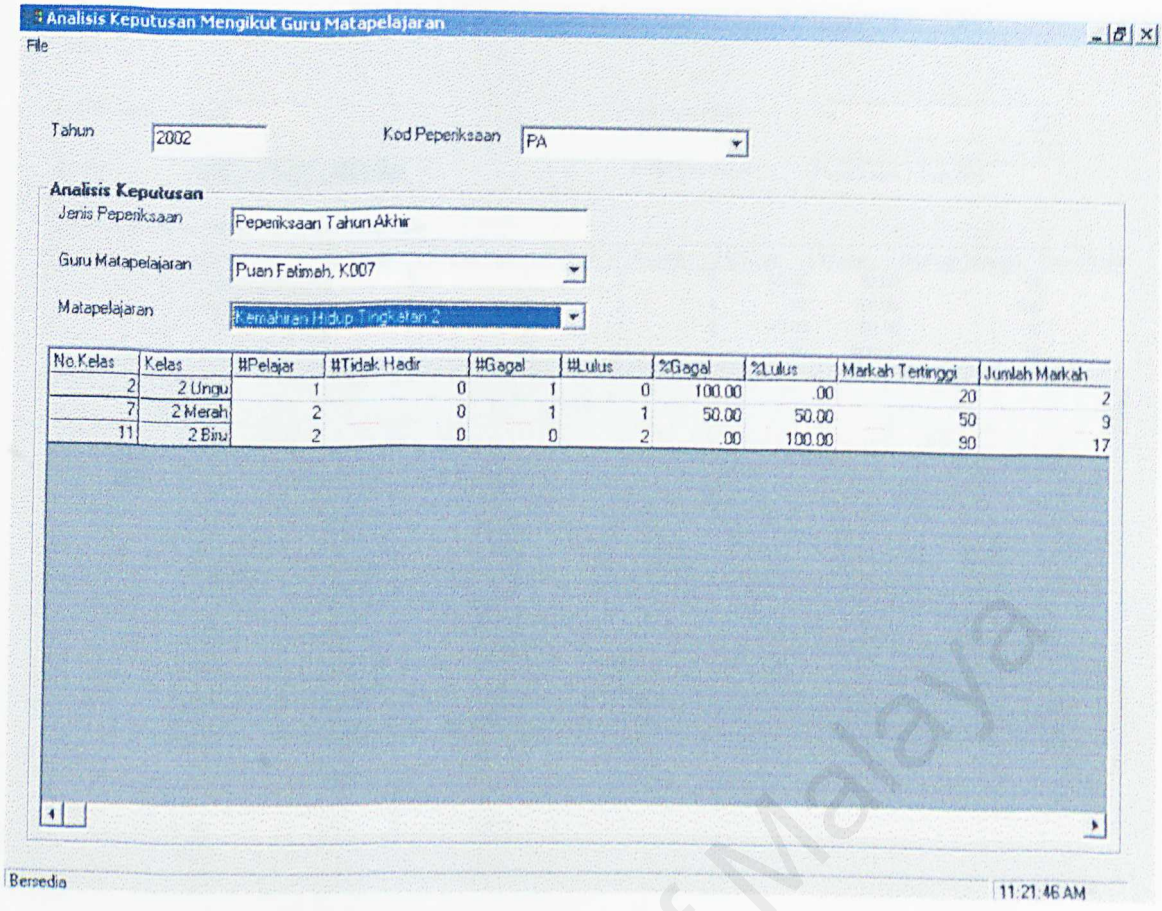


Figure 35. User Manual: Screen of Analysis Result according to subject teacher

From this page, Select an exam code ("Kod Peperiksaan"), select a teacher, and select a subject taught by the teacher. To exit from this page, go to **"File" -> "Keluar".**

U.11 Result Analysis ("Analisis Keputusan")

This module let user to view the analysis of result according to each form.

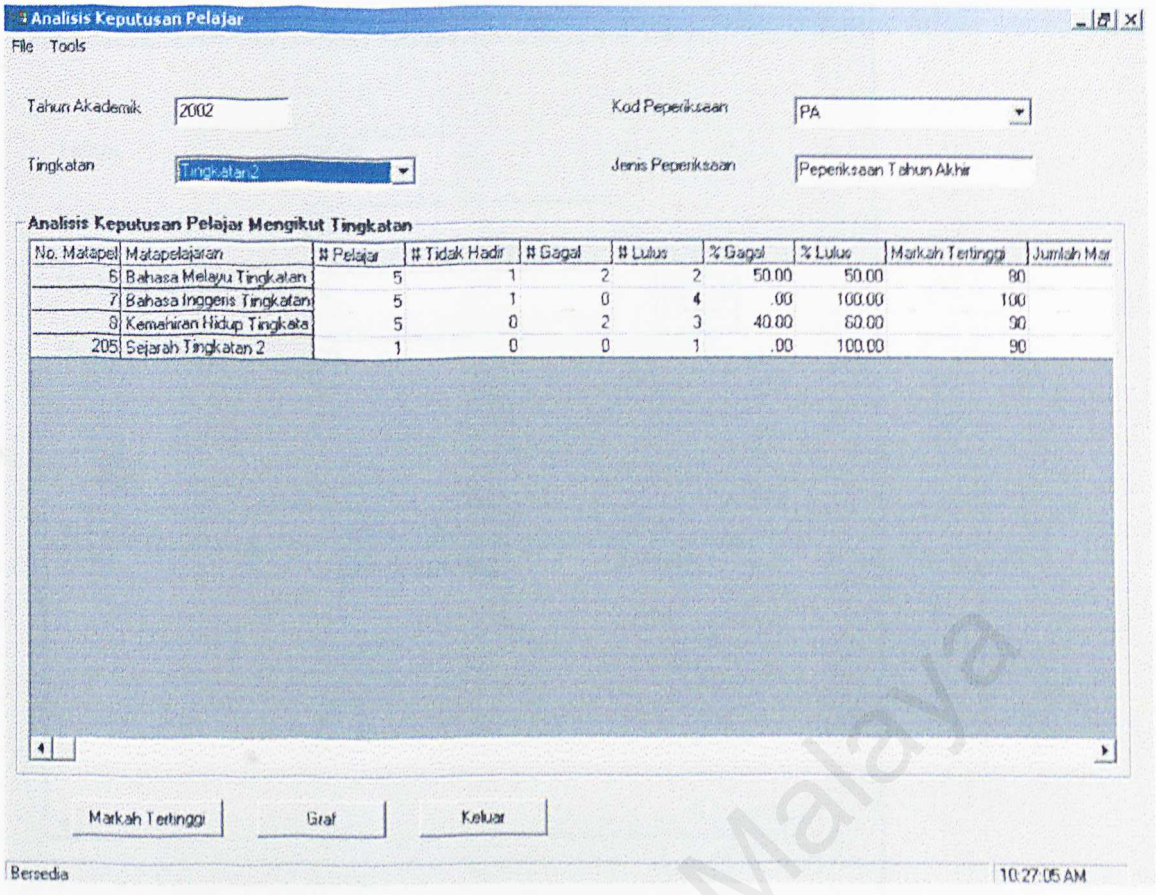


Figure 36. User Manual: Screen of Analysis Result According to Each Form

Select an Exam Code ("Kod Peperiksaan"), select a form at "Tingkatan", and analysis of student result for each subject in that form will be displayed. The analysis will show total student who take the subject ("#Pelajar"), total student absent ("#Tidak Hadir"), total student fail ("#Gagal"), total student pass ("#Lulus"), percentage of student fail ("%Gagal"), percentage of student pass ("#Lulus"), highest mark ("Markah Tertinggi"), total mark ("Jumlah Markah"), and average mark ("Purata"). To preview or print report, go to **"Tools" -> "Laporan"**.

To search for student who obtained the highest mark for a subject, click on the "Markah Tertinggi" button, and a screen shown below will be display.

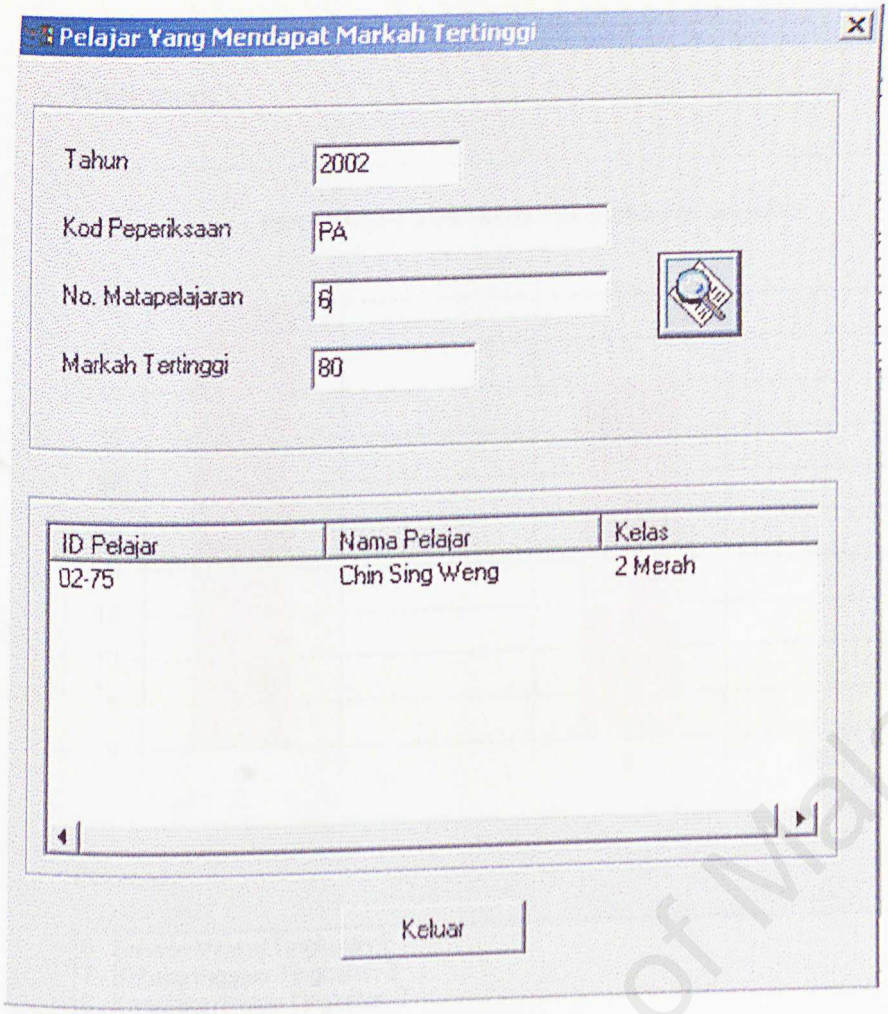


Figure 37. User Manual: Screen of Search Student with Highest Mark

Enter the Exam Code ("Kod Peperiksaan"), Subject Number ("No. Matapelajaran") and press Enter key. It will display the highest mark ("Markah Tertinggi") and student who obtained the highest mark for that subject. Click "Keluar" button to exit.

To display the graph which show the percentage of failure for each subject in that form, click on the "Graf" button. Screen shown below will appeared.

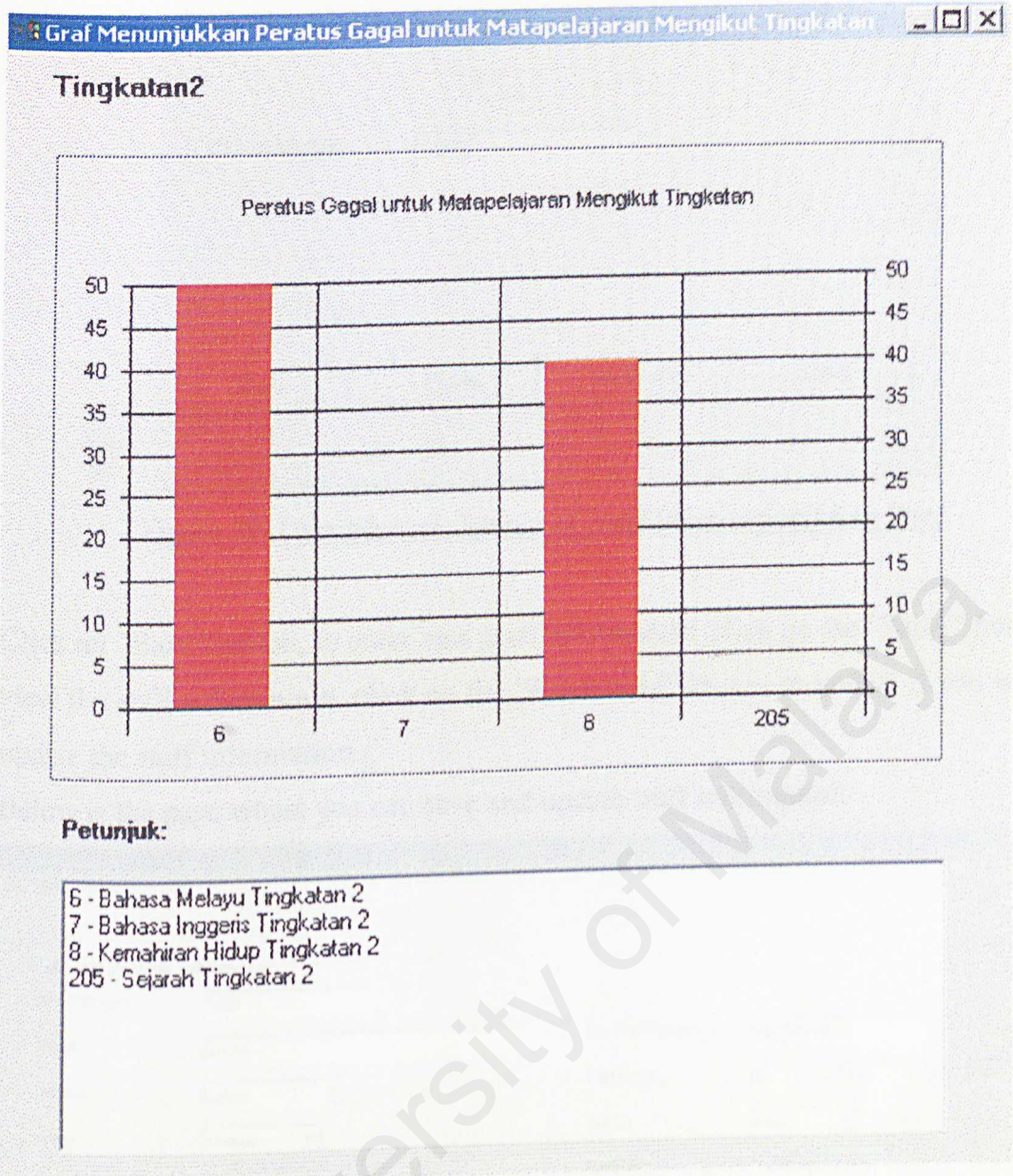


Figure 38. User Manual: Screen of Graph of Analysis Result

To exit the Result Analysis ("Analisis Keputusan") main page, click "Keluar" button.

U.12 Staff Information ("Maklumat Kakitangan")

In this module, only the administrator can add Staff information.

Go to **"Modules" -> "Maklumat Kakitangan"**.

Screen shown below will appeared.

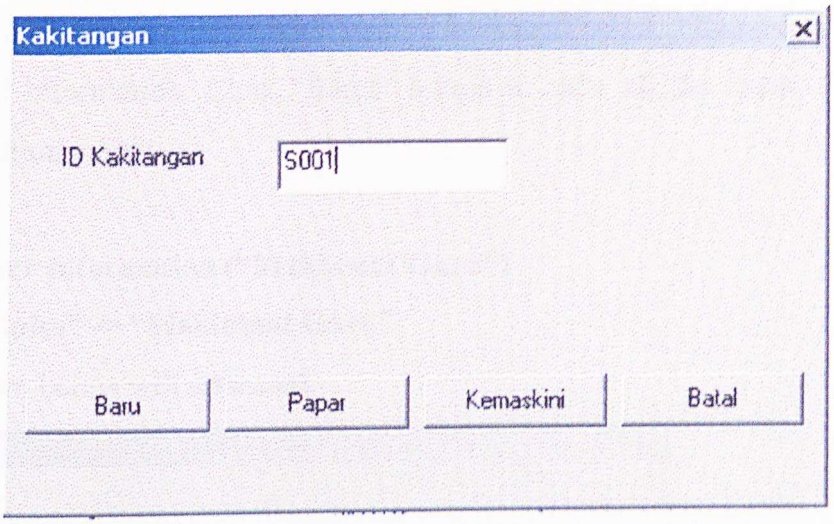


Figure 39. User Manual: Screen of Staff Information Main Page

Click on "Baru" button, to enter new staff information, click on the "Papar" button to view the staff information, click on the "Kemaskini" button to update if you want to update the staff information.

Below is the page where you can save and update staff information.

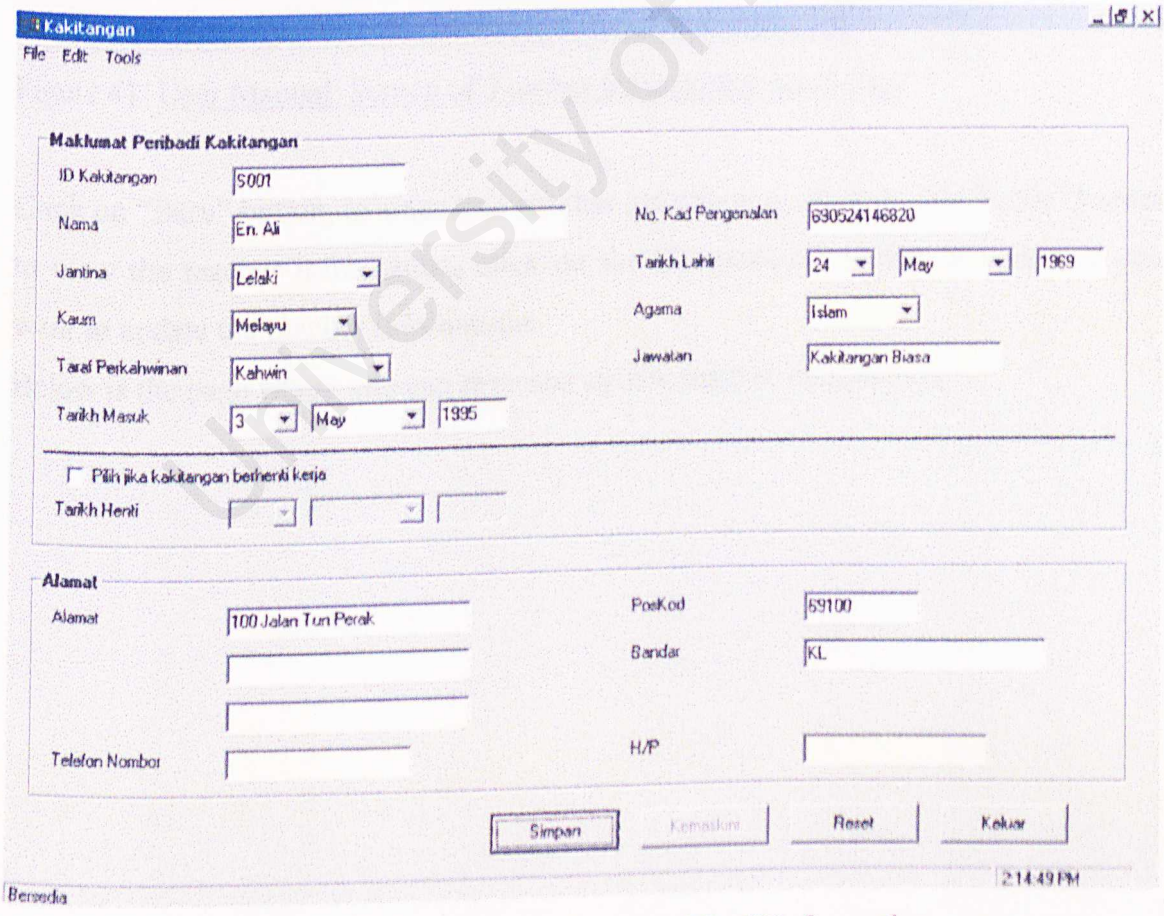


Figure 40. User Manual: Screen of Staff Information

Click "Save" button to save new staff information, click "Kemaskini" button to update staff information. Click "Reset" button to clear all the fields. Click on the "Keluar" button to exit.

U.13 Teacher Information ("Maklumat Guru")

Go to **"Modules" -> "Maklumat Guru"**.

Screen shown below will appeared.

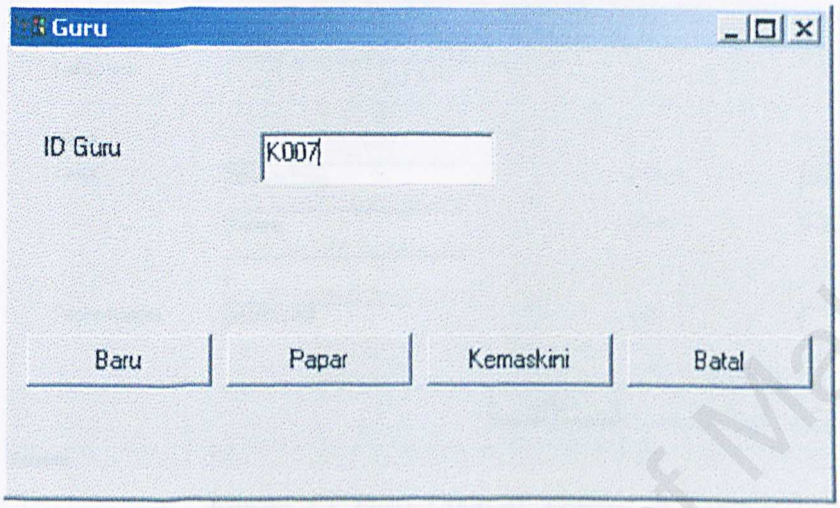


Figure 41. User Manual: Screen of Teacher Information Main Page

Click on "Baru" button, to enter new teacher information, click on the "Papar" button to view the teacher information, click on the "Kemaskini" button to update if you want to update the teacher information.

Below is the page where you can save and update teacher information

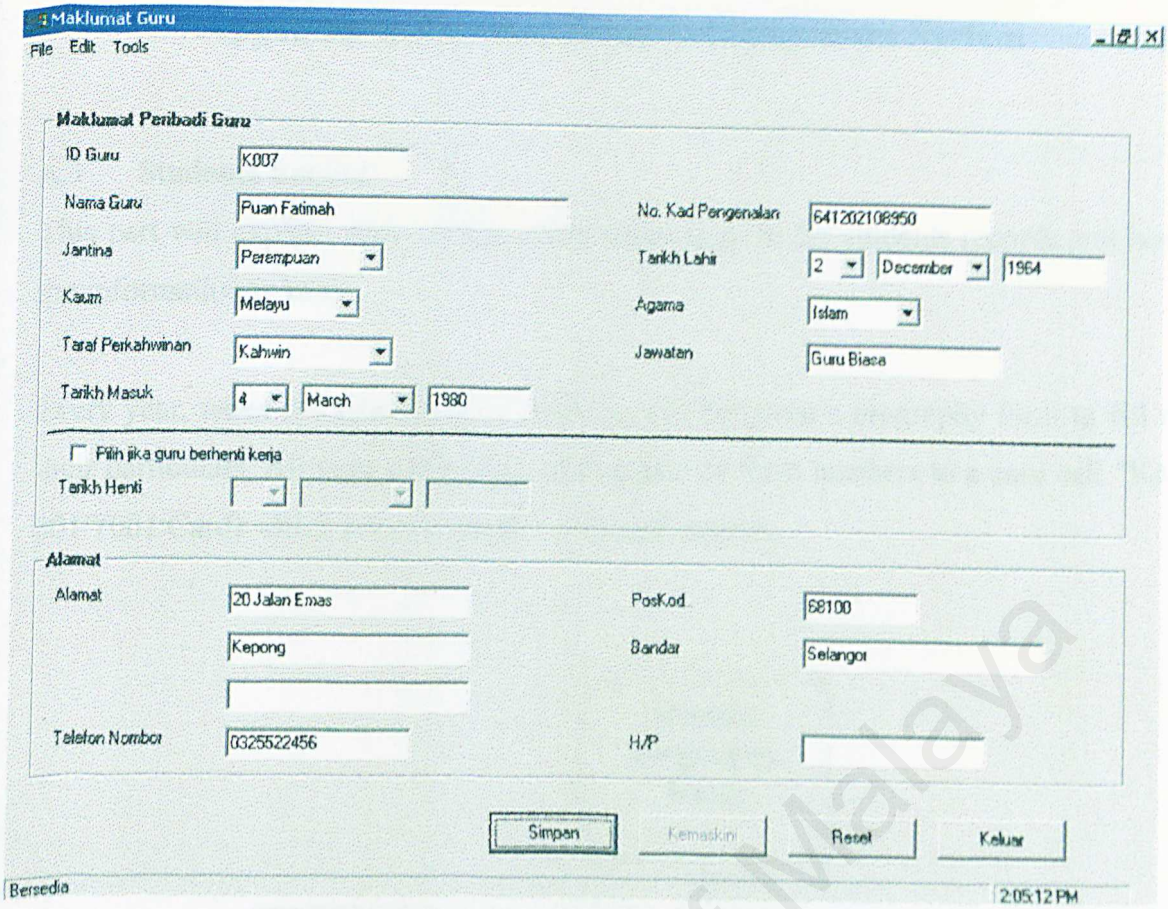


Figure 42. User Manual: Screen of Teacher Information

Click "Save" button to save new teacher information, click "Kemaskini" button to update teacher information. Click "Reset" button to clear all the fields. Click on the "Keluar" button to exit.

Appendix A. Current School Management System

A.1 Students Record

This part will explain what are the information keep in the students records and how the information is keep.

Every year, when the school starts, students will be given a biography form to fill in their particulars. All their particulars will update by form teachers to a card call "Kad 001"(001 Card) which keeps students' personal records.

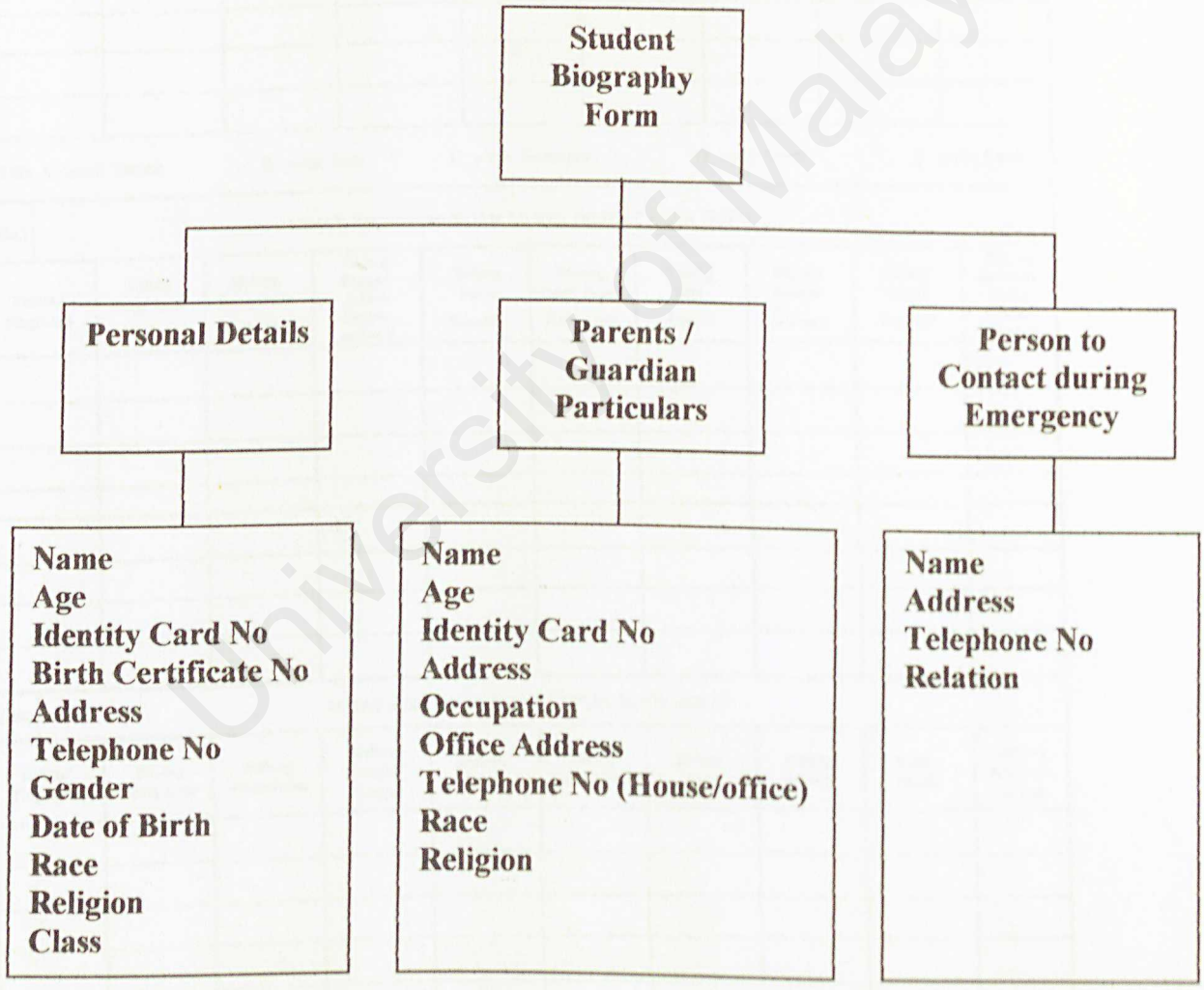


Figure A-1 The information that are requires to fill in the students biography form.

Besides students' particulars, the card also keeps other information (show in Figure A-2 and Figure A-3), which will be fill by the form teachers yearly.

All the cards will be kept according to class and in the cabinet. Old students record will be kept at the back of the school office.

7	REKOD PERIBADI (SULIT)	Muka 3

(I)	SIFAT-SIFAT DIRI

Tarikh / Tingkatan	Kelakuan	Keyakinan *Confidence*	Kerjasama *Cooperation*	Kerajinan *Industry*	Tanggung-jawab *Responsibility*	Dayausaha *Initiative*	Pergaulan *Sociability*	Perawakan *Appearence*

Tulis A untuk Terbaik ; B untuk Baik ; C untuk Sederhana ; D untuk Lemah ; E untuk Buruk.

(IIA)	MINAT KECENDERUNGAN MURID DILIHAT OLEH GURU

Tarikh / Tingkatan	Bidang Kerja Luar *Outdoor*	Bidang Persawatan *Mechanical*	Bidang Penghi-tungan *Compu-tational*	Bidang Sains *Scientific*	Bidang Daya Pujuk *Persuasive*	Bidang Seni *Artistic*	Bidang Sastera *Literary*	Bidang Muzik *Musical*	Bidang Khidmat-Sosial *Social Service*

(IIB)	MINAT KECENDERUNGAN DIPILIH OLEH MURID

Tarikh / Tingkatan	Bidang Kerja Luar	Bidang Persawatan	Bidang Penghi-tungan	Bidang Sains	Bidang Daya Pujuk	Bidang Seni	Bidang Sastera	Bidang Muzik	Bidang Khidmat-Sosial

Di Bahagian (IIA) dan (IIB) untuk tiap-tiap ruangan Minat Kecenderungan
Tulis A untuk Tersuka ; B untuk Sangat Suka ; C untuk Suka ; D untuk Kurang Suka ; E untuk Tidak Suka

FigureA-2 Personal Record (Card "001",page 3)

Figure A-3 Personal Record(Card "002", page 4)

(III) APA SAYA HENDAK JADI BILA SAYA SEORANG DEWASA (Pilihan Pekerjaan di Masa Depan)

Tahun / Tingkatan	Pilihan Pertama	Pilihan Kedua	Pilihan Ketiga

(IV) TEMURAMAH DENGAN MURID

Tarikh Temuramah	Alasan / No. Laporan	Tarikh Temuramah	Alasan / No. Laporan

(V) TEMURAMAH DENGAN IBU BAPA / PENJAGA

Tarikh Temuramah	Alasan / No. Laporan	Tarikh Temuramah	Alasan / No. Laporan

(VI) REKOD KESIHATAN MURID

Tarikh	Umur	Tinggi Meter	Berat Kg.	Kecacatan Anggota	Pendengaran	Penghihatan	Pertuturan

Kod : ✓ - Seperti Biasa ; x - Perlu Perhatian ;

(VII) REKOD PENYAKIT TERUK

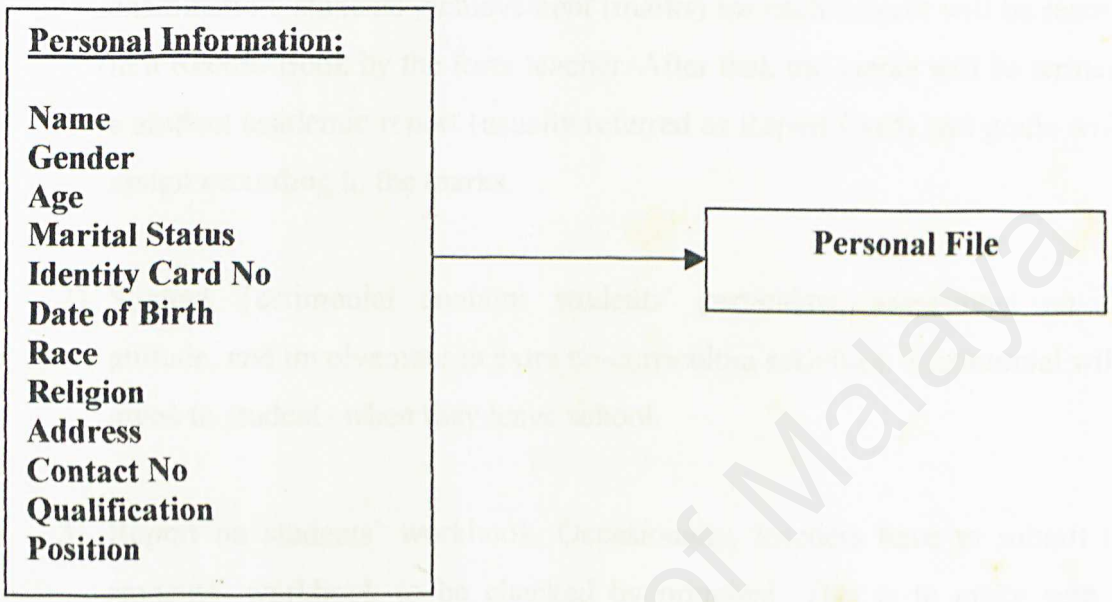
Tarikh	Umur	Penyakit	Tarikh	Umur	Penyakit	Tarikh	Umur	Penyakit

A.2 Teachers and Staffs Record

This part will explain on what are information keep in the teachers and staffs records.

Figure A-4 shows some of the teachers and staff personal detail that keep in the file.

Figure A-4 Teacher and Staff Personal Details



Teachers have two personal files, one file is kept in Jabatan Pendidikan and another file is kept in the school. Each personal file has a unique number. The file will be kept in the cabinet according to the file numbers. Old files will be kept at the back of the office.

A.3 Reports and Letters

This part will review on some of the important reports and letters in the school.

School Reports are intended to keep parents or guardian informed of their children attendance, academic progress and participation in school activities.

A.3.1 Reports

The following are some of the important reports:

1) Academic reports will show results achieved in progress test examinations as well as well as an assessment for effort. There will be a few progress tests, Middle Year Examination and End of Year Examination. After a test or examination, students' achievement (marks) for each subject will be recorded in a Record Book by the form teacher. After that, the marks will be written to a student academic report (usually referred as Report Card) and grade will be assign according to the marks.

2) Student Testimonial contains students' particulars, assessment on their attitude, and involvement in extra co-curriculum activities. Testimonial will be given to students when they leave school.

3) Report on students' workbook. Occasionally, teachers have to submit their students' workbook to be checked by principal. This is to make sure that teachers had consistently checked on the students work and the question set by the teachers is appropriate.

A.3.2 Letters

The following are the letter that will be given to parents to informed them on:

1) Students' extra co-curriculum activities, day and time. (Example, Figure A-5)
2) Students extra class after school hour.
3) Students' discipline (for example, student have been absent for a period of time).

Figure A-5 Letter on student activities

SEK. MEN. KEB. CONVENT BUKIT NANAS,
JALAN BUKIT NANAS,
50250 KUALA LUMPUR. TEL : 03-2380559

Kepada,

.....

.....

..... Tarikh : .....

Tuan / Puan,

Aktiviti Di Luar Waktu Sekolah (Di Dalam Kawasan Sekolah).

Dengan hormatnya dimaklumkan bahawa anak/jagaan tuan .....

Tingkatan ..... diperlukan hadir untuk aktiviti berikut :-

1. Persatuan/Kelab/Rumah Permainan ..... akan mengadakan aktiviti tambahan pada ..... bertempat di ..... dari pukul.....hingga.....
2. Mereka akan ditemani/dilatih oleh.....
   (Jurulatih / Guru Bertugas).
3. Mereka akan berpakaian seragam Sekolah / Persatuan / Kelab / Sukan.
4. Latihan / aktiviti ini adalah bertujuan .....
   .....

Sekian, terima kasih.

" BERKHIDMAT UNTUK NEGARA "

Saya yang menurut perintah,

.....
(PUAN ALICE GEORGE)
Pengetua.

Tandatangan : .....
Nama : .....
(Guru Penasihat / Bertugas)

JAWAPAN IBU BAPA/PENJAGA

Saya..... No. K/P .....
bersetuju / tidak bersetuju anak / jagaan saya menghadiri aktiviti yang tersebut di atas.

Yang benar,

.....

REFERENCE

Books

[1] Ab. Rahim Selamat.(2000). Pengurusan Sekolah Bestari (satu pengenalan) Johor Bharu: Badan Cermerlang.

[2] Chua, Chooi See.(1997). Visual Basic For Beginners: The quick & easy way to create your own Windows applications. Selangor: Federal Publications.

[3] Davis, A.M. (1993). Software Requirements: Objects, functions, and States. New Jersey: Prentice Hall.

[4] Kroenke, D.M.(2000). Database Processing: Fundamentals, Design & Implementation (7th ed.). New Jersey: Prentice-Hall.

[5] Pertroutsis, E. (1998). Mastering Visual Basic 6. San Francisco: SYBEX.

[6] Pfleeger, C.P.(1997). Security in Computing.(2th ed.). New Jersey: Prentice-Hall.

[7] Sellappan.(1997). Visual C++ Through Example: A Reference For Beginners and Developers. Selangor: Federal Publications

[8] Sommerville, I. (2001). Software Engineering (6th ed.). London: Addison-Wesley.

[9] Whitten,J.L. , Bentley, L.D. ,& Dittman, K.C.(2001). System Analysis and Design Method (5th ed.). New York: McGraw-Hill.

Internet

[10] Abang Ahmad Ridzuan & Salawati Talib.(1996). Sikap Pentadbir Terhadap Penggunaan Komputer Dalam Pengurusan Sekolah. Prosiding Seminar Pendidikan Sempena Jubli Emas Maktab Perguruan Batu Lintang (MPBL) 1998. Retrieved July 24, 2001from World Wide Web: http://www2.moe.gov.my/~mpbl/Research/jurnal.htm

[11] Advantages of Visual Basic. Retrieved August 2, 2001 from World Wide Web: http://www.darmstadt.gmd.de/~guenthk/Lava/LavaDoc/html/AdvantVB.htm

[12] An Introduction to the Client /Server Technology- A proven approach to distributed computing. Retrieved July 31, 2001 from World Wide Web: http://compnetworking.about.com/library/weekly/aa050201a.htm

[13] BESTARInet -Sekolah Bestari Malaysia. Retrieved August 1, 2001 from World Wide Web: http://www.moe.edu.my/

[14] Client Servers. Retrieved August 1, 2001 from World Wide Web: http://www.ethoseurope.org/ethos/Techterm.nsf/All/CLIENT+SERVERS

[15] Client/Server Enterprise Computing Solutions: Framework approaches and Technology. Retried August 2, 2001 from World Wide Web: http://oden.csom.umn.edu/idsc6452/Papers/RFakhrazeyev/clientserver.html

[16] Gary Beene's VB Information Center VB History. Retrieved August 2, 2001 from World Wide Web: http://www.vbinformation.com/history.htm

[17] Microsoft SQL Sever-Features. Retrieved August 22, 2001 from World Wide Web: http://www.microsoft.com/sql/evaluation/features/default.asp

[18] MySQL Information. Retrieved July 1, 2001 from World Wide Web: http://www.mysql.com/information

[19] Peter, D.R. (1998). Smart use of IT in the teacher training college and some potential barriers to its implementation. Jurnal Pendidikan 1998, Jilid 6. Retrieved July 24, 2001 from World Wide Web: http://www2.moe.gov.my/~mpbl/Research/Deva.html

[20] PowerSchool. Retrieved July1, 2001 from World Wide Web: http://www.powerschool.com

[21] SchoolSpace[tm] Product Tour. Retrieved July 1, 2001 from World Wide Web: http://www.limitless.net

[22] Schussel, G. Client/Server: Past, Present and Future. Retrieved July 31, 2001 from World Wide Web: http://news.dci.com/geos/dbsejava.htm

CD ROM

[23] MSDN Library-January 2000.