STUDENT HEALTHCARE MANAGEMENT SYSTEM

MEGAT AHMAD MUSTAQIM BIN MEGAT MOHD

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

BORANG PENGESAHAN STATUS TESIS

JUDUL: <u>STUDENT HEALTHCARE MANAGEMENT SYSTEM (SHMS)</u>

SESI PENGAJIAN: <u>2015/2016</u>

Saya <u>MEGAT AHMAD MUSTAQIM BIN MEGAT MOHD</u>

mengaku membenarkan tesis (PSM) ini disimpan di Perpustakaan Fakulti Teknologi Maklumat dan Komunikasi dengan syarat-syarat kegunaan seperti berikut:

1. Tesis adalah hak milik Universiti Teknikal Malaysia Melaka.
2. Perpustakaan Fakulti Teknologi Maklumat dan Komunikasi dibenarkan membuat salinan untuk tujuan pengajian sahaja.
3. Perpustakaan Fakulti Teknologi Maklumat dan Komunikasi dibenarkan membuat salinan tesis ini sebagai bahan pertukaran antara institusi pengajian tinggi.
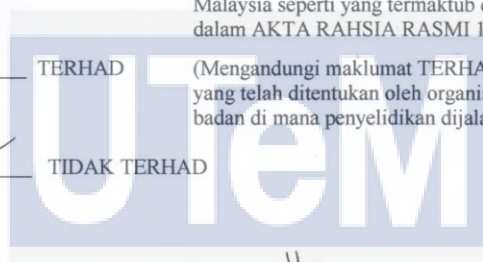4. ** Sila tandakan (/)

_____	SULIT	(Mengandungi maklumat yang berdarjah keselamatan atau kepentingan Malaysia seperti yang termaktub di dalam AKTA RAHSIA RASMI 1972)
	TERHAD	(Mengandungi maklumat TERHAD yang telah ditentukan oleh organisasi/ badan di mana penyelidikan dijalankan)
	TIDAK TERHAD	

(TANDATANGAN PENULIS)   (TANDATANGAN PENYELIA)

Alamat :No 22, Jln Sri Temenggong 4,   En.Yahaya Bin Abd Rahim

Tmn Temenggong Ahmad,

84000,Muar, Johor.

Tarikh: <u>22 OGOS 2016</u>   Tarikh: <u>16 Ogos 2016</u>

CATATAN: * Tesis dimaksudkan sebagai Laporan Akhir Projek Sarjana Muda (PSM)

** Jika tesis ini SULIT atau TERHAD, sila lampirkan surat daripada pihak berkuasa.

STUDENT HEALTHCARE MANAGEMENT SYSTEM

MEGAT AHMAD MUSTAQIM BIN MEGAT MOHD

This report is submitted in partial fulfillment of the requirement for the Bachelor of
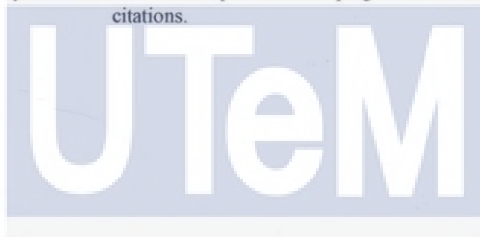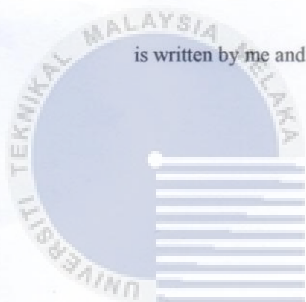Computer Science (Database Management)

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2016

DECLARATION

I hereby declare that this project report entitled

STUDENT HEALTHCARE MANAGEMENT SYSTEM

is written by me and is my own effort and that no part has been plagiarized without
citations.

STUDENT: _____ DATE: 22 Ogos 2016.
(MEGAT AHMAD MUSTAQIM)

SUPERVISOR: _____ DATE: 16 Ogos 2016
(EN YAHAYA ABD RAHIM)

DEDICATION

To my beloved mother who gave me full support, encourage and inspire me during my difficult time to complete this project

To my respective supervisor, Mr. Yahaya Abd Rahim who guide, advice and assist me to develop this project.

To all my friends who always give me the moral support and help me during my difficult time to complete this project and whenever I am in need.

ACKNOWLEDGEMENT

First of all, I would like to thank to my supervisor Mr. YahayaAbd Rahim because of his kindness to accept me as one of his student under his supervision. Special thanks also dedicated to him for all the idea, comment, guidance and giving assistant to complete my PSM project successfully. Also not forgotten, thanks a lot to my beloved mother who have support me and motivate me during completion of my project. Appreciation also goes to my friends that always give me their opinions and help me to complete this project. Lastly, thanks to all that have been involved during the development duration on this project.

ABSTRACT

Student Healthcare Management System (SHMS) is an online system that keep students health information into the system. Other than that, this system also provide site where staff can find the students info by entering some information in searching student info site. Staff can also view the students data and see their frequency of their coming to the clinic. Time also can be save by using this system, students don't need to insert their data anymore because their data has been saved in the system. This system will include the functions of student registration, login, search students information, and view the frequency of sickness for the students. Besides, the functions for search, add, update and delete record and generate report are built in this system to maintain the record in database. Trigger and stored procedure has been implemented directly into the database as a built-in add-on to optimize the system performances. Hence, this system was developed using PhP programming language to create the dynamic webpage in which it is helped by the Apache 2.5 so that the web server can view Student Healthcare Management System's interface on Google Chrome web browser with the data from Oracle 11g database. Some notable advantages of this platform to the online distribution service provider include reduction of cost of paper work, additional new adertisement channels and improved staff-merchants interaction plus efficiency increment in system.

TABLE OF CONTENTS

LIST OF TABLES

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

LIST OF FIGURES

LIST OF ABBREVIATIONS

DBA	**-**	**Database administrator**
DCL	**-**	**Data control language**
DDL	**-**	**Data dictionary language**
ERD	**-**	**Entity relationship diagram**
LAN	**-**	**Local area network**
PSM	**-**	**Projek sarjana muda**
RAM		**Random access memory**
SA	**-**	**System administrator**
SCM	**-**	**Software configuration management**
SDLC	**-**	**Software development life cycle**
SSADM	**-**	**Structures system analysis and design method**
SQL	**-**	**Structure query language**

CHAPTER I

INTRODUCTION

1.1 Project background

Student Healthcare Management System (SHMS) is an online system that keep students health information into the system. Other than that, this system also provide site where staff can find the students info by entering some information in searching student info site. Staff can also view the students data and see their frequency of their coming to the clinic. Time also can be save by using this system, students don't need to insert their data anymore because their data has been saved in the system. This system will include the functions of student registration, login, search students information, and view the frequency of sickness for the students. Besides, the functions for search, add, update and delete record and generate report are built in this system to maintain the record in database. All the functions are built in administration's menu. I used interview and research as the method of solution to find more information about the management of system. The expected output from the implementation of this project is expanded to others location and country.

1.2 Problem statement

i) Problem faced by student

- students need to write down their information on a form that has been prepared by the office. This process may take a long time because of this.

ii) Problem faced by the staff

- if any disaster happen like fire and etc that cause all the students record loss, burned, and damaged. There are no backup data.
- The management of the data is not systematic like student management and all the tools.
- Doctor need to write on a paper what disease did the student suffer and what medicine should the student took.
- All information is difficult to search using the current system. Staff needs to search data from one file to another because it is recorded manually. It will waste their time.

1.3 Objectives

- **i)** To manage the information of student, medicine, diseases and the student's disease.
- **ii)** To ease the doctor and key in data about student disease and the medicine directly into the system.

iii) This system will help to reduce the time for searching. Staff can search and see the frequency of students coming to clinic for particular month.

iv) This system will also provide a better report for staff such as report toview the frequency of disease for each student.

1.4 Project scope

The scopes of the SMHS will be focused on two major points of view which are firstly focused on users and finally focused on system itself. Focused on user is based on the wide range of users that will use this system internally and externally. Meanwhile, the focused on systems are divided into several modules that related with system that will be develop. The scopes are:

1.4.1 Scope of user

There are two main of users will use this system internally and externally. The internal users are students who browse the website for registration process. Besides, the external users are the staff who is act as administrator for this system. The staff will use this system to manage the system and generate report.

1.4.2 Scope of system

i) Registration Module

The objective of this module is to manage the student registration. The student registration is important because they should register before use the system. When the student make registration, they will state their matric number. Then, theirmatric number will be used by staff for processin the system whenever they want to go to clinic.

1.4.3 Project significance

The main purpose for developing this system is to help the office staff to have a better management for their system. Using this system, the data or information are secure because all the information will be save into the database system. Moreover, this system will help the staff to decrease the use of papers and files to keep the data. All the data are inserted will automatically save into the database and it is easier for them for manage it.

1.5Conclusion

As the conclusion, to complete the overall process to develop this system, the cooperation from supervisor and client are needed in order to achieve all the objective listed and solve the problem that had been face by using the current system. This project had accomplish all the project scope and the objectives of the system. Function for searching information also including in project scopes. It will help the staff to search any information about the students. Moreover, it will help the students to quickly make registration process by using this system and no need to fill a form on a paper anymore. Finally, the objective of this project is to give solutions to the problems faced by students and management in the current system.

CHAPTER II

METHODOLOGY

2.1 Introduction

Methods that are used during develop this system is to estimate the time of the system to be delivered on the stage are important. For this Student Healthcare Management System (SHMS) project, waterfall model is applied because by using waterfall model, if there is any problems in any stages, it can detect and refer to stages before and make an error correction for it. Besides, it is easy rather than a correction with same error on the further next stages. In advance, waterfall is simple approach and argue, easily understandable and explainable phases. There are stages in waterfall model, which are Analysis, Design, Implementation, Testing, and Maintenance. Every stage will only start if the stage before have been finished or nearly finish. Thus, Waterfall model is used based on Development Life Cycle (DBLC) as methodology to develop this system.

2.2 Methodology in Developing Database

The current system that are used now does not efficient and effective during the operation. Student Healthcare Management System (SHMS) is as the system that will be used to replaces the current system. There are three main module to be made better which are Registration Module, Medicine Take Module and Student Sick Frequency Module. The Waterfall Model in DBLC starts from Analysis, Design, Implementation, Testing and Maintenance.

The Database Lifecycle (DBLC) contains six phases: database initial study, database design, and implementation and loading, testing and evaluation, operation and maintenance and evaluation.

I. **Database Initial Study**

The purpose of the database initial study is to analyze the company situation, define problems and constraints, define objectives and define scope and boundaries.

a. **Analyze company situation**

The company situation describes the general conditions in which a company operates its organizational structure, and its mission. The database designer need to discover what the company's operational components are, the function and how the interaction are. The design must satisfy the operational demands created by the organization's mission. When the database designer know who controls what and who reports to whom it make them easy to defined required data flow, specific report and query formats, and so on.

b. **Define problems and constraints**

Information can be divided into two categories which are formal and informal. Most of the information are difficult to search. This is because, the current system record the information of data manually that need the staff to search from one file to another file. Through this some of the time will be waste for doing searching. Other than that, the student information are not secure because anyone can read the file.

c. **Define objectives**

The database system that wants to be developed must be designed in order to solve at least the major problems that identified during the problem discovery process. Sometimes sources cannot be discovered. So, the designer will always keep note of the initial study phase where it also contribute to the problem solution. The designer's job is to make sure the database objective always fulfill the end-users expectation.

d. **Define scope and boundaries**

The designer must get to know about the existence of two sets of limits which are known as scope and boundaries. The system's scope will define the extent of the design related to the operational requirement. By knowing the scope, it will help to define the required data structures, the type and numbers of entities, the physical size of the database and so on. The boundaries are known as external to the system. Boundaries also required by existing hardware and software. Preferably, the designer can choose the hardware and software that will best accomplish the system goals.

II. **Database Design**

The second phase focuses on the design of the database model that will support the objectives. In the process of database design, we must concentrate on the data characteristics required to build the database model. In short, we have 2 views of the data within the system: the business view of data as a source of information, and the designer's view of the data structure, its access, and the activities required to transform the data information. Below are the main processes in database design:-

a. **Create the conceptual design**

In this stage, the data modelling will be used to create the abstract database structure, which represents the real world objects in more natural. It also will be easier to understand. Moreover, it also must represent a clear view of the business and its' functional parts. Abstraction level can define where the hardware and the database model not yet identified. The design must be software and hardware independent, where the system can be built within any hardware and software platform which will be chosen by later by the development team.

b. **DBMS Software Selection**

The selection of the DBMS software is very important to the information's system for a smooth operation. Therefore the proposed DBMS software has its own advantages and disadvantages where it must be studied carefully. Other than that, the end users also must be always aware of both DBMS and the database.

c. Create the logical design

Logical design can be define as the major component where it will be used to translate the conceptual design into the internal model for a selected database management system (DBMS) such as phpMyAdmin, Oracle and MariaDB. Furthermore, all the objects in the model will be mapped to a specific constructs that used by the database. The logical design for a relational DBMS is included by tables, trigger, procedure, views and so on.

d. Create the physical design

Physical design can be define as a process of select the data storage and data access characteristics of the chosen database. The characteristics of storage are the types of devices supported by the hardware, type of data access methods supported by the system and the DBMS. The physical design will affect the location of the data in the storage device and also the performance of the system. Other than that, can also say that the physical design described the technical job and more typical of the client or server.

III. Implementation and Loading

In modern relational DBMS such as IBM DB2, Oracle or Microsoft SQL Server, a new database implementation requires the creation of special storage-related constructs to address the end-user tables. After the database has been created, the data must be stored in to the database tables. If the data currently stored are different from the new DBMS requirement, the data must be converted first before loaded. During the implementation and loading phase, it is a must to address performances, security, backup and recovery.

IV. Testing and Evaluation

Once the data have been loaded into the database, the DBA will test and fine tunes the database for performance, integrity, and concurrent access and security constraints. The testing and the evaluation phase using the database tools. If the database implementation is fails to meet the system's evaluation criteria or requirement, several options will be considered to enhance the system such as follows:-

• For performances related issues, the designer must consider fine tuning specific system and DBMS configuration parameters. The best sources of information are the hardware and software technical reference manuals.

- Modify the physical design
- Modify the logical design
- Upgrade or change the DBMS software or the hardware platform.

V. **Operation**

Once the database has been passed the evolution stage, it will consider being operational. At this point the database, management and users will compose a complete information system. The beginning of the operational phase consistently starts the process of the system evolution. When all the targeted end-users entered the operation phase, the problems that could not predict during the testing phase can be detected.

VI. **Maintenance and Evolution**

The database administrator must be prepared to perform routine maintenance activities within the database. Some of the required periodic maintenance activities included such as follows:-

- Preventive maintenance (backup)
- Corrective maintenance (recovery)
- Adaptive maintenance (enhancing performance, adding entities and attributes and so on)
- Assignment of access permission and their maintenance for new and old users
- Improve the efficiency and usefulness of system audits and to monitor system performance
- System security using access level.

2.2.1   Database Development Methodology

The methodology used to implement this database system is Waterfall Development. It is a step-by-step approach to the Database Life Cycle that moves logically from one phase to the next. Below are the phases involved:

a. Planning

The project planning starts in this phase. First, the information is gathered about the current system and also the expected system. Then, the scope objectives and the goals for the proposed system are set up.

b. Analysis

For this project, Database Life Cycle (DBLC) is used as project methodology. In the database initial study phase, the current system has been studied and from that, the business process can be explored. The problem statement of the system can be defined throughout the observation. It will become the objective for the system. The scope can be extracted from the objective to develop the system.

c. Design

Database design is define as the third phase, where a design for the database is form. It can support the Student Healthcare Management System (SHMS) operational and objective, such as phpMyAdmin and Windows is choosing for the database management. The minimum requirement for the installation need to be confirmed first in order for the DBMS in the server to run smoothly. Moreover, the Entity Relationship Diagram (ERD), and data dictionary is create where it will explain the main basic workflow of the system. All relationship between the tables, define the storage structures and the access paths will be known. While business rules that extract from a detailed description will help to create actions within the organizations environment. The business rules defined willproperly describe the entities, attributes, relationship and connectivity and constraints.

d. **Implementation**

During this phase, the database management that has been design will be load and implement. I install the database that has been design for the system. The data will be load to create tables and defined the relationship.

e. Testing

The next phase of DBLC is testing and evaluations. Once the data have been load into the database, the database is test for performance, integrity, and concurrent access and security constraints. Other than that, testing and evaluate the system parallel with application programming is done. After the evaluation stage, it can pass through the operational system. This phase involve the users that will use this system. The testing and evaluation phase occurs in parallel with applications programming. If the database implementation fails to meet the user's requirement, several options will be in order enhancing the system.

f. Evaluation and Maintenance

Maintenance and evolution is the last stage in the methodology and it also life time stage. The system developer will perform routine maintenance to the Student Healthcare Management System (SHMS) which periodic maintenance require doing on the system backup, recovery, enhancing or normal maintenance.

2.3   Requirement of Database System Development

The requirement of Database System Development oversees two smaller requirements. The two elements are software requirement and hardware requirement will be used to fulfil the system requirements.

Name	Function

2.3.1   Software Requirement

There have listed the requirement and specification of software components, which have been used in Student Healthcare Management System (SHMS) .There are:

a) phpMyAdmin	Database system. Use to develop my system.
b) Windows 7	Used in developing system.
c) Microsoft Visio 2013	Uses vector graphics to create diagrams such as data flow diagram (DFD), context diagram, and entity relationship diagram (ERD).
d) Microsoft Project 2010	Tasks schedule and chart.
e) Microsoft Word 2010	Documentations

Table 2.2.1: Software Requirement and Function

2.4 Hardware Requirement

There have listed the requirement and specification of hardware components, which have been used in Student Healthcare Management System (SHMS). There are:

a) Personal computer specification.

- Intel Core 2 Duo Processor and above

- 2GB RAM and above

b) Other accessories.

- Printer – print documentation

- USB Drive – temporary storage

- External Hard Disk – backup all the file and source code

2.5 Conclusion

This chapter is discussing about the literature and project methodology that are using to define the planning of the project. This chapter is covered on introduction of the chapter, domain for the system, existing system and comparison of the existing system, project methodology, project requirement which is including the software requirement, hardware requirement and other requirement, and project schedule and milestone

CHAPTER III

ANALYSIS

3.1 Introduction

Analysis Design is prepared accordance with the requirements of the user needed. It is a process of producing a detailed of the current system and flow of the system that will be develop. This chapter will show how the system is illustrated to make the idea of the to-be-system. This logical data model contains all the needed logical and physical design choices and physical storage parameters needed to generate a design which can be used to create a database. Lastly, data requirement will make the system more clearly about how the form that can be change to database.

3.2 Problem Analysis

Problem analysis is an approach to software requirement analysis. It provides a view of understanding to the problems when developing the project. It produces system improvement objective that address the problems. It includes the task of

studying the problem domain. Analyzing problem and establishing system improvement.

3.2.1 Current System Analysis

For the current Student Healthcare Management System, they are still using the manual system to save data about the students, staff, and the medicines detail. Doctor still need to write down what is the disease of the student that meet him/her and what kind of medicine should be given manually on a piece of paper. Then the receipt will be given to the staff to be save in into the system. For the stock quantity of the medicine, staff will need to check the stock book and check the stock of medicine manually. This process will take some time to be done. Besides that, for the first time students that came to this clinic, they need to fill in a form to register their name. Then, staff will save student's data into the manual file system. This can cause a lot of problems in managing it because it will be difficult to search the data about the student from one file to another. Other than that, the probability data will lost is high because the data about student and the medicine is save in manual file system.

3.3 Developing Database System Analysis

3.3.1 Context Diagram
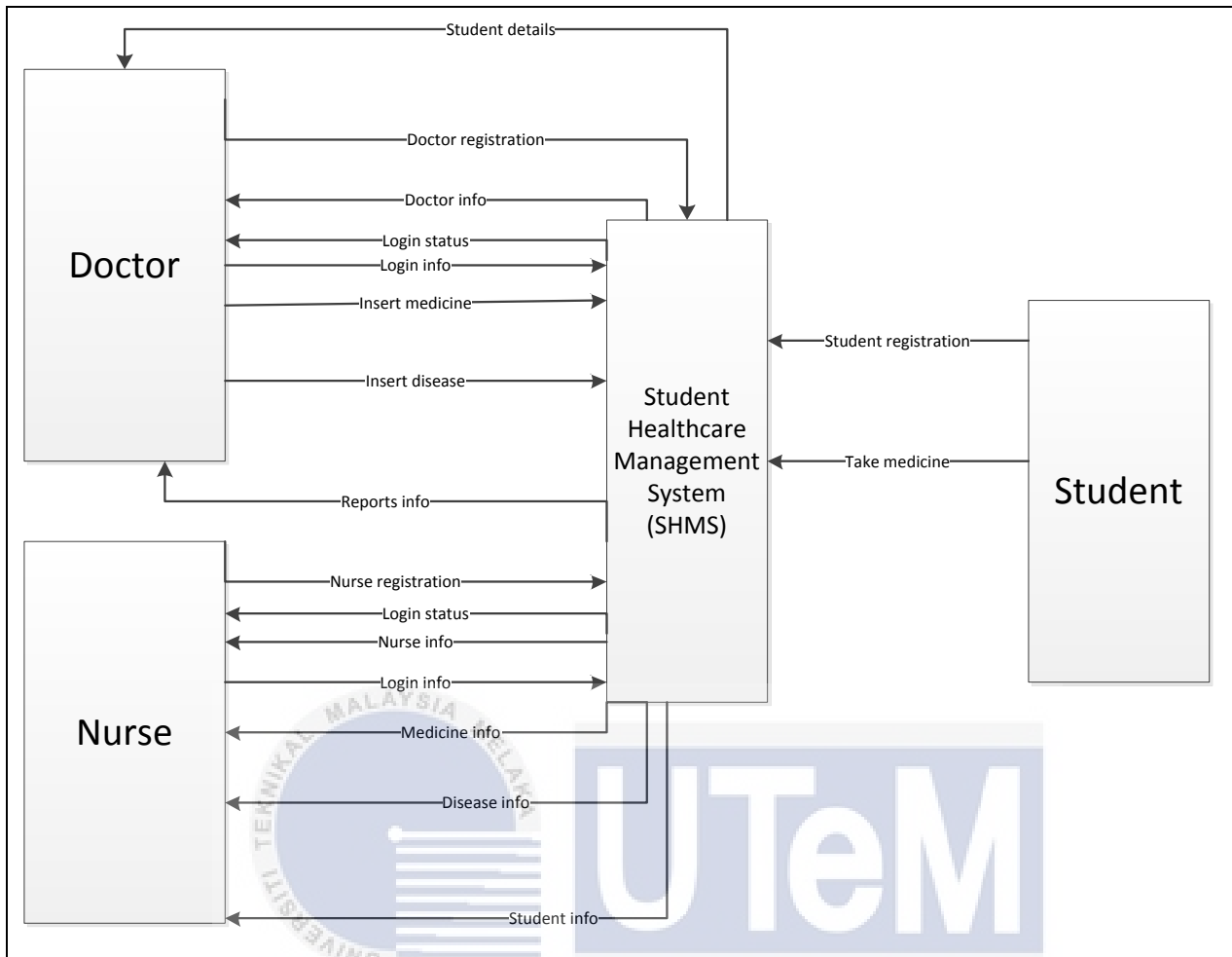
Diagram 3.1 Context diagram

3.3.2 Data Flow Diagram
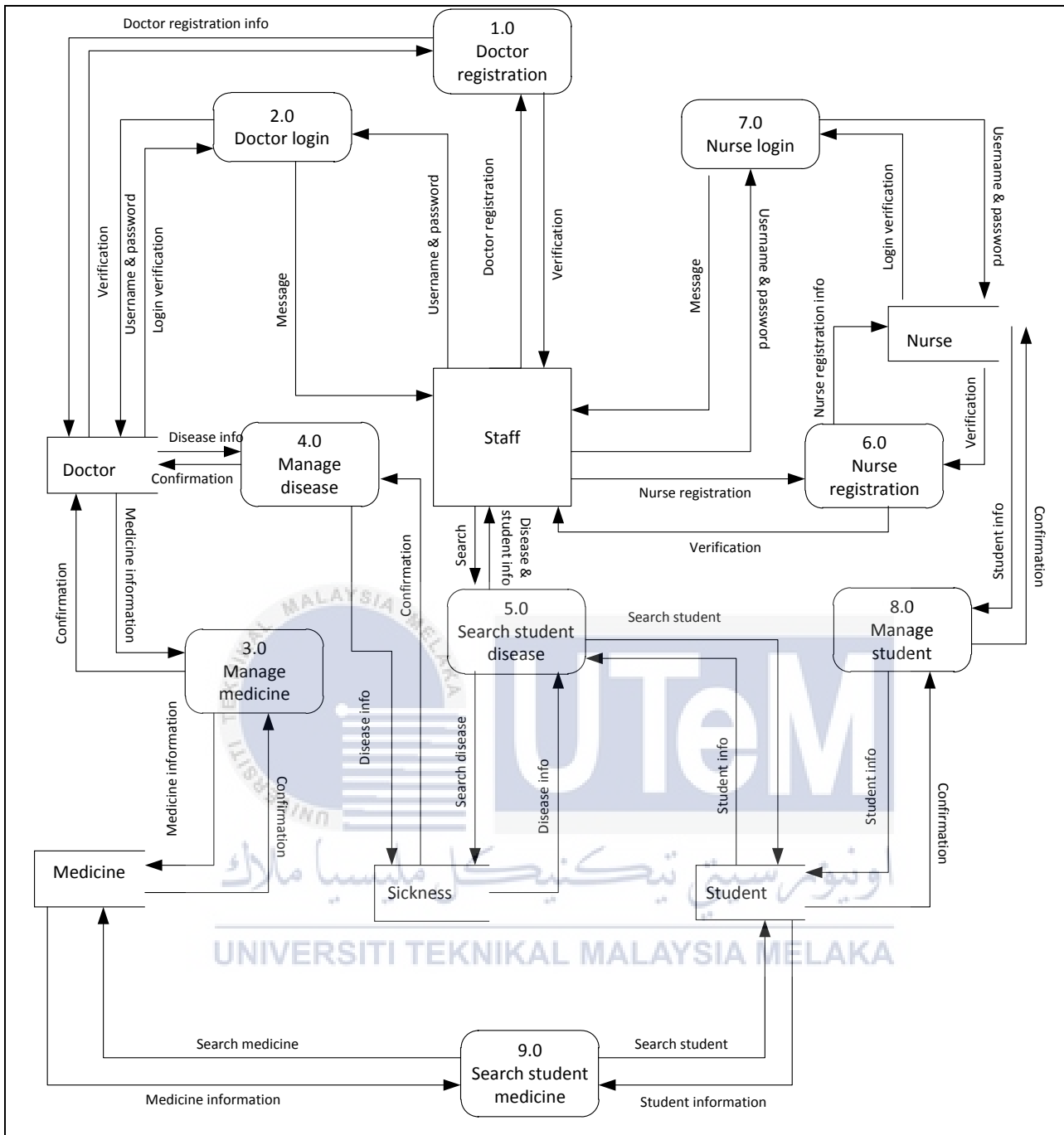
DFD Level 0

Diagram 3.2 : Data flow diagram level 0

DFD Level 1
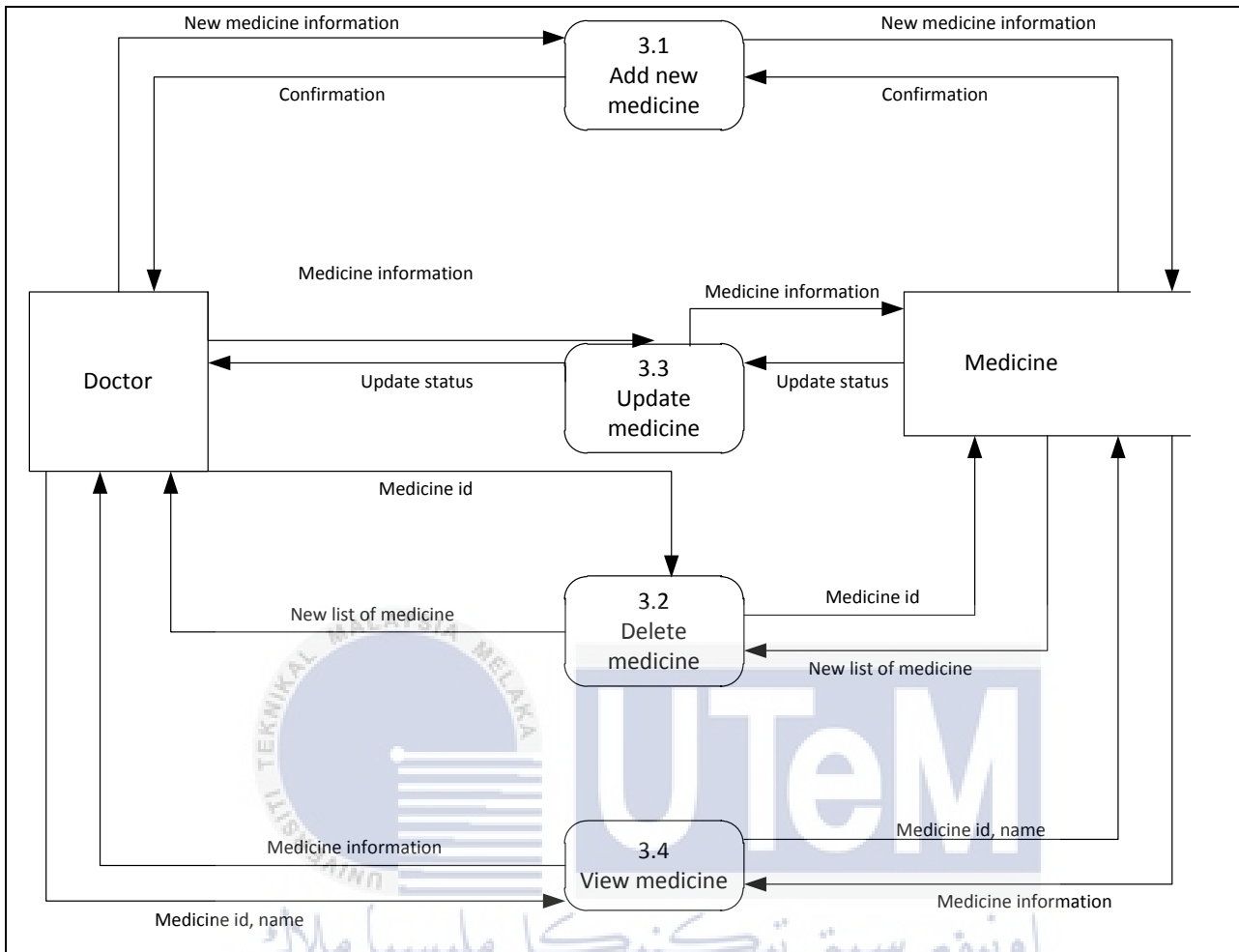
i. **Manage medicine**

Diagram 3.3: Data flow diagram for manage medicine level 1
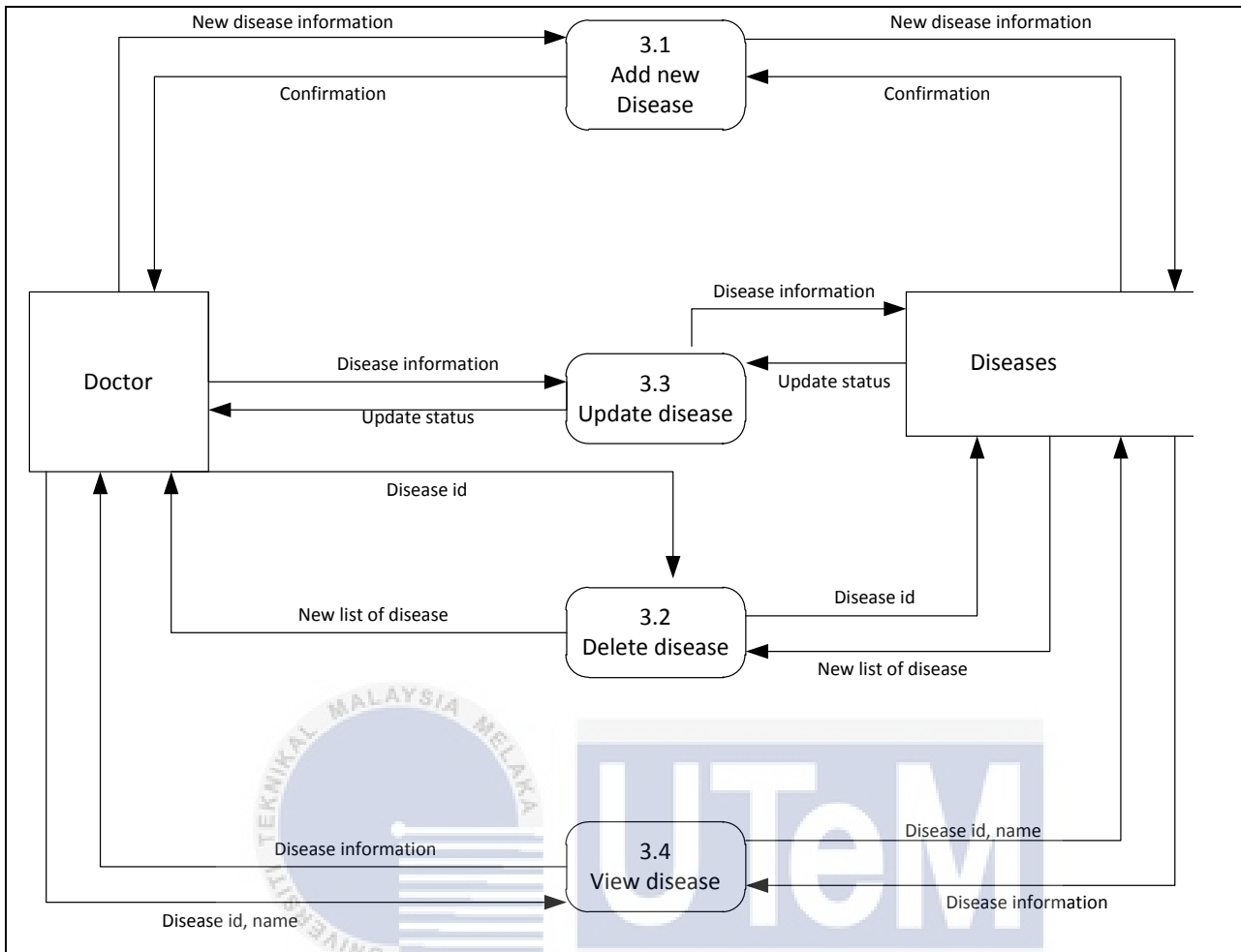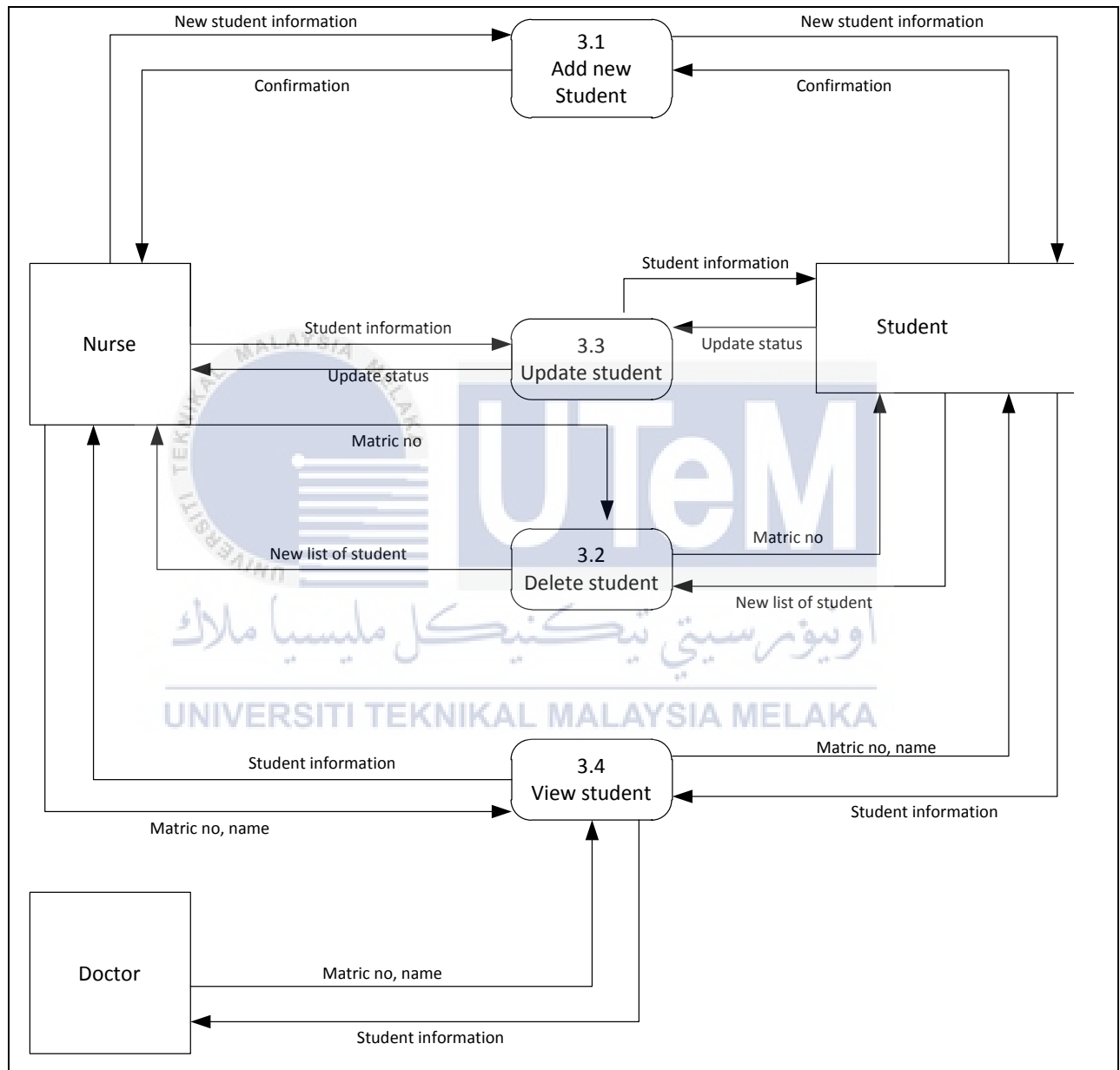
ii. **Manage disease**

Diagram 3.4: Data flow diagram for manage disease level 1

iii.   **Manage student**

Diagram 3.5: Data flow diagram for manage student level 1

3.3.3 Business Rules

1. One staffmanages zero or many medicine

   One medicineis manage by one staff.

2. One staffmanages many student

   One studentmanage by one staff

3. One staffmanage many disease

   One diseasemanaged by one staff

4. One studenttake one or many medicine

   One medicine is taken by one or more student

   Stud_med is the bridge between student and medicine

5. One student has one or more disease

   One diseasesufferby one or more student

   Stud_sick is the bridge between student and sickness

6. One medicine_category has one or many medicine

   One medicine has one or many medicine_category

The data required in the developing database system is identified as below

No	Data Required	Description
1	Student information	The student information is recorded because the staffneeds to refer the student information based on the medicine taken.
2	Staff information	Staff is divided into two user that is nurse and doctor. They can add,update and delete the data in the system.
3	Medicine information	Medicine details is needed to see the description of medicine and give it to student.
4	Disease details	Disease details is needed to keep the details about disease and the disease that the student suffer.
5	Medicine category details	Medicine category details is needed to keep the data about form or type of medicine in the clinic.

3.4 Conclusion

In conclusion,StudentHealthcareManagement System (SHMS) improved the old system that exists by adding some new features to solve the problems faced by user. Some of the features are greatly ease the management. For example, if there any natural disaster happened, the data have some backup in the database. Besides, SHMS provide user likes staff to keep the data within the time period such as date of birth and date of medicine takenby the students. Furthermore, SHMS guide the user for some information needed for accessing the data from database.

CHAPTER IV

DATABASE SYSTEM DESIGN

4.1 Introduction

Database design has three main phases which are conceptual database design, logical database design, and physical database design. Conceptual database design is the first phase in database design methodology. Process of constructing a model of information used in an enterprise, independent of all physical considerations. In this phase, entity types, relationship types, identify and associate attributes with entity or relationship types has been identified. It also determine attribute domains, determine candidate and primary key attributes, and consider use of enhanced modelling concepts. At the same time model for redundancy and validate local conceptual model against user transactions and review local conceptual data model with user has also been checked.

Logical database design is the second phase in the database design methodology. The process of constructing a model of information used in an enterprise based on a specific data model (e.g. relational), but independent of a particular DBMS and other physical considerations. In this phase, relations for local logical data, validate relations using normalization and validate relations against user transactions was derived. At the same time define the integrity constraints and make review local logical data model with user.

While, physical database design is the third and last phase of database design methodology. Process of producing a description of the implementation of the database on secondary storage. Describes the base relations, file organizations, and indexes design used to achieve efficient access to the data, and any associated integrity constraints and security measures. In this phase, Database Designation Language (DBDL) was coded for the system.
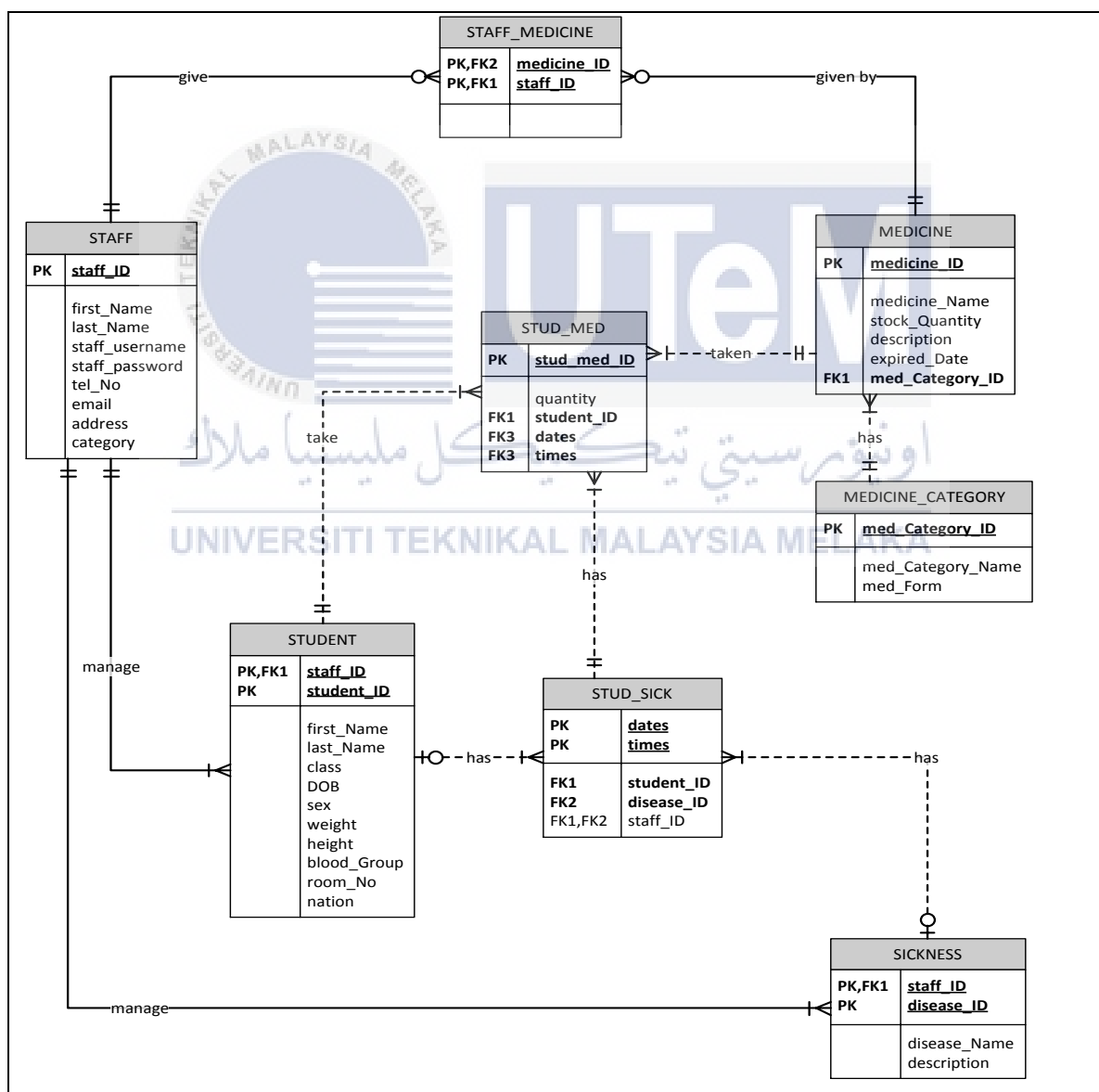
4.2 Conceptual design

Diagram 4.1: ERD

4.3 Logical design

Table Name	Attribute Name	Description	Type / Size	Required	PK/FK	Null/Not Null	Related Table
STAFF	staff_ID	Staff identity number	Varchar2(10)	YES	PK	NOT NULL	
	first_Name	Staff first name	Varchar2(50)	YES		NULL	
	last_Name	Staff last name	Varchar2(50)	YES		NULL	
	staff_username	Staff identity	Varchar2(15)	YES		NULL	
	staff_password	Staff login code	Varchar2(10)	YES		NULL	
	address	Staff address	Varchar2(50)	YES		NULL	
	category	Staff category (Doctor/Nurse)	Varchar2(15)	YES		NULL	
	email	Staff email	Varchar2(50)	YES		NULL	
	tel_No	Staff phone number	Varchar2(12)	YES		NULL	
STUDENT	matric_No	Student identity number	Varchar2(10)	YES	PK	NOT NULL	
	first_Name	Student first name	Varchar2(50)	YES		NULL	
	last_Name	Student last name	Varchar2(50)	YES		NULL	
	Student_class	Student class	Varchar2(10)	YES		NULL	

	sex	Male /female	Varchar2(10)	YES		NULL	
	Blood_group	Student blood group	Varchar2(3)	YES		NULL	
	Room_No	Student's room no	Varchar2(20)	YES		NULL	
	height	Student's height	Number	YES		NULL	
	weight	Student's weight	Number	YES		NULL	
	nation	Student's races	Varchar2(20)	YES		NULL	
	Date_of_birth	Student's birthday	Date	YES		NULL	
	Staff_ID	Staff identity number	Varchar2(10)	YES	FK	NOT NULL	Staff
MEDICINE	medicine_ID	Medicine identity number	Varchar2(10)	YES	PK	NOT NULL	
	Medicine_Name	Medicine name	Varchar2(100)	YES		NULL	
	Stock_Quantity	Stock of medicine	Number	YES		NULL	
	description	Description of medicine	Varchar2(400)	YES		NULL	
	Total_student	Total of student take medicine	Varchar2(10)	YES		NULL	
	Med_category_ID	Medicine category identity number	Varchar2(10)	YES	FK	NOT NULL	Medicine_category
	Total_student	Total of student take medicine	Varchar2(20)	YES		NULL	
	Staff_ID	Staff identity number	Varchar2(10)	YES	FK	NOT NULL	Staff

	Expired_Date	Expired date of medicine	Date	YES		NULL	
SICKNESS	disease_ID	Disease identity number	Varchar2(10)	YES	PK	NOT NULL	
	Disease_Name	Disease name	Varchar2(200)	YES		NULL	
	description	Description of disease	Varchar2(400)	YES		NULL	
	Total_Student	Total student for each disease	Varchar2(10)	YES		NULL	
	Staff_ID	Staff identity number	Varchar2(10)	YES	FK	NOT NULL	Staff
STUD_MED	Stud_med_ID	Stud_med identity number	Varchar2(10)	YES	PK	NOT NULL	
	quantity	Quantity of medicine taken	Number(2,0)	YES		NULL	
	Matric_No	Student's identity number	Varchar2(10)	YES	FK	NOT NULL	Student
	Medicine_ID	Medicine identity number	Varchar2(10)	YES	FK	NOT NULL	Medicine
	times	Time student come to clinic	Varchar2(10)	YES	FK	NULL	Stud_sick
	dates	Date student come to clinic	Date	YES	FK	NOT NULL	Stud_sick
STUD_SICK	dates	Date student come to clinic	Date	YES	PK	NOT NULL	
	times	Time student come to clinic	Varchar2(20)	YES	PK	NOT NULL	
	Matric_No	Student identity number	Varchar2(10)	YES	FK	NOT NULL	Student
	Medicine_ID	Medicine identity number	Varchar2(10)	YES	FK	NOT NULL	Medicine
	Staff_ID	Staff identity number	Varchar2(10)	YES	FK	NOT NULL	Staff

	Prescription	Prescription of disease	Varchar2(30)	YES		NULL	
	Disease_ID	Disease identity number	Varchar2(10)	YES	FK	NOT NULL	Sickness
MEDICINE_CATEGORY	Med_category_ID	Med_category identity number	Varchar2(10)	YES	PK	NOT NULL	
	Med_category_Name	Med_category name	Varchar2(20)	YES		NULL	
	Med_form	Form of medicine	Varchar2(20)	YES		NULL	
	Total_medicine	Total of medicine based on type	Varchar2(20)	YES		NULL	

4.4 Physical design

4.4.1 Create table staff

```
create table staff(
staff_ID varchar2(10)not null  primary key,
first_Name varchar2(50),
last_Name varchar2(50),
tel_No varchar2(12),
email varchar2(50),
staff_username varchar2(15),
staff_password varchar2(10),
address varchar2(50),
category varchar2(15)
);
```

4.4.2 Create table student

```
create table student(
matric_No varchar2(10) not null primary key,
first_Name varchar2(15) ,
last_Name varchar2(15),
student_class varchar2(10),
sex varchar2(10),
blood_Group varchar2(3),
room_No varchar2(10),
height number,
weight number,
```

```
staff_ID varchar2(10),

nation varchar2(20),

date_Of_Birth date,

FOREIGN KEY (staff_ID) references staff(staff_ID));
```

4.4.3 Create table medicine

```
create table medicine (

medicine_ID varchar2(10) not null primary key,

medicine_Name varchar2(100),

stock_Quantity number,

description varchar2(400),

staff_ID varchar2(10),

expired_Date date ,

total_student varchar2(20),

med_category_ID varchar2(10)

FOREIGN KEY (staff_ID) references staff(staff_ID),

FOREIGN KEY (med_category_ID) references
medicine_category(med_category_ID)

);
```

4.4.4 Create table medicine_category

```
create table medicine_category(

med_category_ID varchar2(10) not null primary key,

med_category_name varchar2(20),

med_form varchar2(20),

total_medicine varchar2(20)

);
```

4.4.5 Create table sickness

```
create table sickness(
disease_ID varchar2(10) not null primary key,
disease_Name varchar2(200),
description varchar2(400),
total_student varchar2(20),
staff_ID varchar2(10),
FOREIGN KEY (staff_ID) references staff(staff_ID)
);
```

4.4.6 Create table stud_med

```
create table stud_med(
stud_med_ID varchar2(10) not null primary key,
quantity number(2,0),
matric_No varchar2(10),
medicine_ID varchar2(10) ,
times varchar2(20),
dates date,
FOREIGN KEY (matric_No) references student(matric_No),
FOREIGN KEY (medicine_ID) references medicine(medicine_ID),
FOREIGN KEY (dates,times) references stud_sick(dates,times)
);
```

4.4.7 Create table stud_sick

```
create table stud_sick(
dates date ,
times varchar2(2) ,
prescription varchar2(100),
matric_No varchar2(10),
staff_ID varchar2(10),
disease_ID varchar2(10),
FOREIGN KEY (matric_No) references student(matric_No),
FOREIGN KEY (staff_ID) references staff(staff_ID),
FOREIGN KEY (disease_ID) references sickness(disease_ID),
PRIMARY KEY (dates,times)
);
```

4.4.8 Trigger before insert into table medicine

```
create or replace trigger trig_medicine_ID
before insert on medicine
for each row
declare
s_idmedicine.medicine_ID%TYPE;
begin
select medicine_seq.nextval into s_id from dual;
:new.medicine_ID := 'M' || s_id;
end;
```

4.4.9 Trigger after insert on table medicine

```
create or replace trigger trig_total_medicine

after insert on medicine

for each row

begin

update medicine_category set total_medicine = nvl(total_medicine,0) + 1

where med_category_ID = :new.med_category_ID;

end;
```

4.4.10 Trigger after delete on table stud_med

```
create or replace trigger trig_delete_student

after delete on stud_med

for each row

begin

update medicine set total_student = nvl(total_student,0) - 1

where medicine_ID = :new.medicine_ID;

end;
```

4.5 Conclusion

Based on the on logical, physical and data dictionary that have been provide in this document, the overall of the database structure have been created with the specific module that will be develop in database Oracle. Overall view of the

conceptual design, it shows the relationship for each entity that depending with each other.

CHAPTER V

IMPLEMENTATION

5.1 Introduction

This chapter is about implementation phase which involve the description of all activities that undertaken in Student Healthcare Management System. The purpose of this system implementation is to transform the planned system design in previously chapter into a run able executable file.

It will meet few strategies which are performance, methods, debugging for error, libraries of use to ensure that the Student Healthcare Management System is deliverables based on its standard specifications and requirements during the system implementation.

The implementation of Student Healthcare Management System includes its software development environment setup, software configuration management, database server implementation and the implementation status for each module.

5.2 System Development Environment Setup

It is need to set up the system implementation development environment before the implementation take place. Student Healthcare Management System uses web based three-tier architecture as presented in previous chapter where user can interact with database through application that are based on a web server and displayed via a web browser as show in Figure 5.1. A single computer will consist of all of these components.
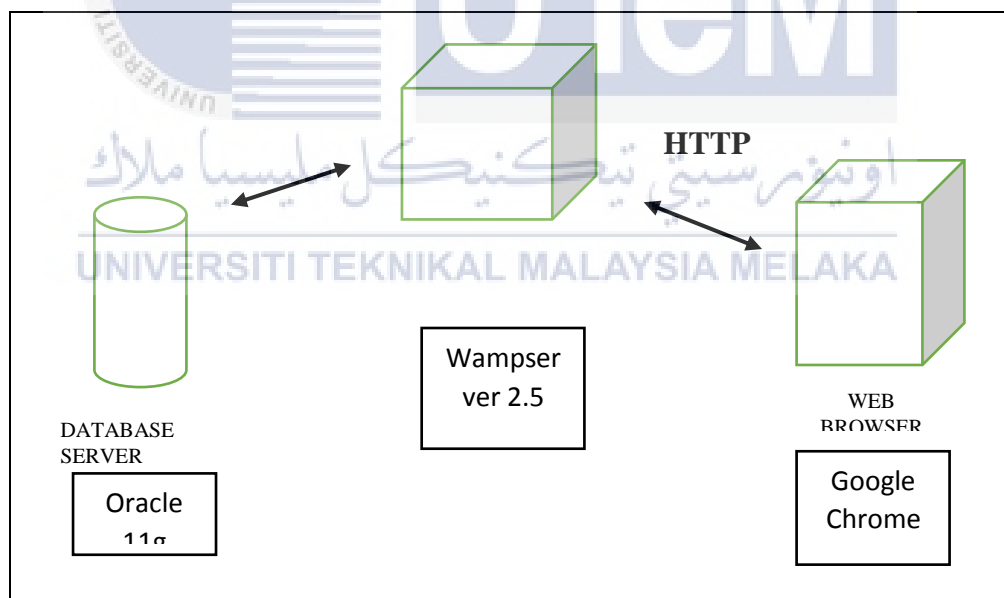
Figure 5.1: Web based, Three-tier client server architecture design of Student
Healthcare Management System

5.2.1 Environment Setup

i. Database

Database which will be used to develop Student Healthcare Management System is stated in table 5.1

Client	Developer
Oracle 11g	Oracle 11g

Table 5.1 : Environment Setup of Database

ii. Software

Software which will be used to develop Student Healthcare Management System is stated in table 5.2

Client	Developer
Web Hosting Server (permanent server)	Wampserver 2.5

Table 5.2 : Environment setup for Server

iii. Browser

Web server which will be used to develop Student Healthcare Management System is stated in table 5.3.

Client	Developer
Google Chrome (version 10 or above)	Google Chrome (version 10 or above)
Mozilla Firefox (version 4.0 or above)	Mozilla Firefox (version 4.0 or above)
Internet Explorer (version 6.0 or above)	Internet Explorer (version 6.0 or above)

Table 5.3: Environment Setup for Web Browser

iv.   Computer Requirement

Hardware which will be used to develop Student Healthcare
Management System is stated in table 5.4

Name	Description
Operating System	Microsoft Window 7 Home Premium 2009 Service Pack 1
RAM	10.0 GB
Processor	Intel® Core(TM) i3-3110M
CPU	2.40Hz

Table 5.4 Environment Setup for Computer Requirements

5.2.2   Installation Step

Student Healthcare Management System uses Microsoft Windows 7 acts as
its platform for development. Hence, all the configuration steps are done based on
Microsoft Windows 7 settings. There is need for the installation of software packages
to develop this system. The following are the software packages to be installed
before starting development of the Student Healthcare Management System:

a. Install and configure Oracle 11g

The installation guide for Oracle 11g DBMS can be referred to the installation
note that comes with the DBMS package. After an installation has been
completed, several objects are created in the Oracle 11g DBMS such as database,
tables, users and other database objects.

b. Install and configure WAMP Server 2.5

The Student Healthcare Management System requires WAMP SERVER 2.5 web
server. After WAMP Server 2.5 has been installed, all of the programming code

(PHP) are written and save in the folders created and within the directory C:\wamp\www.

For each of the software packages that have installed and get ready, some configuration needs to be prior to the interactivity between the software packages. Interactivity between Oracle 11g and WAMP Server 2.5 should be connected before Student Healthcare Management System is deployed. PHP in WAMP Server 2.5 acts as a server-side scripting language use to write the programming language for decorate the interface and connect to the database Oracle 11g to retrieving or display necessary information to the user. While Oracle 11g act as a database or storage where all of the information write and save on Oracle 11g.

5.2.4 Starting the Database Service

When database is managed by Oracle Restart, configure STARTUP options for each individual database service can be managed easily. The management policy for a service turn it to AUTOMATIC in order to service starts automatically when the database is started with SRVCTL.

5.2.5 Database Creation and Database Objects

5.2.5.1 Create Database

Create database command is prior created before database objects can be created. Create database named it as SHMS

```
Create database SHMS;
```

5.3 Software Configuration Management

Software Configuration Management (SCM) is of a software used to manage the design process and if track and control the changes in a software or system. This includes technical aspects of the project, all level of communications, organization, and the control of modifications changes to the project plan by the programmers during the development phase. This means that the management software begin when the system start and will stop or terminate only when the project is out of operation or shutting down. This phase will cover software installation, tool to configure the control application and the version control procedure.
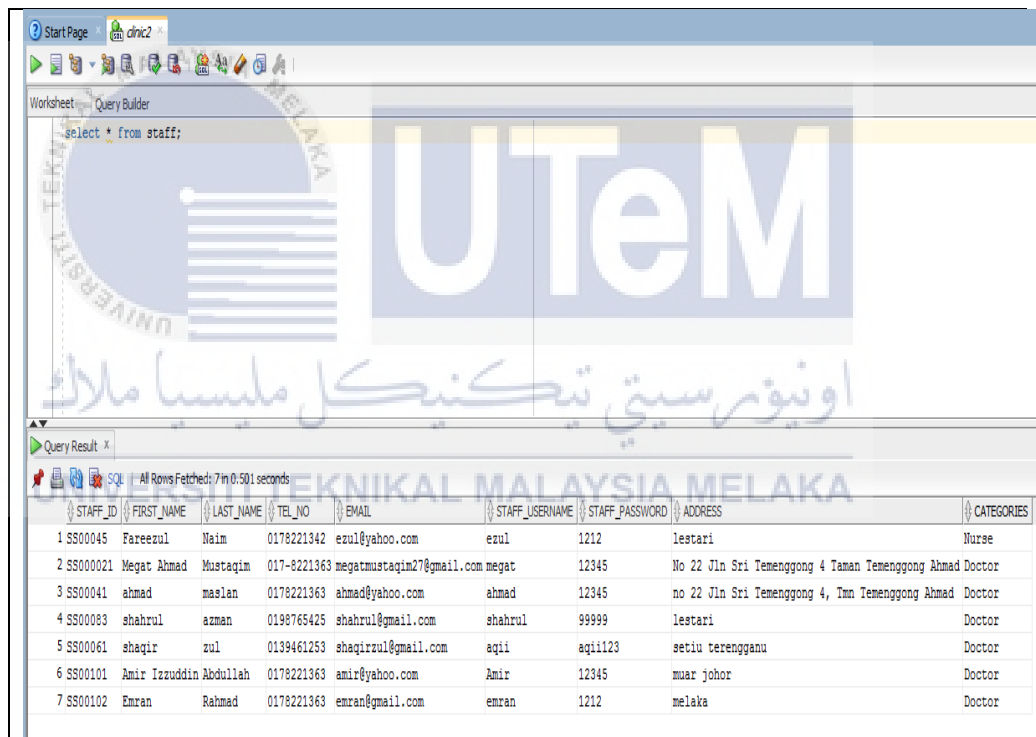
5.4 Implementation Database

This part explains the use of Oracle 11g code through the development of the Student Healthcare Management System. SQL statements are developed to access the data in the database. Few examples are as followed:

i) **SELECT Statement**

The SELECT statement is used to select data from a database. The resultis stored in a result. Figure 5.2 show the output of the query:

Figure 5.2: The output of select statement query

i) Retrieving Selected Columns from a Table

Oracle 11g SELECT statement that accomplishes the desired result:

SELECTstaff_id,first_name,email from staff;

Figure 5.3 show the output of the query:
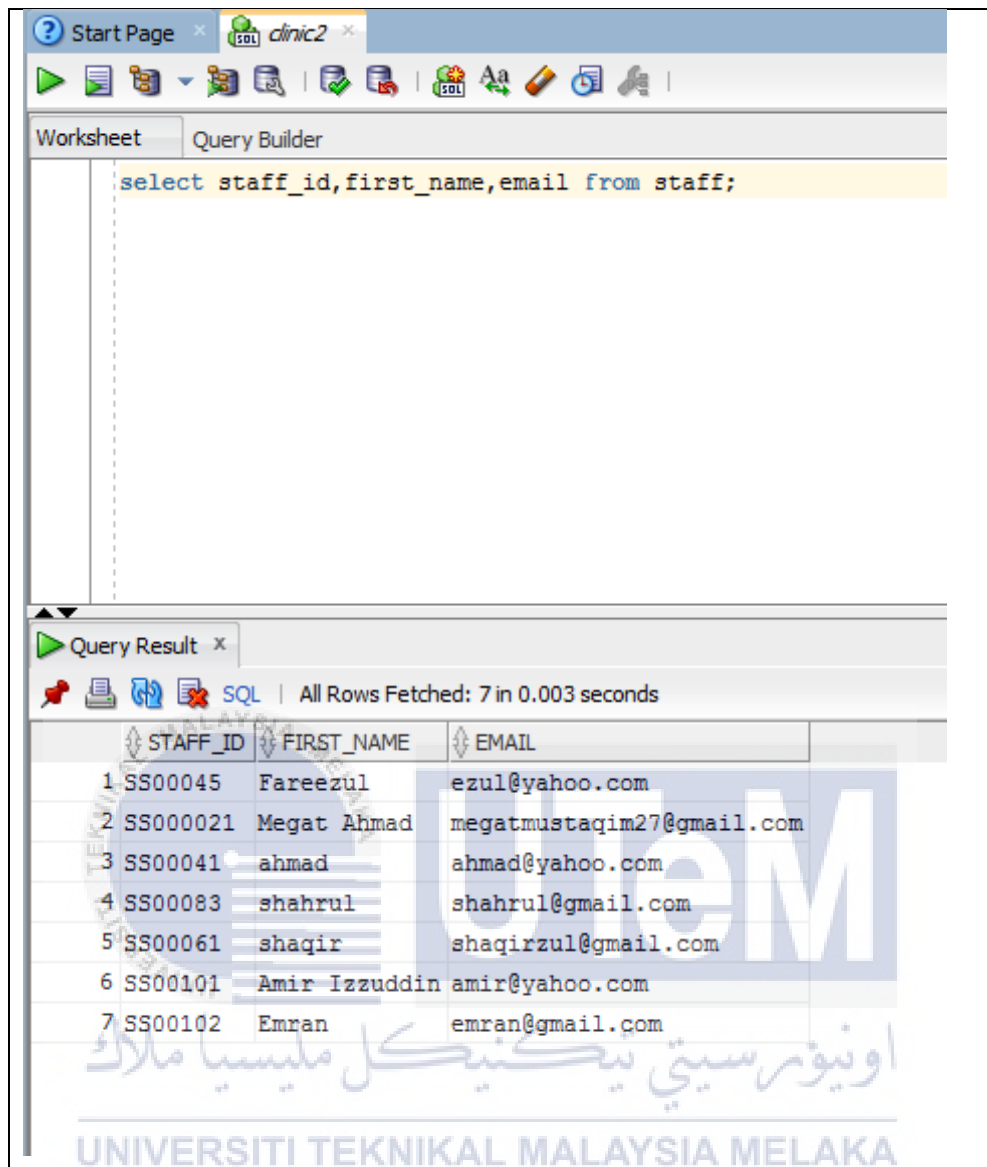
Figure 5.3: The output of retrieving selected columns from a table query

5.5 Implementation Status

The implementation status are made in order to know how much and how far the percentage of the system will be complete. The progress of the development status of the module and component are describes inside the implementation status.

Table 5.5 shows the implementation status for Student Healthcare Management System.

Module/Component	Description	Duration	Status
Login	Authentication of the system access	1 weeks	Completed
Student and Staff Registration	Interface with the coding	1 weeks	Completed
Update Student and Staff Information	Interface with the coding	2 weeks	Completed
Update Medicine Information	Interface with the coding	3 weeks	Completed
Update Disease Information	Interface with the coding	3 weeks	Completed
Manage Report	Interface with the coding	3 weeks	Completed

Table 5.5: Implementation status of Student Healthcare Management System.

5.6   Conclusion

The implementation phase refers to the final process of moving the solution from development status to production status. Student Healthcare Management System implementation includes development tool such as Oracle database and PHP.

Student Healthcare Management System evolves as the system goes on implementing to produce an effective system. The progress of each module is described in implementation status which has successfully implemented. It is considered as the first accomplishment in the implementation of this system.

The next phase is testing. The system is ready to be tested by the user to validate and verify the user requirements and to find any error and debug it. This phase includes unit testing, system testing and integration testing.

CHAPTER VI

TESTING

6.1 Introduction

In this chapter, the testing phase of the Student Healthcare Management System is discussed. System testing is a process to ensure that the system specification is fully developed and fulfills the user requirement. In this phase, a unit test was conducted to test each of the modules and identify the software failure for each function.

Testing activities in the Student Healthcare Management System start with setting up the test plan which describes how the system delivers the testing. It involves in testing process, location or the environment of testing to be carried out and duration of test to be conducted. The next process is the test strategy which contains classes of test, test design consist the testing description which shows the result of testing. Lastly, test result analysis where the result and analysis of testing result had been made and the scale of system evaluated by the developer.

6.2 Test Plan

Test plan is a detailed document stating how to test a system. It describes the overall testing strategy and defines the methodology that will use to conduct an effective test. Test plan is used to test the system to ensure that the system developed meets its specification and requirement including functional and non-functional requirements. Besides that, the test organization, test environment and test schedule will be defined in test plan. It also identifies the hardware, software and tools that required for testing and defines the features and functions that will be tested.

6.2.1 Test Organization

Test organization describes personnel who are involved in testing of the functions and tasks of Student Healthcare Management System. There are personnel who are commonly involved in the testing phase who are software tester, system developer and end users such as staff who registered the student medicine of the diseases. Each person involved in the testing phase will be assign with different responsibilities to test the functional and non-functional requirement of this system. Other than that, error, bugs and defects will be detect and report to be fixed. Table 6.1 describes the test organization information.

No	Personnel Involved	Role and Responsibilities
1	System Developer	Responsible for overall project scheduling functional and non-functional requirement of the system. Person who in charge of unit testing, integration testing and system testing to ensure a successful of the system development.
2	Software Tester	Responsible to detect and track the system error or bug in order to provide feedback to the system

		developer to fixed problem occurred. Person who implements the test strategy and prepares the test scripts to test each of the modules in the system.
3	End User	Test the completed system and signals the approval to implement the system. Person who provides comment toward the system whether it is fulfilled the user requirement or not.

Table 6.1: Responsibilities of Personnel in Testing Process

6.2.2 Test Environment

Student Healthcare Management System consists of test environment set-up which includes hardware, software and supporting tools needed to be carried out. Tester involved is provided with the system manual demonstration before they test the system. The testing environment of this system is on Windows XP Home Premium with Wampserver (server) and SQL Developer (database). Wampserver runs the Student Healthcare Management System by connect it to the database.

Besides that, understanding dependency on external resources can help in exploring the behavior of the system under different stress condition. The environment specification is listed in table 6.2 below.

System Configuration	Specification
Computer operating system	Microsoft Window 7 Home Premium 2009 Service Pack 1
Computer Processor	Intel® Core(TM) i3-3110M
Computer RAM	4.00 GB or above
Computer Hard Disk Space	500 GB or above
Input Device	Mouse, Keyboard
Web Browser	Google Chrome or Internet Explorer

Table 6.2: Test Environment of Student Healthcare Management System

6.2.3 Test Schedule

The test schedule purpose is to define when testing activities will be performed. The schedule will give a guideline to the developer to do the testing on certain time accurately along the duration of project development. Table 6.3 shows the module or component name, activity, duration, start date and end date.

Module/Component	Activity	Duration	Start Date	End Date
System Login	Test unit testing and unit integration	5 day	20/5/2016	25/5/2016
Student Registration, insert new medicine and new diseases	Test unit integration and user acceptance	3 days	20/5/2016	23/5/2016
Update Student	Test unit integration and user acceptance	3 days	23/5/2016	26/5/2016
Update medicine	Test unit integration and user acceptance	4 days	23/5/2016	27/5/2016
Student treatment	Test unit integration and user acceptance	3 days	25/52016	27/5/2016
Manage report	Test unit	3 days	25/5/2016	28/5/2016

	integration and user acceptance			

Table 6.3: Test Schedule for This System Testing Process

6.3 Test Strategy

The test strategy is consists of testing methodologies and testing approach. It is a strategic document used to guide the person who involve in the testing phase about the details of testing process. There are two testing strategy that will be used in this system which are White-Box Testing and Black-Box Testing.

The White-box testing is a test case that is selected based on software structure and implementation. It requires person who has knowledge on programming language to examine the system output and to ensure that the performance of the system works as expected. It is known as a typical of unit and integration test.

The other test strategy that will be used in this system is Black-box testing. This test is focused more on the functional specification which is the tester. It does not require any knowledge on internal design and source code of the system. Each feature and functionality of GUI interfaces is tested without considering how the output is been generated. Table 6.4 shows the test specification for the white box and the black box testing.

White-box testing strategy	Black-box testing strategy
➢ Unit Testing	➢ Functional Testing
➢ Integration Testing	➢ Performance Testing
➢ Security Testing	➢ System Testing
	➢ Usability Testing

Table 6.4: The test specification for the white box and the black box testing.

6.3.1 Classes of tests

There are several test classes that will be implemented to the system which includes of unit testing, integration testing, user acceptance testing and security testing.

i. Unit testing

    Each module of Student Healthcare Management System will be tested whether the functionality of this system is working fine and the unit of coding will be checked to ensure the correctness.

ii. Integration testing

    After unit testing, integration testing of the system is done by the system developer before system testing start in the Software Development Life Cycle. This testing will test all modules after integration together to verify and checking is it working properly and designs correctly.

iii. System testing

    There are two type of system testing in Student Healthcare Management System which is Functional Testing and Performance Testing. The functional testing will test the implementation of the business requirement and need, and the performance testing involves non-functional requirement of the system.

iv. Security testing

    Security testing is very important for the Student Healthcare Management System to protect data and maintains functionality as intended. The security concepts that need to be applied by security testing are availability, integrity, authentication, confidentially and authorization. This security test will be conducted by system developer.

v. User acceptance testing

This testing is the formal testing conducted by end user to determine whether the Student Healthcare Management System satisfies its acceptance criteria. This test is designed to determine whether the system is fit for user or not.

6.4 Test Design

Test Design is divided into two things that are test description and test data. The test description will cover the activities that are required and it is will be documented. It will describe test cases and the expected result. While for test data it will cover about the user acceptance.

6.4.1 Test Description

The test case identification, test case and expected result for each module of Student Healthcare Management System will be designed and documented in the test description. In this section, staff login component is detailed in the table form as shown in Table 6.5 to Table 6.9 below.

Test Case ID	TEST_001-Staff login		
Testing Phase	Unit testing		
Description	To ensure that only the authorized staff can access into the system successfully.		
Test No	**Action**	**Expected result**	**Respondent comment**
SHMS_01	All fields	System will display	Message'Please fill out this field'

	leave blank	reminder to enter the username and password	willappearwhen the user does not fill in the field.
SHMS _02	Username field leave blank	System will display reminder to enter the username	Message 'Please fill out this field' will appear
SHMS _03	Password field leave blank	System will display reminder to enter password	Message 'Please fill out this field' will appear
SHMS _04	Enter wrong username and password	System will display error message of incorrect username or password	Message 'Wrong username or password' will pop up
SHMS _05	Enter correct username and password	System will display staff homepage	Message 'Login success, Welcome ……..' will be appear

Table 6.5: Test case for staff login component

Test Case ID	TEST_002-Student registration		
Testing Phase	Unit testing		
Description	To insert new student into the system		
Test No	Action	Expected result	Respondent comment
SHMS _06	All fields leave blank	System will display reminder to enter the value for each field	Message'Please fill out this field' willappearwhen the user does not fill in the field.
SHMS _07	Unfilled certain field	System will display error warning	Message 'Please fill out this field' will appear
SHMS _08	Valid input for each field	New student successfully stored in database	Successfully make student registration

Table 6.6: Test case for student registration component

Test Case ID	TEST_003-Insert new medicine and diseases		
Testing Phase	Unit testing		
Description	To ensure that new medicine and disease is successfully insert into the database		
Test No	Action	Expected result	Respondent comment
SHMS _09	All fields leave blank	System will display reminder to enter the value for each field	Message'Please fill out this field' willappearwhen the user does not fill in the field.
SHMS _10	Unfilled certain field	System will display error warning	Message 'Please fill out this field' will appear
SHMS _11	Valid input for each field	New medicine/disease successfully stored in database	Successfully insert new medicine/disease

Table 6.7: Test case for insert new medicine and disease component

Test Case ID	TEST_004-Update student/medicine		
Testing Phase	Unit testing		
Description	To ensure that student and medicine information can be update		
Test No	**Action**	**Expected result**	**Respondent comment**
SHMS _12	All fields leave blank	System will display reminder to enter the value for each field	Message'Please fill out this field' willappearwhen the user does not fill in the field.
SHMS _13	Unfilled certain field	System will display error warning	Message 'Please fill out this field' will appear
SHMS _14	Valid input for each field	Student/medicine information updated	Successfully update student/ medicine information

Table 6.8: Test case for updating medicine and student component

Test Case ID	TEST_005-Manage student treatment		
Testing Phase	Unit testing		
Description	To ensure that student can get the medicine based on the sickness		
Test No	**Action**	**Expected result**	**Respondent comment**
SHMS _15	All fields leave blank	System will display reminder to enter the value for each field	Message'Please fill out this field' willappearwhen the user does not fill in the field.
SHMS _16	Unfilled certain field	System will display error warning	Message 'Please fill out this field' will appear
SHMS _17	Valid input for each field	Student and medicine required successfully stored into database	Successfully give student the medicine required.

Table 6.9: Test case for medicine given to student component

Test Case ID	TEST_006-Integration module		
Testing Phase	Integration testing		
Description	To test all modules can run smoothly without error after integration together		
Test No	**Action**	**Expected result**	**Respondent comment**
Integration_01	Test all modules in order it can run smoothly after all modules integrated	Functionality of each module can be performed as specified	

Table 6.10: integration module

Test Case ID	TEST_007-Integration system		
Testing Phase	System testing		
Description	To ensure Student Healthcare Management System can function efficiency and meets its functional and non-functional requirement		
Test No	**Action**	**Expected result**	**Respondent comment**
SYSTEST_01	Whole system's functionalities of the system is tested which include GUI interface	This system is error free and able to function well.	

Table 6.11: integrated system

6.4.2 Test Data

Test data created is representing the real data processing conditions of the test transaction. It is important that the test for each module of Student Healthcare Management System be covered. The test data of staff login components are detailed in table 6.10 to table 6.15 below.

Module	Attribute	Test data
Staff login (Doctor)	Username	Megat
	Password	Abc123
Staff login(Nurse)	Username	Ezul
	Password	Abc456

Table 6.12: Test data for staff login component

Module	Attribute	Test data
Student registration	Matric No.	B031310390
	First name	ShahrulNizam
	Last name	Azman
	Class	BITD S1G2
	Date of birthday	08/27/1994
	Sex	Male
	Weight	50 (in kg)
	Height	130 (in cm)
	Blood group	A (Dropdown menu)
	Race	Malay(Dropdown menu)

Table 6.13: Test data for student registration component

Module	Attribute	Test data
Insert new medicine	Medicine ID	*Auto increment*
	Medicine name	Panadol
	Stock quantity	100
	Medicine form/types	Pills or tablets
	Description	Relieving mild to moderate pain such as headache, migraine, nerve pain (neuralgia), toothache, sore throat, earache, period pain and rheumatic and muscular aches and pains
	Expired date of medicine	09/20/2020

Table 6.14: Test data for insert new medicine component

Module	Attribute	Test data
Insert new diseases	Disease ID	*Auto increment*
	Disease category	Fever
	Description	An abnormally high body temperature, usually accompanied by shivering, headache, and in severe instances, delirium.

Table 6.15: Test data for insert new disease component

Module	Attribute	Test data
Update student information	Class	BITS S1G2
	Height	150 (in cm)
	Weight	51 (in kg)

Table 6.16: Test data for update student information component

Module	Attribute	Test data
Update medicine information	Medicine name	Antibiotic
	Description	Antibiotics, also called antibacterials, are a type of antimicrobial drug used in the treatment and prevention of bacterial infections
	Stock quantity	100

Table 6.17: Test data for update medicine information component

Module	Attribute	Test data
Student treatment	Student matric No	B031310390
	Category of disease	Cold, flu, infectious diseases
	Description	Hot fever
	Date	*sysdate*

Table 6.18: Test data for student treatment component

6.5 Test Results and Analysis

The test result is the identification of the Student Healthcare Management System based on the system function testing. During the testing phase, any error and bugs occurred is recorded so that it can be fixed. The test case will be conducted again to make sure that it meets the user requirement and error free. After that, this system will be delivered by the user. Each function that has been tested and the result of the test cases is shown in table 6.16 below.

Test case ID	Module	Software problem submitted	Result (Ok/Failed)
TEST_001	Staff login	None	Ok
TEST_002	Student registration	None	Ok
TEST_003	Insert new medicine and diseases	None	Ok
TEST_004	Update student and medicine info	None	Ok
TEST_005	Manage student treatment	None	Ok
TEST_006	Integration module	None	Ok
TEST_007	Integration system	None	Ok
TEST_008	View diseases info	None	Ok
TEST_009	View student based on disease for certain month	None	Ok
TEST_0011	System function validation	None	Ok

Table 6.19: Test summary result

The overall test case result of Student Healthcare Management System is satisfied by intended user with the scale of 4 over 5 because this system still has something that need to be improve and will be discuss in the next chapter.

6.6 Conclusion

In conclusion, this chapter has discussed about the test plan and how the testing is done and result that has been record and reported. This is the critical stages during the development of system. This phase is conducted in order to identify the problem of the system and to ensure that it is well function and meet the user requirement. The activity will be carried out in the final chapter and will be discuss more about the project conclusion which will listed out all of the strength and the weakness of Student Healthcare Management System. Besides that, some improvement will be suggested in the next chapter and these proposals will be useful for the future upgrading purpose.

CHAPTER VII

CONCLUSION

7.1 Introduction

In this chapter, the strengths and the weaknesses of Student Healthcare Management System will be discussed. Besides that, some suggestions has been listed in order to overcome the weakness of the system that is faced by the user. The completion of this project has met the scope and the criteria that has been proposed before start developed this system.

7.2 Observation on Strengths and Weaknesses

Student Healthcare Management System has its own strengths and the weaknesses. The observation of its detail will be described in the following section.

7.2.1 Strengths

Student Healthcare Management System can be an effective and helpful system for staff in the clinic to manage the student, medicine and others. The strengths of the system is listed as below:-

1. Enable staff to register new student, new medicine and added new diseases using system.
2. To ease the staff to search student data and student history
3. To ease the staff to view the statistic of student come to clinic and do the reporting process.
4. Enable staff to view and manage medicine information.

7.2.2 Weakness

Student Healthcare Management System has some weaknesses that need to be improved in the further testing. During testing stage, this system has meet a lot of challenges. Some of the weaknesses is listed as bellow:-

1. Doctor can't view the student that they has treated.
2. Student can't login into the system and can't book for the treatment with the doctor.
3. This system is not really user friendly because user need to see the user manual in before use this system.

7.3Proposition for improvement.

In order to be a useful and more user friendly system, Student Healthcare Management System will need to make some improvement. Below are the proposition for improvement for the weaknesses that has been mention before.

1. Doctor should be allow to view all the student and also the student that has he/she treated in order to follow up the student health status.
2. Allow the student to login to the system so that they can book or make an appointment with the doctor on certain day.
3. Use the interface that is easily to understand by the new user.

7.4 Contribution

TheStudent Healthcare Management System contributes to:

- Ease the doctor to manage medicine information and the stock of medicine.
- Ease the staff to view statistic of student disease in the clinic for each month
- Ease the staff to manage registration module for student using this system.
- This system can be accessed anywhere by doctor or nurse.

7.5 Conclusion

In conclusion, Student Healthcare Management System has achieved the main objectives and the scope that has been proposed by the developer of this system. This system has improved the typical way to handle the web based system of student checkup to the clinic. Besides to add new student, new medicine and to added stock of medicine the staff also can view the real time information about the medicine such as total medicine given to student based on the report provided from the system. The functional and non-functional requirement that is stated in the user requirement has also been delivered by this system. This project can bring out an effective enhancement of process and change in any system. However, there are still some improvements that must be put into consideration for future uses. By develop this system, various benefits can be provided and make it possible to improve the current system operation.
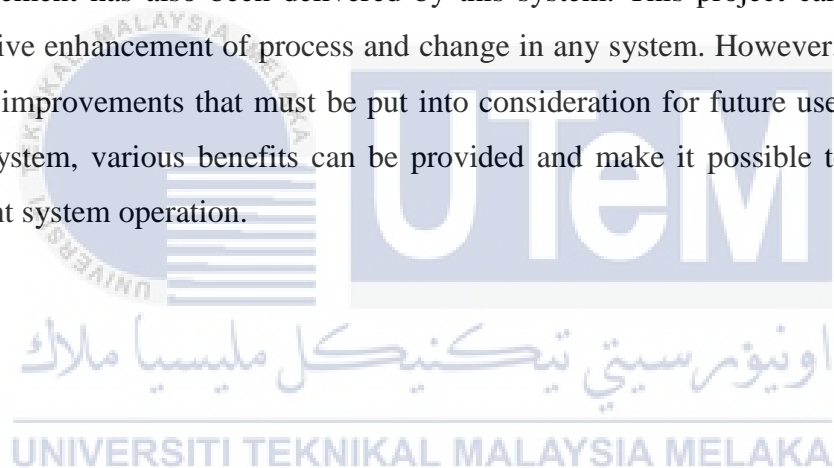
REFERENCES & BIBLIOGRAPHY

Robert Viera (2007)." Professional SQL Server 2005 Programming".Indianapolis

    Indiana : Wiley Publishing Inc. 329-742 .

S.Richard and M.Neil (2001)."Beginning Databases".UK :Wrox Press ltd. 47-168

 Anthony Butcher (2003). "Sams Teach Yourself MySQL in 21 Days".2nd

    Edition.United States of America :Sams Publishing. 1-172.

N.Priya,Greenberg.Nancy (2000)."Introduction to oracle 9i: SQL".Volume 1. US

    Government : Nita Brozowski .1-1 – 10-27.

SQL select statement. [Online] Available at:

    http://www.w3schools.com/sql/sql_select.asp

Oracle SQL Server for the Oracle pro. [Online] Available at:

    http://searchoracle.techtarget.com/guide/SQL-Server-for-the-Oracle-pro

Multiple Select Statements using SQL Server 2005 "WITH" Statement. [Online]

    Available at:

    http://stackoverflow.com/questions/954016/multiple-select-statements-using-

    sql-server-2005-with-statement

APPENDIX

GANTT CHART

	Task Name	Duration	Start	Finish
1	**Inventory Boutique System**	80 days	Mon 2/22/16	Fri 6/10/16
2	Proposal PSM : Submission & Presentation (Proposal assessment and verification)	5 days	Mon 2/22/16	Fri 2/26/16
3	Proposal Correction/Improvement (List of supervisor/title)	5 days	Mon 2/29/16	Fri 3/4/16
4	Chapter 1 (system Develoment Begins)	5 days	Mon 3/7/16	Fri 3/11/16
5	Chapter 1 & chapter 2	5 days	Mon 3/14/16	Fri 3/18/16
6	Chapter 2	5 days	Mon 3/21/16	Fri 3/25/16
7	Chapter 2 & Chapter 3 (Student Status)	5 days	Mon 3/28/16	Fri 4/1/16
8	Project Demo & Chapter 3, Chapter 4	5 days	Mon 4/4/16	Fri 4/8/16
9	MID SEMESTER BREAK	5 days	Mon 4/11/16	Fri 4/15/16
10	Project Demo & Chapter 4	5 days	Mon 4/18/16	Fri 4/22/16
11	project Demo & Chapter 4 (Student Status)	5 days	Mon 4/25/16	Fri 4/29/16
12	Project Demo (Determination of student status(Continue/Withdraw)	5 days	Mon 5/2/16	Fri 5/6/16
13	Project Demo & PSM Report	5 days	Mon 5/9/16	Fri 5/13/16
14	Project Demo & PSM Report (Presentation Schedule)	5 days	Mon 5/16/16	Fri 5/20/16
15	Project Demo & PSM Report	5 days	Mon 5/23/16	Fri 5/27/16
16	FINAL PRESENTATION (PA)	5 days	Mon 5/30/16	Fri 6/3/16
17	REVISION WEEK (Correction draft report based on supervisor's and evaluator's comments during the final presentation session) Submission overall marks to PSM/PD committee	5 days	Mon 6/6/16	Fri 6/10/16
18	FINALEXAMINATION SEMESTER			

Figure A.1 : Gantt chart of this project
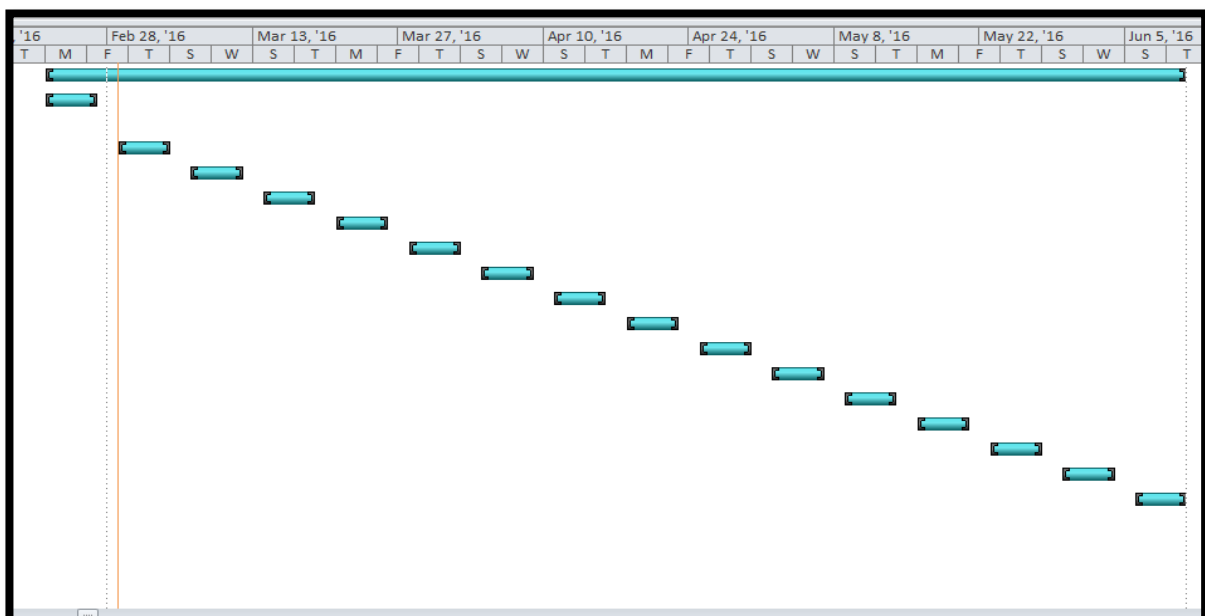
Figure A.2 : Gantt chart of this project

Trigger before insert,after insert and before delete

Before insert trigger to view time automatically.

```
create or replace trigger proc_trig_time
before insert on stud_sick
for each row
begin
:new.times:=to_char(sysdate,'hh:mi:ss AM');
end;
```

Before insert trigger for medicine category id

```
create or replace trigger trig_category_ID
before insert on medicine_category
for each row
declare
s_id medicine_category.MED_CATEGORY_ID%TYPE;
begin
select category_seq.nextval into s_id from dual;
if s_id < 10 then
:new.MED_CATEGORY_ID :='C0' || '000' || s_id;
elsif s_id < 100 then
:new.MED_CATEGORY_ID :='C0' || '00' || s_id;
elsif s_id < 1000 then
:new.MED_CATEGORY_ID :='C0' || '0' || s_id;
end if;
end;
```

Before delete trigger to update stock of medicine

```
create or replace trigger trig_delete_medicine
before delete on stud_med
for each row
begin
update medicine set stock_quantity = stock_Quantity + :old.quantity,
total_student = nvl(total_student,0) - 1
where medicine_ID = :old.medicine_ID;
end;
```

After insert trigger to update stock of medicine after take medicine

```
create or replace trigger trig_medicine_detail
after insert on stud_med
for each row
begin
update medicine set stock_quantity = stock_quantity - :new.quantity
where medicine_ID = :new.medicine_ID;

update medicine set total_student =nvl(total_student,0) + 1
where medicine_ID = :new.medicine_ID;
end;
```

After insert trigger to calculate total of medicine taken by student

```
create or replace trigger trig_total_medicine
after insert on medicine
for each row
begin
update medicine_category set total_medicine = nvl(total_medicine,0) + 1
where med_category_ID = :new.med_category_ID;
end;
```

After insert trigger to calculate total of student for certain disease

```
create or replace trigger trig_total_student
after insert on stud_sick
for each row
begin
update sickness set total_student = nvl(total_student,0) + 1
where disease_ID = :new.disease_ID;
end;
```

Stored procedure that used in this project

Stored procedure to deleteexisting student.

```
create or replace procedure delete_student(
matric_No1 in student.matric_No%TYPE
)
is
begin
delete from student WHERE matric_No= matric_No1;
end;
```

Stored procedure to insert new student

```
create or replace procedure insert_student(
matric_No in student.matric_No%TYPE,
first_Name in   student.first_Name%TYPE,
last_Name in   student.last_Name%TYPE,
student_class in   student.student_class%TYPE,
sex in   student.sex%TYPE,
weight in  student.weight%TYPE,
height in  student.height%TYPE,
blood_Group in   student.blood_Group%TYPE,
date_Of_Birth in  varchar2,
nation in student.nation%TYPE,
staff_ID in medicine.staff_ID%TYPE
)
is
begin
insert into student(matric_No,first_Name ,last_Name, student_class ,sex ,
weight,height, blood_Group,date_Of_Birth, nation,staff_ID) values
(matric_No,first_Name,last_Name,student_class,sex, weight, height,
blood_Group,to_date(date_Of_Birth,'yyyy-mm-dd'),nation,staff_ID);
end;
```

Stored procedure for staff login

```
create or replace PROCEDURE loginStaff(
      p_id IN   staff.staff_username%TYPE,
        p_pass IN  staff.staff_password%TYPE ,
        categories out varchar2,
        p_count out number
        )
IS
      BEGIN
       SELECT count(*),categories
       into p_count,categories
       FROM  staff
       WHERE staff_username = p_id and staff_password = p_pass
       group by categories;
       commit;
END;
```

Stored procedure to search student based on month

```
create or replace PROCEDURE search_bymonth(bulan in VARCHAR2,myrc out
sys_refcursor) as
  BEGIN
   OPEN myrc FOR SELECT DISTINCT  extract(MONTH from st.dates)as
bulan,s.matric_No,sm.quantity, s.first_Name
,me.medicine_Name,d.disease_Name,st.dates,st.times from student s,stud_med
sm, stud_sick st,medicine me, sickness d
where s.matric_No = st.matric_No
and sm.matric_No = st.matric_No
and me.medicine_ID = sm.medicine_ID
and d.disease_ID = st.disease_ID
and st.dates = sm.dates
and extract(MONTH from st.dates) = bulan
order by st.dates;
  END;
```

Stored procedure to search student based on matric no

```
create or replace PROCEDURE search_diseasename(WRDSRCH in
VARCHAR2,myrc out sys_refcursor) as
 BEGIN
  OPEN myrc FOR SELECT DISTINCT s.matric_No,sm.quantity, s.first_Name
,me.medicine_Name,d.disease_Name,st.dates,st.times from student s,stud_med
sm, stud_sick st,medicine me, sickness d
where s.matric_No = st.matric_No
and sm.matric_No = st.matric_No
and me.medicine_ID = sm.medicine_ID
and d.disease_ID = st.disease_ID
and st.dates = sm.dates
and st.dates >= sysdate -1
and s.matric_No LIKE '%' || WRDSRCH || '%'
OR s.first_Name LIKE '%' || WRDSRCH || '%'
order by st.dates asc;
 END;
```

Stored procedure to update student information

```
create or replace procedure update_student(
matric_No1 in student.matric_No%TYPE,
student_class1 in  student.student_class%TYPE,
height1 in  student.height%TYPE,
weight1 in  student.weight%TYPE

)
is
begin
UPDATE student set student_class=student_class1,height=
height1,weight=weight1
WHERE matric_No= matric_No1;
end;
```

Stored procedure to view all the student

```
create or replace procedure view_student(myrc out sys_refcursor) as
begin
open myrc for select * from student;
end;
```

Stored procedure to view report on a pie chart

```
create or replace procedure view_pie1(myrc out sys_refcursor) as
begin
open myrc for select s.sex,dd.disease_Name, count(*) as total from sickness dd,
student s, stud_sickst
where dd.disease_ID = st.disease_ID
and s.matric_No = st.matric_No
and s.sex ='Male'
group by dd.disease_Name, s.sex
order by total;
end;
```

Stored procedure to view student on a bar chart

```
create or replace procedure view_report(myrc out sys_refcursor) as
begin
open myrc for select max(total)as total,matric_No from
(select count(matric_No) as total,matric_No from stud_sick group by matric_No)
group by matric_No
order by total desc;
end;
```

Graphical User Interface (GUI) design.
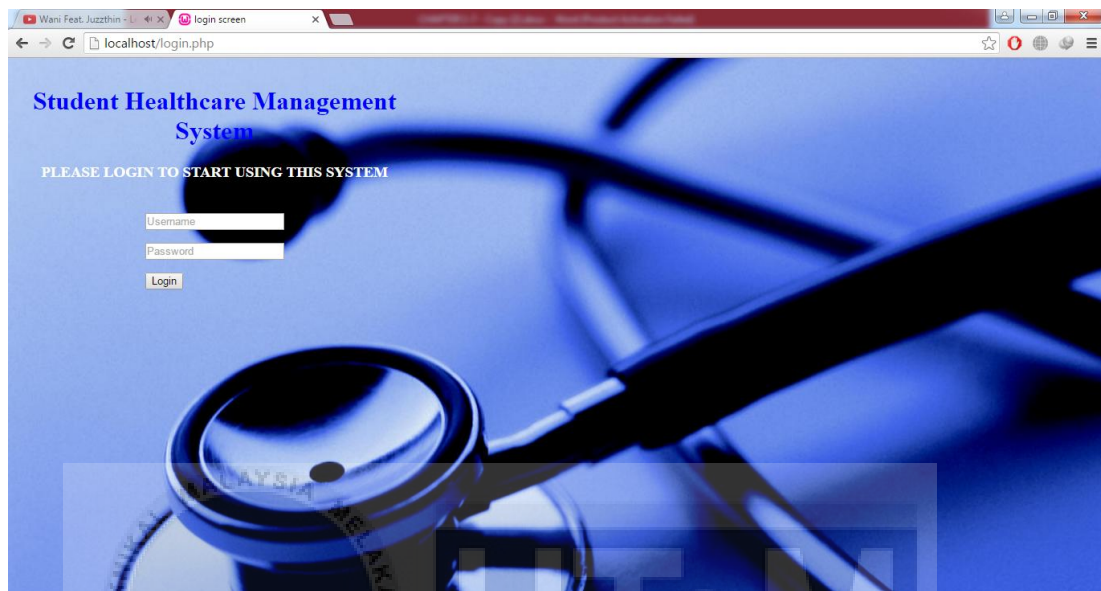
Figure A3 : Application home page
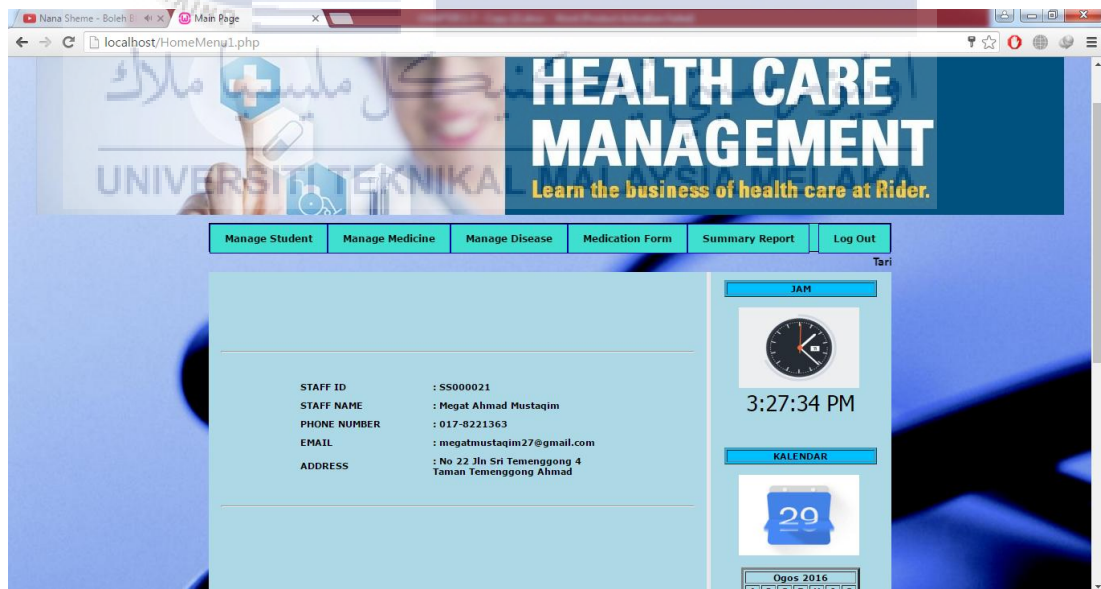
Figure A4: Doctor home page
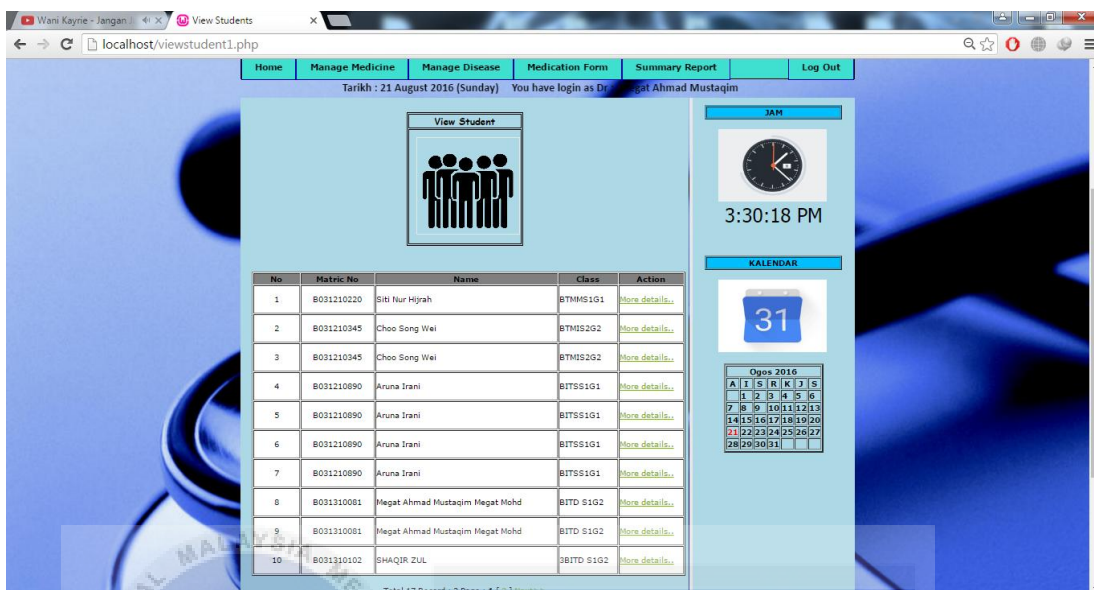
Figure A5: List of student
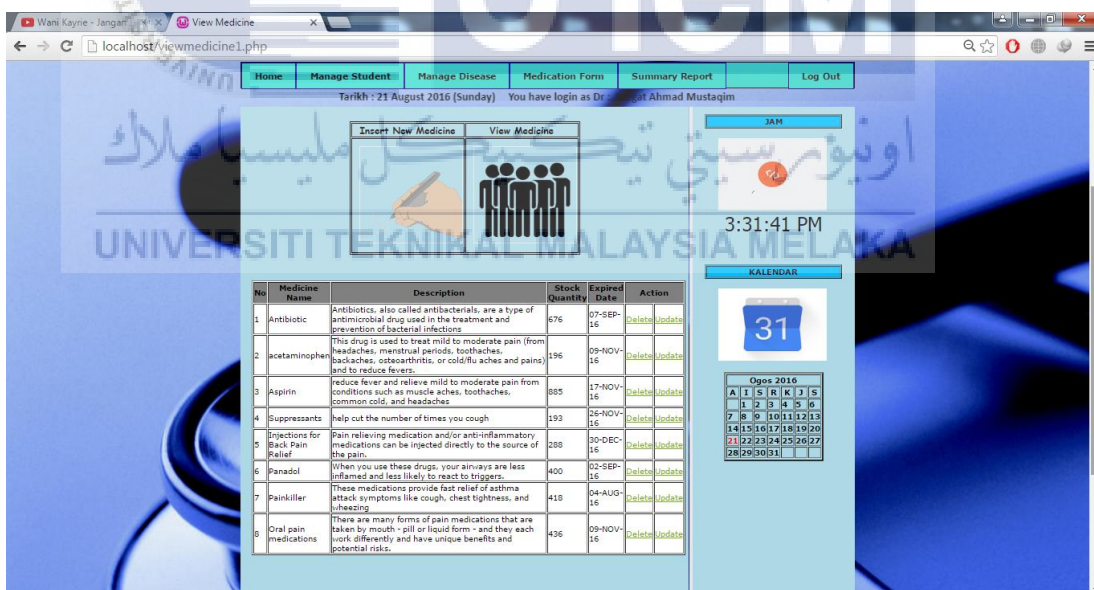
Figure A6: List of medicine
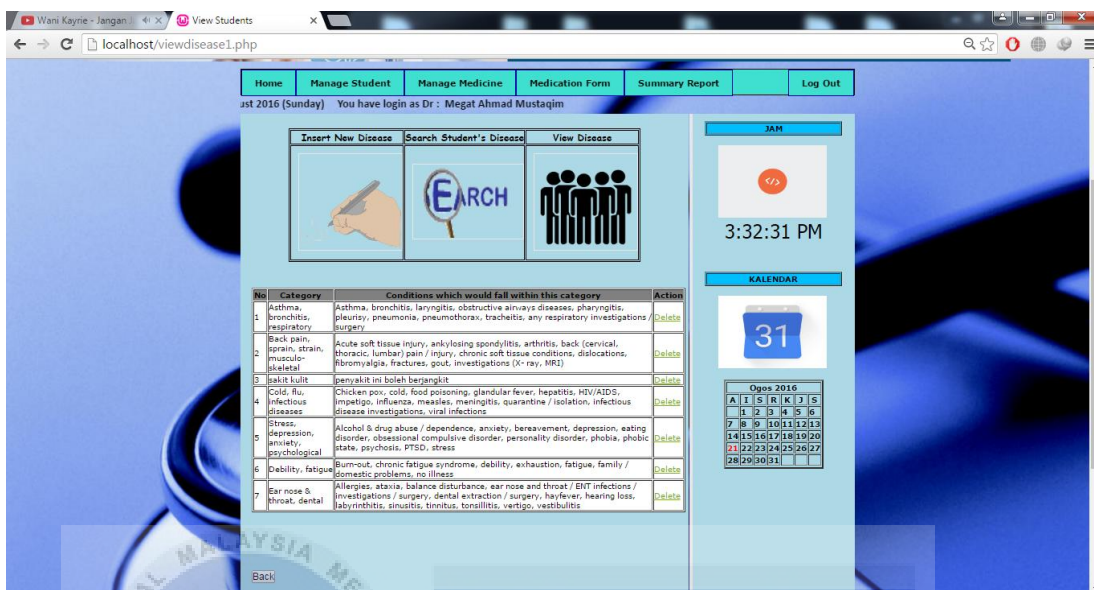
Figure A7: List of diseases

Figure A8: Medication form
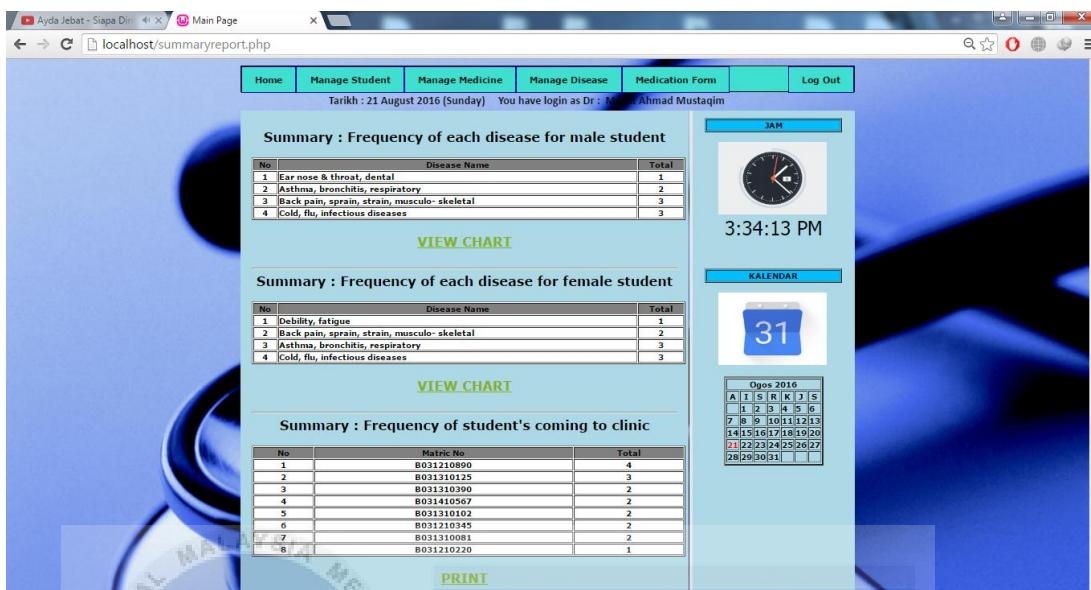
Figure A9: View report
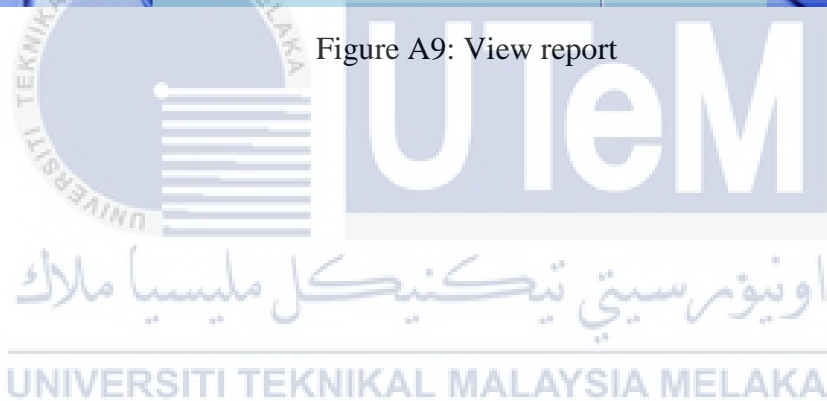
User manual

This is the login page for staff that is doctor and nurse before start using this system.

For doctor:-
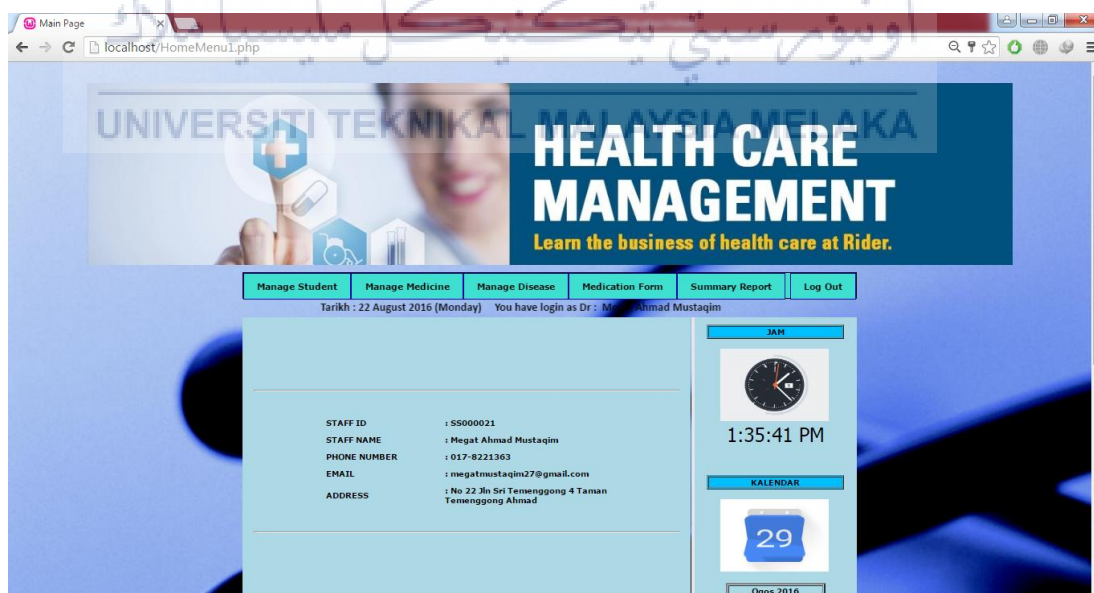
This is the homepage for doctor after login. There are 5 menus that doctor can do, which are manage student, medicine, disease, medication form and view repoprt.

When doctor click on manage student, he/she can only view the student that he/she has treated.

When doctor click on manage medicine, he/she can insert new medicine or view the medicine that is already in the stock.

The form when doctor wants to add new medicine into the system. This process can only be done by doctor.

When doctor click on view medicine all medicine will be shown.
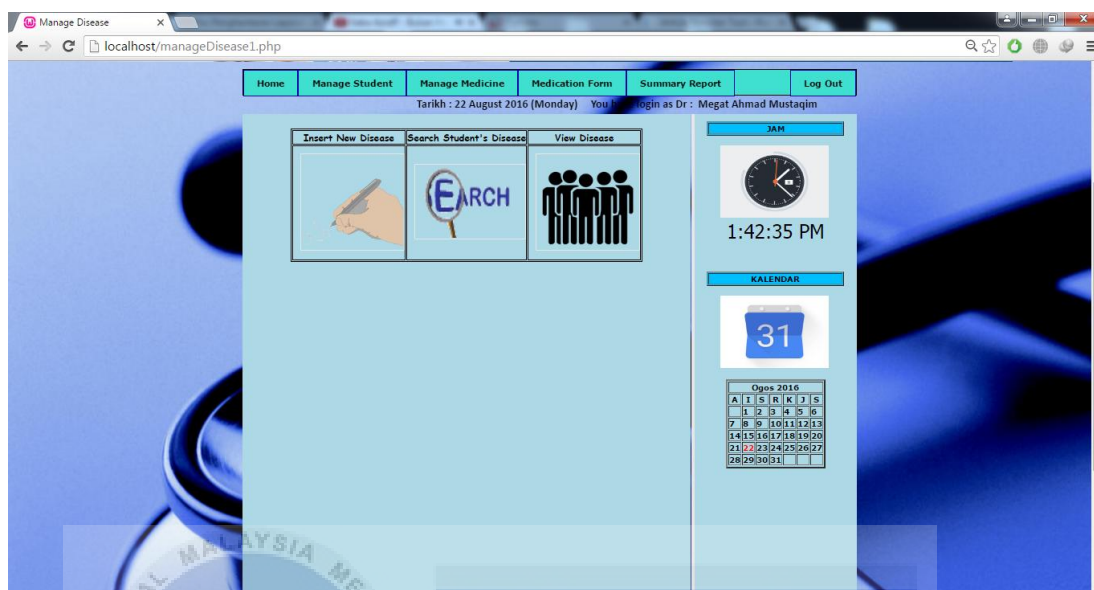
When doctor click on manage disease, there are 3 option that he/she can do. Firstly is, insert new disease, search student info for certain disease and view the disease.
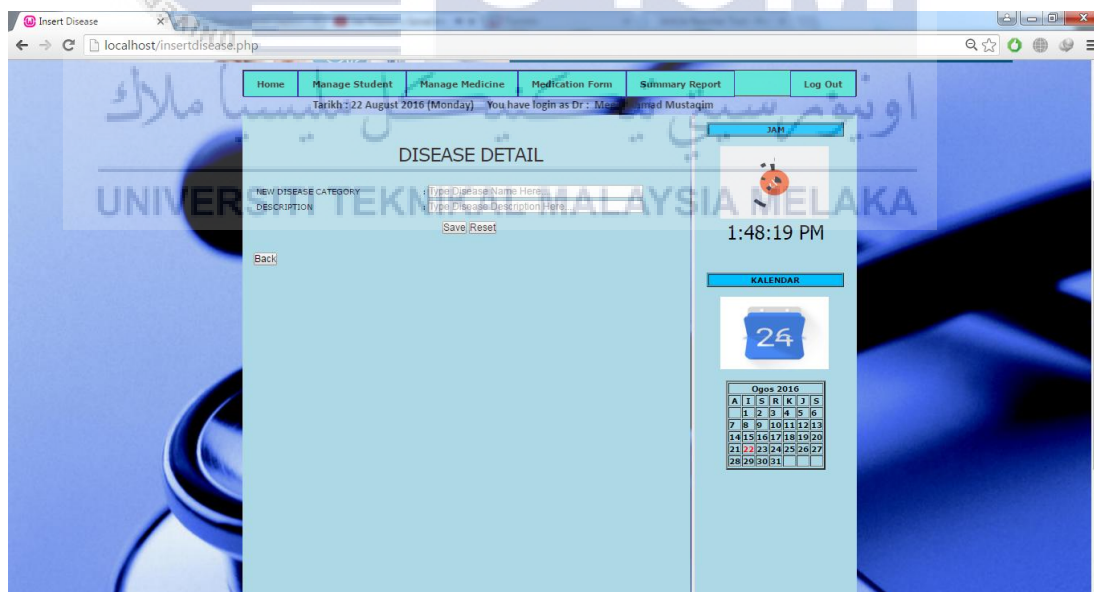
Doctor can insert new disease.
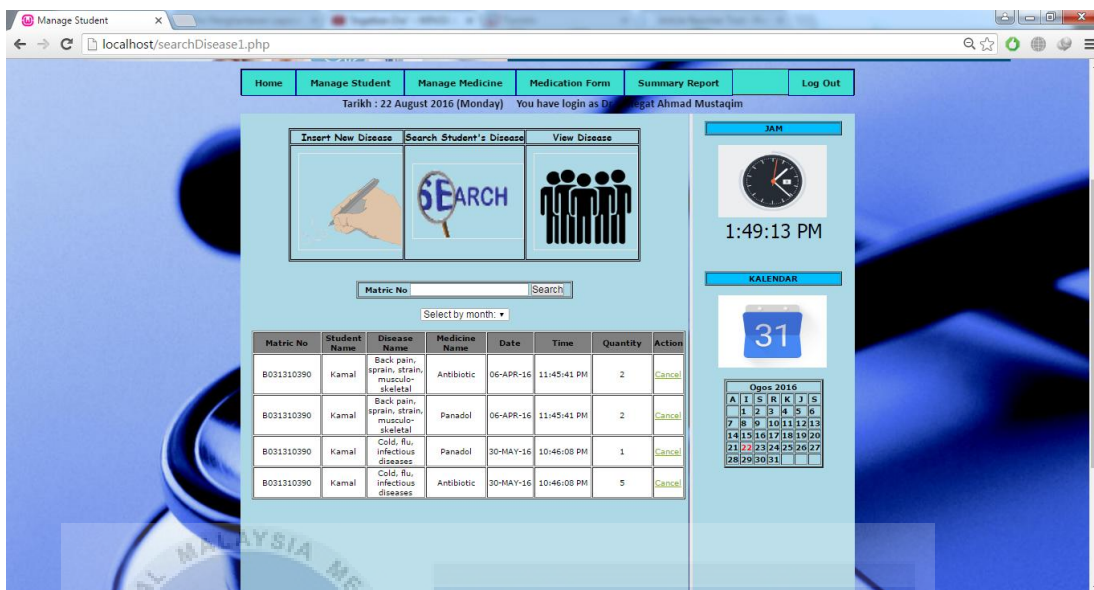
When doctor click on search student disease, there are another 2 option whether to search student based on matric number or by month.
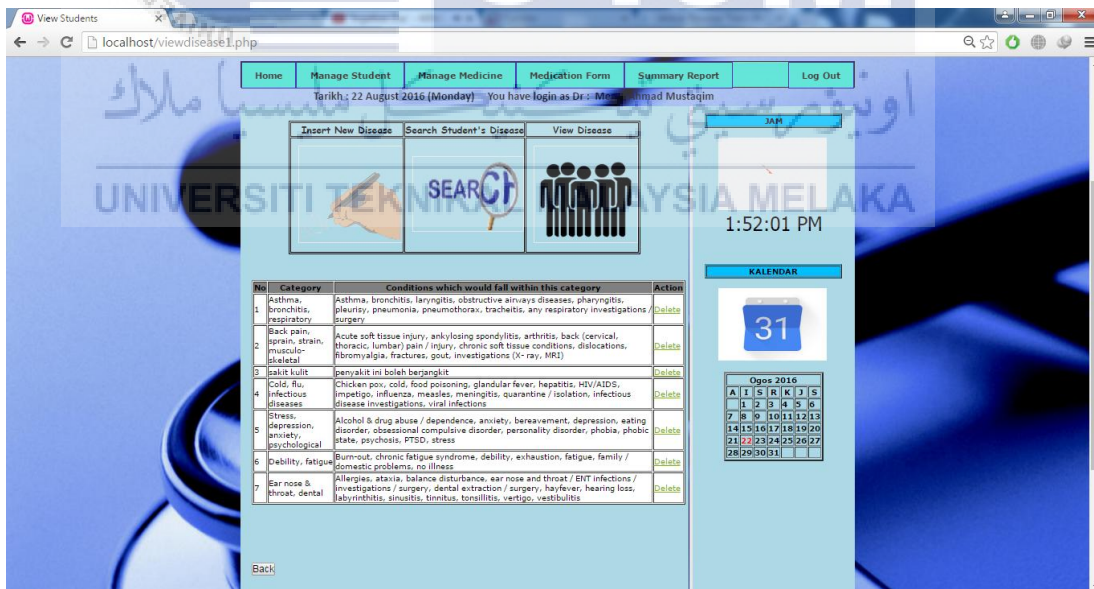
All details of disease will be shown when doctor click on view disease.

This is the form for doctor to insert student details before take medicine.

After doctor click on save button, data will be saved and medicine can be choose before give it to student for disease that they suffer.
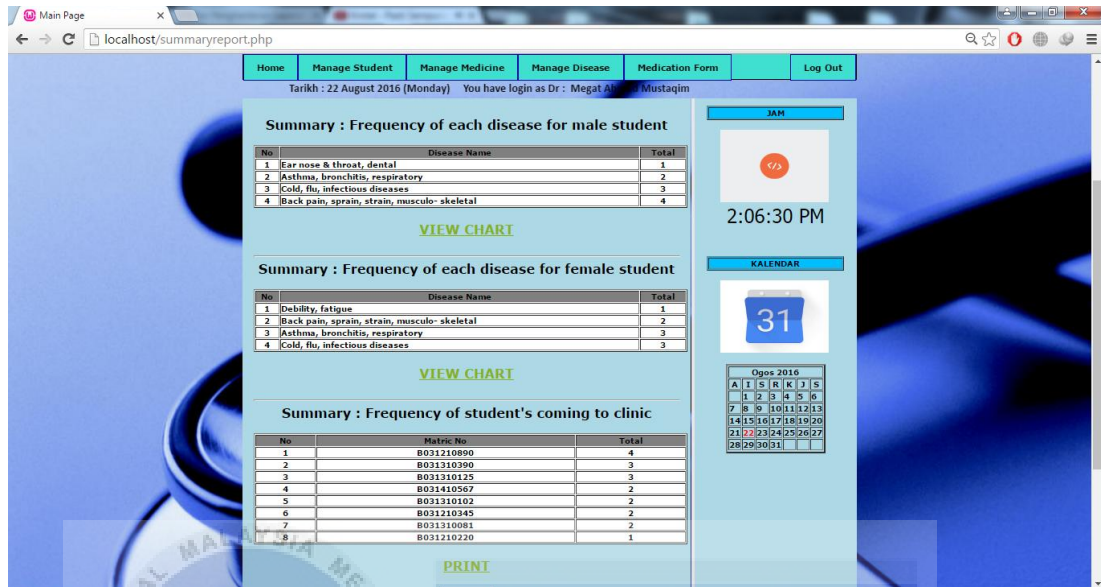
For the last menu doctor can view the summary report for example frequency of student coming to clinic, frequency of each disease for male and female student, etc.
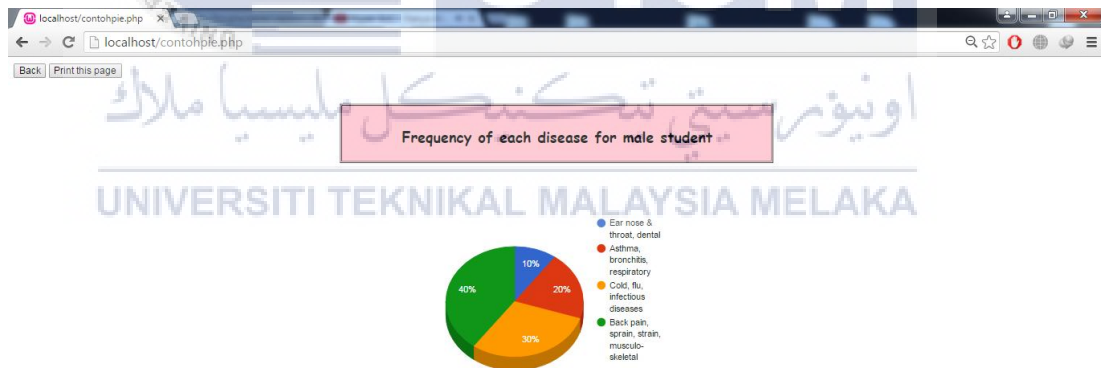
Doctor can also view the chart to see the percentage.

For nurse:-
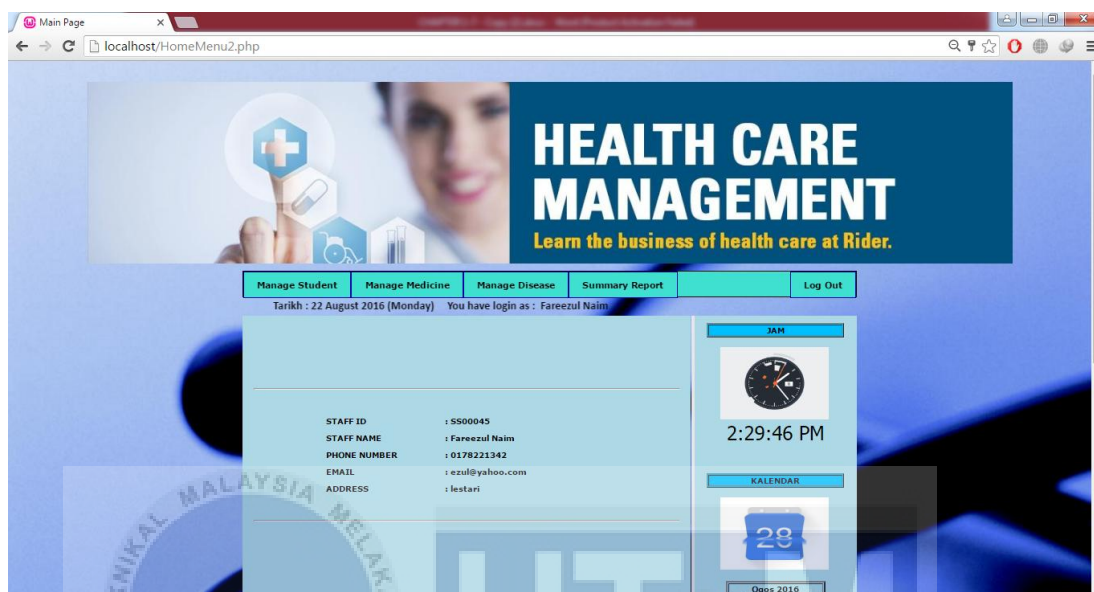
Homepage for staff, there are one menu that is not in nurse homepage but has in doctor homepage. It is medication form menu.
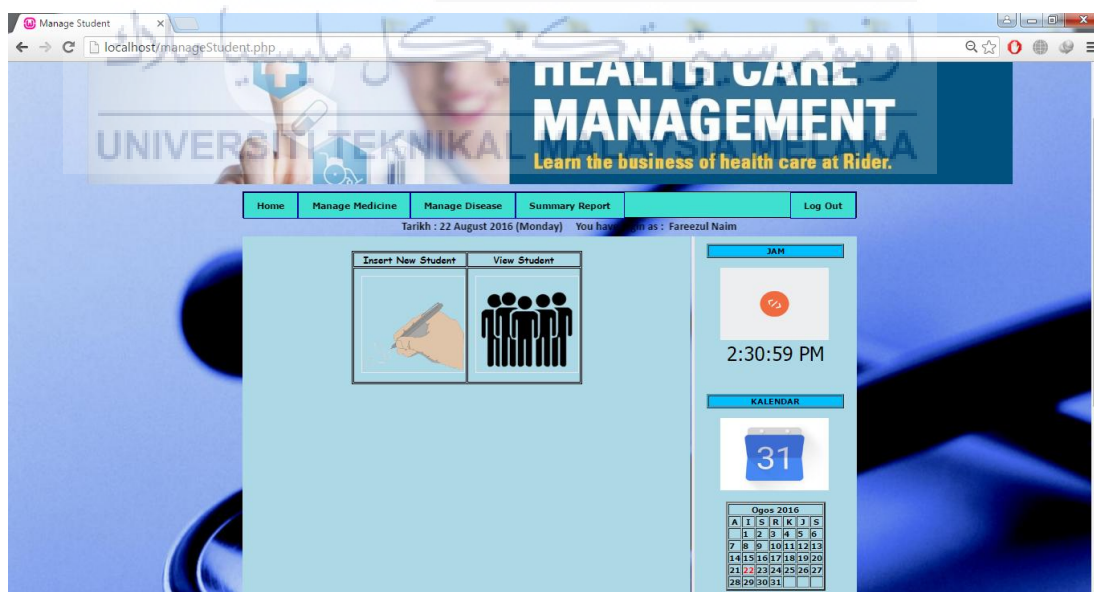
For manage student menu, it has 2 option instead of one option for doctor. Only nurse can add new student for registration process.

Registration form for new student.

For view option, nurse can view all student that has registered in the system.

For manage medicine menu, nurse can only view the medicine and cannot edit anything.

For manage disease menu, nurse can only search for student to view the disease suffered and view the disease. They cannot add new disease like the doctor can do.

107

For summary report menu is just the same as the doctor can do. Nurse can view the statistic of student coming to clinic and other else.
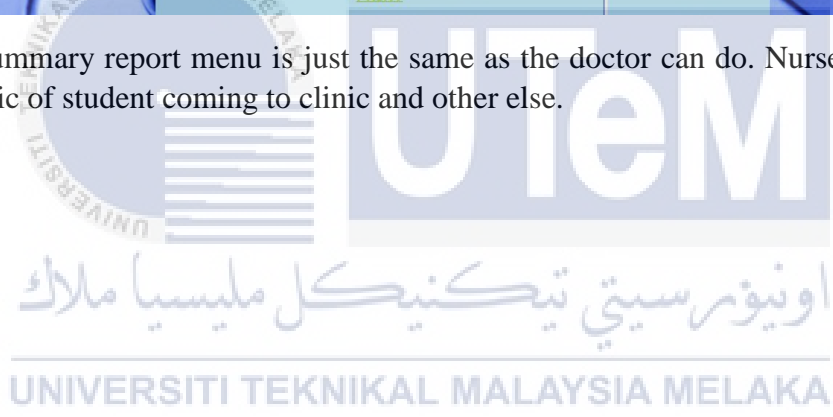
The user acceptance testing

Question	YES/NO (first user)	YES/NO (second user)
Learning to operate the system was easy	YES	YES
It took a reasonable amount of time to complete most task	YES	NO
Procedures were simple and required a minimum number of steps	YES	YES
Errors were easy to recover from and the error messages gave useful information	YES	NO
The interface, menus and screens were laid out in a logical fashion	YES	YES
The prompts and error messages were appropriate in form	YES	YES
Command name and options made sense	NO	YES
The computer did not take over the work or 'get in the way'	YES	YES
Pacing was comfortable,neither too fast nor too slow	YES	YES
The user manual was informative	YES	NO
It was easy to find the commands and information required to complete the task	YES	NO