BLOOD BANK MANAGEMENT SYSTEM

(BBMS)

SITI ROSNIEZA EILISA BINTI JAMAL

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

DECLARATION

BLOOD BANK MANAGEMENT SYSTEM

Is written by me and is my own effort and that no part been plagiarized without citation.

STUDENT : _____ Date: 24 AUGUST 2017

(SITI ROSNIEZA EILISA BINTI JAMAL)

I hereby declare that I have read this project report and found this project report is sufficient in term of the scope and quality

for the award of Bachelor of Computer Science (Database Management) With Honours.

SUPERVISOR : _____ Date: 25/8/2017

(DR. SAFIZA SUHANA BINTI KAMAL BAHARIN)

BLOOD BANK MANAGEMEMT SYSTEM

(BBMS)

SITI ROSNIEZA EILISA BINTI JAMAL

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

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2017

BORANG PENGESAHAN STATUS TESIS*

JUDUL: __BLOOD BANK MANAGEMENT SYSTEM__

SESI PENGAJIAN: __2017 /2018__

Saya __SITI ROSNIEZA EILISA BINTI JAMAL__

<div align="center">(HURUF BESAR)</div>

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

1. Tesis dan projek adalah hakmilik 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 1972)	
_____ TERHAD	(Mengandungi maklumat TERHAD yang telah ditentukan oleh organisasi/badan di mana penyelidikan dijalankan)	
____✓____ TIDAK TERHAD		

(TANDATANGAN PENULIS)

(TANDATANGAN PENYELIA)

Alamat tetap: __NO. 10 LOT 3641__
__PARIT 7½ BARAT,__
__45300 Sg BESAR, SEL.__
Tarikh: __24 AUGUST 2017__

Tarikh: __24 / 8 / 2017__

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

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

DEDICATION

To my beloved mother and father whom I love so much,

Study in IT is hard.

This is for you.

ACKNOWLEDGEMENTS

ABSTRACT

Blood Bank Management System (BBMS) is a system which is proposed to improve the efficiency in managing blood donated by the donor. The hospital is still used manual system and there are problems in managing the donors' records. The records of the donor is not be kept safely. This problem will lead to the missing of donor's records due to human error or any disasters. Besides, errors might occur when the staff keeps more than one record for the same donor. Since there is also no centralized database used to keep the donors' records, the data duplication of the donor's record might be occur. In addition, there are also problems in keeping track of the actual amount of the blood in the blood bank. There is also no alert available when the blood in the bank has expired. Because of that, this system is proposed. The primary aims of this project is to enhance the efficiency of blood department in the hospitals. BBMS is a computer-based system. The user of this system is the administrator and staffs who worked at the blood bank. This system will save the information of the staff, the donors and the receiver of the blood. With this system, the records of all donors and receiver of blood and their history are store in one centralized database and thus reducing duplication of data in the database. The record of donation is maintained by the system. The blood bank staff can manage the blood stock. All the blood will be stored in the blood bank according to the blood type. When blood is removed from the blood bank to a receiver in need, the quantity of blood is updated. The system will also provide an alert system to the staff whenever the blood when the blood in stock has expired. In the other hand, the system is able to generate pre-defined reports such as the list of donor, volunteers, the blood receiver and the amount of blood donated according to the blood type.

ABSTRAK

Sistem Pengurusan Darah Bank (BBMS) adalah sistem yang dicadangkan untuk meningkatkan kecekapan dalam menguruskan darah yang disumbangkan oleh penderma. Hospital masih digunakan sistem manual dan ada masalah dalam menguruskan rekod penderma. Rekod penderma tidak disimpan dengan selamat. Masalah ini akan mengakibatkan hilangnya rekod penderma kerana kesilapan manusia atau sebarang bencana. Selain itu, kesilapan mungkin berlaku apabila kakitangan menyimpan lebih daripada satu rekod untuk penderma yang sama. Memandangkan tidak ada pangkalan data terpusat yang digunakan untuk menyimpan rekod penderma, duplikasi data rekod penderma mungkin berlaku. Di samping itu, terdapat juga masalah untuk menjejaki jumlah sebenar darah di dalam darah. Tidak ada juga waspada apabila darah di bank telah tamat. Oleh itu, sistem ini dicadangkan. Matlamat utama projek ini adalah untuk meningkatkan kecekapan jabatan darah di hospital-hospital. BBMS adalah sistem berasaskan komputer. Pengguna sistem ini adalah pentadbir dan kakitangan yang bekerja di bank darah. Sistem ini akan menyelamatkan maklumat kakitangan, penderma dan penerima darah. Dengan sistem ini, rekod semua penderma dan penerima darah dan sejarah mereka disimpan dalam satu pangkalan data berpusat dan dengan itu mengurangkan pendua data dalam pangkalan data. Rekod sumbangan dikekalkan oleh sistem. Kakitangan bank darah boleh menguruskan stok darah. Semua darah akan disimpan di dalam bank darah mengikut jenis darah. Apabila darah dikeluarkan dari bank darah kepada penerima yang memerlukan, kuantiti darah dikemas kini. Sistem ini juga akan menyediakan sistem amaran kepada kakitangan apabila darah apabila stok darah telah tamat tempoh. Sebaliknya, sistem ini dapat menghasilkan laporan yang telah ditetapkan seperti senarai penderma, sukarelawan, penerima darah dan jumlah darah yang disumbangkan mengikut jenis darah.

TABLE OF CONTENTS

CHAPTER VII   CONCLUSION

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBRIVATIONS

BBMS	-	Blood Bank Management System
DBMS	-	Database Management System
DCL	-	Data Control Language
DDL	-	Data Definition Language
DML	-	Data Manipulation Language
GUI	-	Graphical User Interface
ID	-	Identity
PSM	-	Projek Sarjana Muda
SDLC	-	System Development Life Cycle
DBLC	-	Database Life Cycle
SQL	-	Structured Query Language

LIST OF ATTACHEMENT

CHAPTER I

1.0 INTRODUCTION

1.1   Introduction

Nowadays, the public can knows about the blood donation events through conventional media means such as radio, newspaper or television advertisements. With all the advertisement, people would love to donate their blood as they can help person who need blood to save life. Usually, hospitals or clinics is the organization that manages the blood donation campaign. After the donation, all the blood will be stored in a place called blood bank for people in need.

For the staff who manage the blood bank, it becomes difficult for the staff worked in the blood bank to search blood in case of emergency. The only option is to manually search and match donors and then make phone calls to every donor. This is because all the records of the donor is not store in the centralized database.

The hospital is having their own records of donors but the donor's record is save in the books. If a donor makes donation in different days, no previous records can be traced except if the donor brings along the donation certificate.

Hence, the donor is considered to be a first-timer if they make blood donation. Besides, without an automated management system, there are also problems in keeping track of the actual amount of the blood in the blood bank.

1.2 Project Background

Since there are problems faced by the staff worked in the blood bank department in managing the blood bank management, so I have been come out with a new solution. The idea is to develop a new system named Blood Bank Management System (BBMS). BBMS is a computer-based system that has been develop to solve the problems faced while managing the blood bank in the same time to change the way of blood department management from the manual system to the automated system.

1.3 Problem Statements

i. The current system in managing donors' records are still using manual system (paper-based) and there is no guarantee of safety because no centralized database to keep the donors' records.

ii. Difficult to find list of donors when needed.

iii. The actual amount of blood in the blood bank is difficult to keep track.

iv. Staff do not know the expired date of the blood in the blood bank because no alert are available when the blood is expired.

1.4 Objectives

i. To develop a system that can save all the records of donors in a centralized database.

ii. To provide an efficiency searching platform of the donors when needed.

iii. To easier the staff to keep track actual amount blood in the blood bank.

iv. To provide a platform that can help staff checking blood that will be expired by providing an alert system in the system.

1.5 Scopes

The project scope is divided into two section which are the module to be develop and the target user for the system.

1.5.1 Module to Develop

a) Human Recourse
- Store the information of manager (administrator) and the staff.

b) Donor/Receiver Management
- The records of all donors and receivers and their history are stored in one centralized database. This reducing duplicate data in the database. The record of donation is maintained by the system.

c) Blood Management
- All the blood donated by the donors will be saved in this system. The system will provide a feature of alert for the blood if the blood is going to be expired within one week. So that the staff can get the notification and as a reminder the next process of the blood.

d) Blood Bank Management
- This module is developed for the manager or staff who manage the blood bank. All the blood will be stored in the blood bank according to the blood type. When blood is removed from the blood bank to a receiver in need, the quantity of blood is updated. The system will also raise alert to the staff whenever the blood when the blood in stock has expired.

e) Reporting
- The system is able to generate pre-defined reports such as the list of donor, volunteers, the recipient and the amount of blood donated according to the blood type. Besides, this system can also generate the repost of analysis how many blood have been transferred to the receiver for every month.

1.5.2 Target User

i. Manager
- Manager is the administrator of the system.
- Manager has the full privileges of the function in the system.
- Manager can view the report according to what they want to see.

ii. Staff
- The staff is person who worked in the blood bank.
- Staff are responsible in managing the registration of donor, the blood stock in blood bank and the transfer of blood to the receiver.

1.6 Project Significance

1. To develop a system that can replace the current system in managing the blood bank effectively.
2. To make sure all the donors' record stored in the centralized database to avoid the duplication of data.
3. To prevent the blood bank from stores blood that has expired and cannot be used again.

1.7 Expected Output

Output 1: The total blood that have been transferred to the receiver in every month can be calculate accurately and printed through the system.

Output 2: System can display the alert to remind the expired date for the blood stored in the blood bank.

1.8   Conclusion
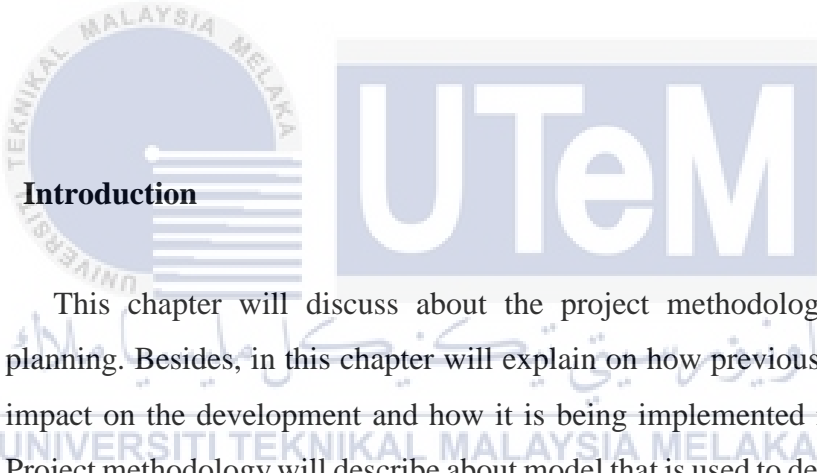
As a conclusion, Blood Bank Management System (BBMS) is a computer-based system that is develop to replace the current system in managing the blood bank. This system is allow the manager and the receptionist as the user of the system. The system provide a new platform for the receptionist to store record of the donor in the centralized database. This will avoid system from store the duplicate data of donors. This system also have the new features which is can give alert to the receptionist when the blood in the blood bank has expired. This is a very effective way to maintain the availability of the blood bank. With this new system, the management of the blood bank will became more convenient compare to the current system. The next chapter will discuss about the project methodology and system planning.

CHAPTER II

2.0 PROJECT METHODOLOGY AND PLANNING

2.1   Introduction

This chapter will discuss about the project methodology and system planning. Besides, in this chapter will explain on how previous research have impact on the development and how it is being implemented in this project. Project methodology will describe about model that is used to design, planning, implement and achievement of project objectives. In this chapter literature review will be focused on previous research and analysis that have been done and the methodology being use for the development of this project.

2.2   Project Methodology

To develop any system, the method chosen to be used is very important. This is because the method is used to estimate the time taken of the system to be delivered on the stage. To develop Blood Bank Management System (BBMS), Incremental Model is applied. The main reason Incremental Model is chosen as the method to develop this system is because when there are any problem arise while developing the system, the previous stages can be refer

and make the correction on parts which causes the problems to arise. Besides, the Incremental Model is simple and easy to understand and apply.

2.2.1 General Overview of Incremental Model
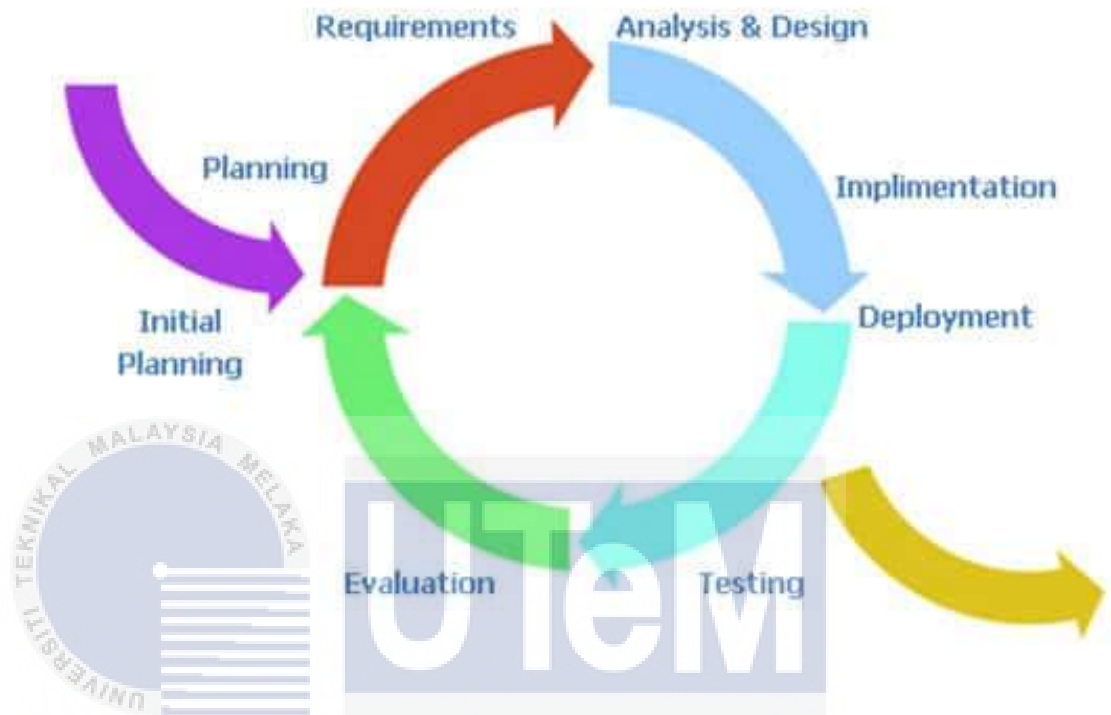
Figure 2.1 Incremental Model

2.3 Project Schedules and Milestones

Project schedule and milestones will help to make sure that the development and deployment of Blood Bank Management System in track.

Table 2.1 PSM 1 Schedule and Milestone

PSM 1				
Week	**Activities**	**Start Date**	**End Date**	**Deliverables**
1	Discussion of the project title and system to be developed with the supervisor.	13/02/2017	17/02/2017	Student finalize project proposal
2	Submission and presentation of system proposal to the supervisor.	20/02/2017	24/02/2017	Proposal submission and presentation to the supervisor
3	Prepare and submit ERD. Present ERD to supervisor	27/02/2017	03/03/2017	Student present the completely ERD to the supervisor
4 - 5	Prepare the system interface	06/03/2017	17/03/2017	Student present the system interface to the supervisor
6	Chapter 1: Introduction report submission. Chapter 2: Project Methodology and Planning report submission.	20/03/2017	24/03/2017	Student prepare Chapter 1: Introduction report submission to the supervisor Chapter 2: Project Methodology and Planning report submission to the supervisor
7	Chapter 3: Analysis report submission.	2703/2017	31/03/2017	Student prepare Chapter 3: Analysis report submission to the supervisor

8 – 10	Database design Chapter 4: Design report submission Progress 1 presentation	3/04/2017	21/04/2017	Chapter 4: Database Design report submission and presentation of Progress 1 to the supervisor
11	System testing Preliminary assessment of student status	24/04/2017	28/04/2017	System testing result and preliminary of student assessment by supervisor
12	Submission of complete report	1/05/2017	5/05/2017	Chapter 1 – 4 Full complete report submission to the supervisor
13	Full complete system	08/05/2017	12/05/2017	Full complete system and present to the supervisor
14 – 15	Full complete report Final presentation	15/05/2017	26/05/2017	Full complete report submission to the supervisor and final presentation to the assessor

Table 2.2 PSM 2 Schedule and Milestone

PSM 2				
Week	**Activities**	**Start Date**	**End Date**	**Deliverables**
1	Chapter 5: Implementation report submission	29/05/2017	2/06/2017	Student prepare Chapter 5: Implementation report submission to the supervisor
2	Chapter 6:	5/06/2017	09/06/2017	Student prepare

	Testing report submission			Chapter 6: Testing report submission
3	Chapter 6: Testing report submission	12/06/2017	16/06/2017	Student prepare Chapter 6: Testing report submission
4	Student assessment	19/06/2017	23/06/2017	Presentation of Progress 2 to the supervisor
5	Chapter 7: Conclusion report submission	26/06/2017	30/06/2017	Student prepare Chapter 7: Conclusion report submission
6	Chapter 7: Conclusion Complete PSM draft report	3/07/2017	7/07/2017	Student prepare Chapter 7: Conclusion report submission
7	Final presentation	10/07/2017	14/07/2017	PSM draft report submission to PSM committee, assessor and supervisor.
8	PSM draft report	17/07/2017	21/07/2017	Students marks by supervisor
9	Final presentation	24/07/2017	28/07/2017	Complete the PSM 2

2.4   Conclusion

As a conclusion, the project methodology and literature review is discussed in this chapter. The project methodology is used to define the project planning. The Incremental Model method is applied for developing BBMS. Besides, time schedules and project milestone also included in this chapter as well. The project schedule and milestone will be as the time management of developing this system.

CHAPTER III

3.0 ANALYSIS

3.1   Introduction

In this chapter, the analysis phase during Blood Bank Management System (BBMS) will be discussed. Before developing the system, analysis on the current system should be done. This is to be sure the new system can overcome the weaknesses of the current system. In the problem analysis section, the problems of the current system will be identified. The flow chart will be used as a diagram to illustrate the flow of the current system.

Next, in the requirement analysis section will identify and explain the functional requirement existing data, software requirements, hardware requirements and network requirement. All of the requirements are needed to identify to understand how the current system works.

The analysis phase describes and gathers information about current application being developed as well as identifying high-level requirements by detailing functional and non-functional requirements of the application. For problem analysis, the problem of the current system will be explained in detail and flowchart is used to show the flow of the system. Requirements analysis will cover the topic about the data requirement which is the data that should be the system input and output and the data that the system should store internally.

Non-functional requirement will cover the system performs its intended functions. It will be carried out by investigating the application scenario and extracting the required analysis information as described in the following sub-chapter.

3.2   Project Analysis

Problem analysis is a process of investigation in order to find and understand the problem, at the same time leads to find ways to solve it. The process of problem analysis for the Blood Bank Management System will include the current system analysis and the proposed improvements for the system.

3.2.1   Current System Analysis

Currently, blood department in the hospital is still using manual system. There are several problems have been identified in the current process of staff, donor, and receiver registration, the management of blood in the blood bank and generate report.

3.2.1.1 Donor and Receiver Registration

The main problem of registration system is there are no centralized database that can keep all the data and information. In some point, there also still used paper-based when the new customer want to make a registration as a donor and as a blood receiver.
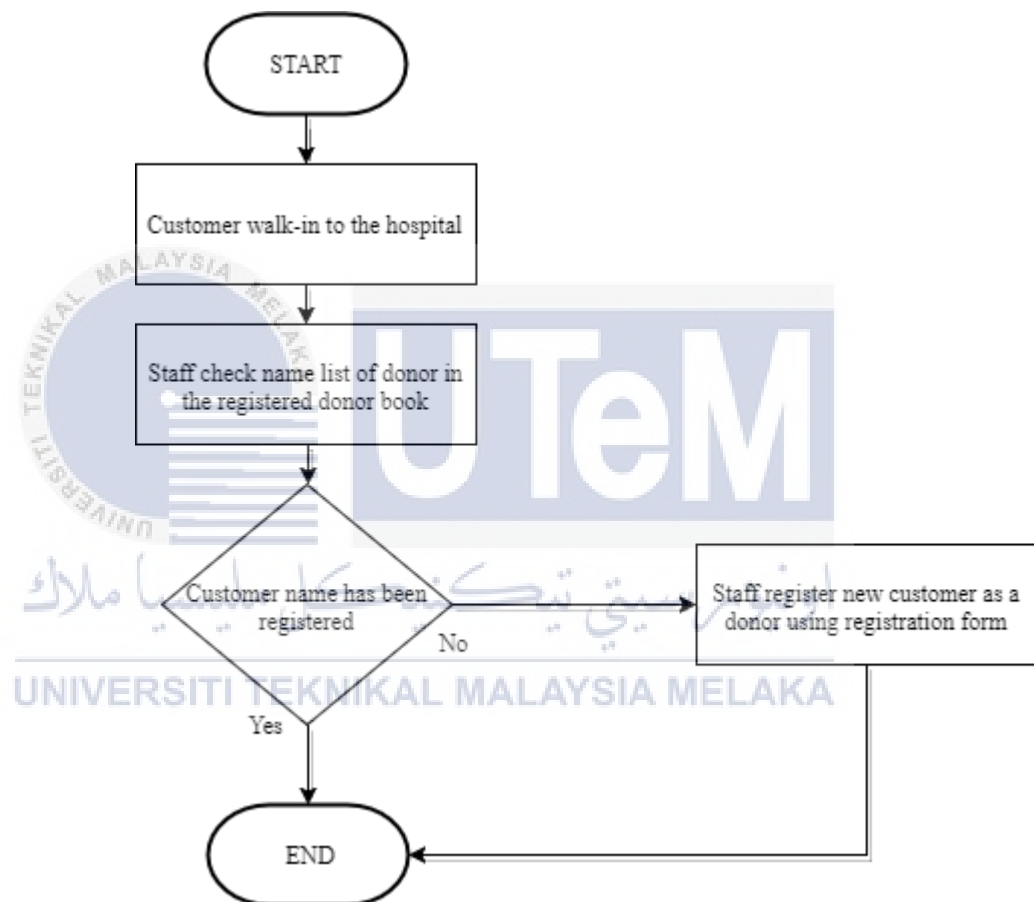
Figure 3.1 Current Donor Registration Flow Chart
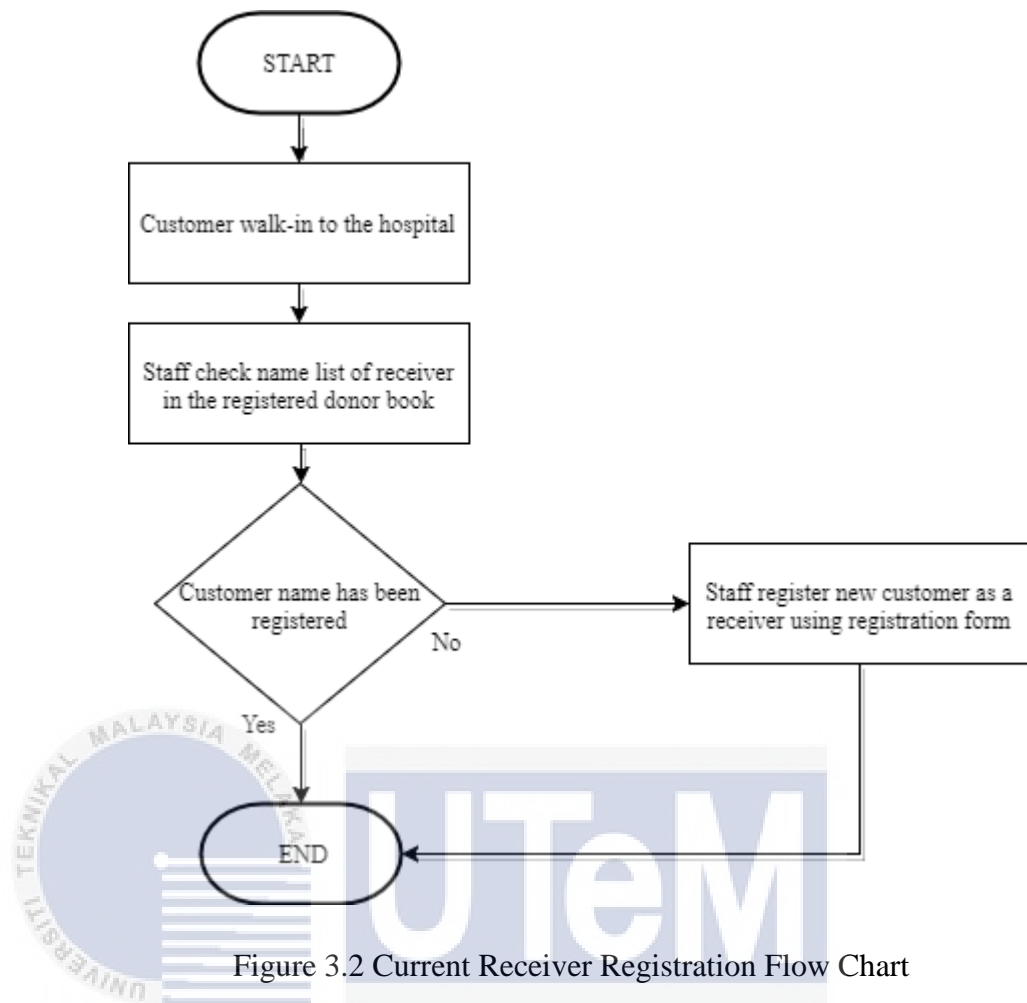
Figure 3.2 Current Receiver Registration Flow Chart

3.2.1.2   Blood Stock Management

For the blood stock management, there are no system that can keep track actual amount of each type of blood in the blood bank. Staff need to check manually to know the total amount of blood in the blood bank for each type of blood every week.
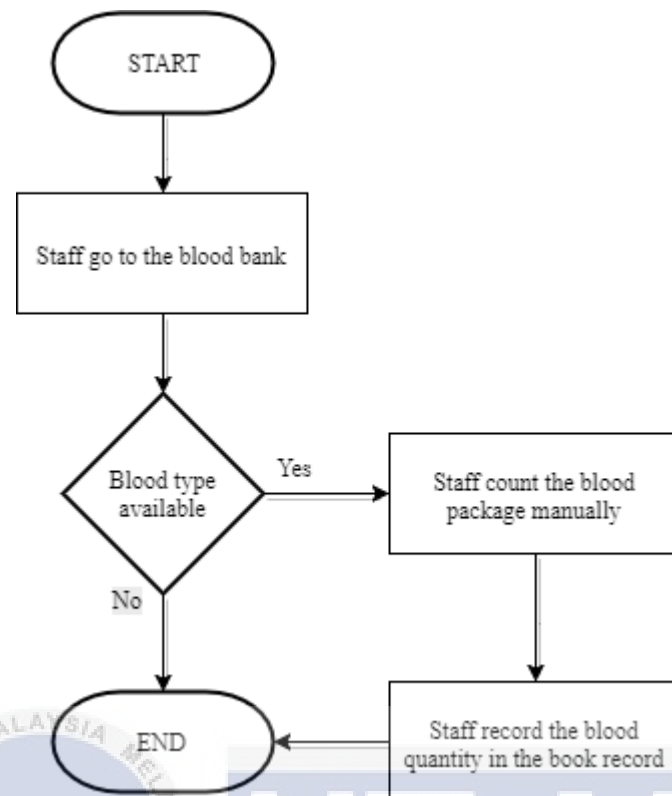
Figure 3.3 Current Blood Stock Management Flow Chart

3.2.1.3 Blood Expired Management

Staff usually go to the blood bank to check the expired date of each blood. Staff will check the blood expired manually because there is no system that can give alert if the blood in the blood bank will be expired.
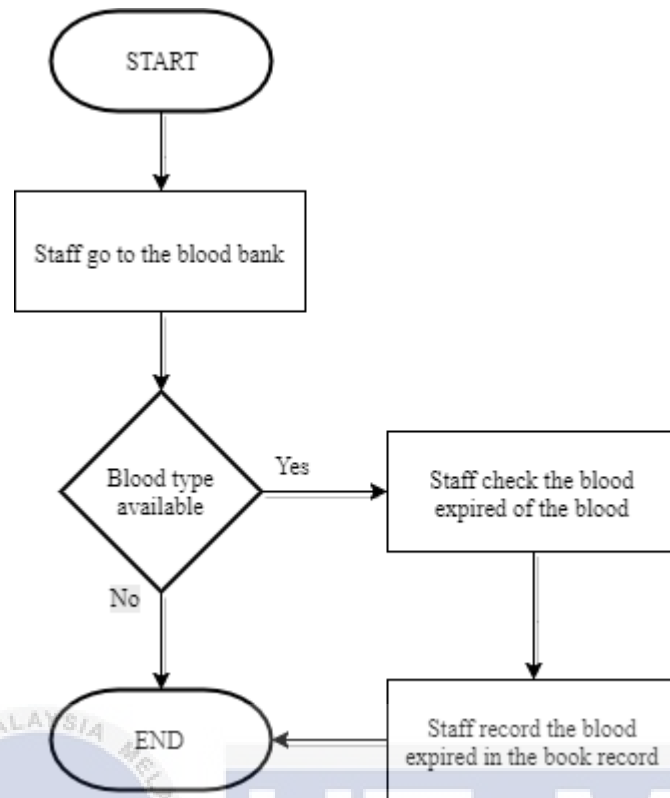
Figure 3.4 Current Blood Expired Management Flow Chart

3.3 Requirement Analysis of the To-Be-System

There are two important requirements that need to be focus on which are the functional requirement and non-functional requirement.

3.3.1 Functional Requirement

Functional requirement will define the function of the system and its components. Context Diagram and Data Flow Diagram (DFD) from Level 0 to Level 1 will be include for the functional requirement.

3.3.1.1 Context Diagram

Context diagram will show the relationship that the system has with the other external entities. For the BBMS, the entities that will interact with the system will include the manager, staff and the blood bank.
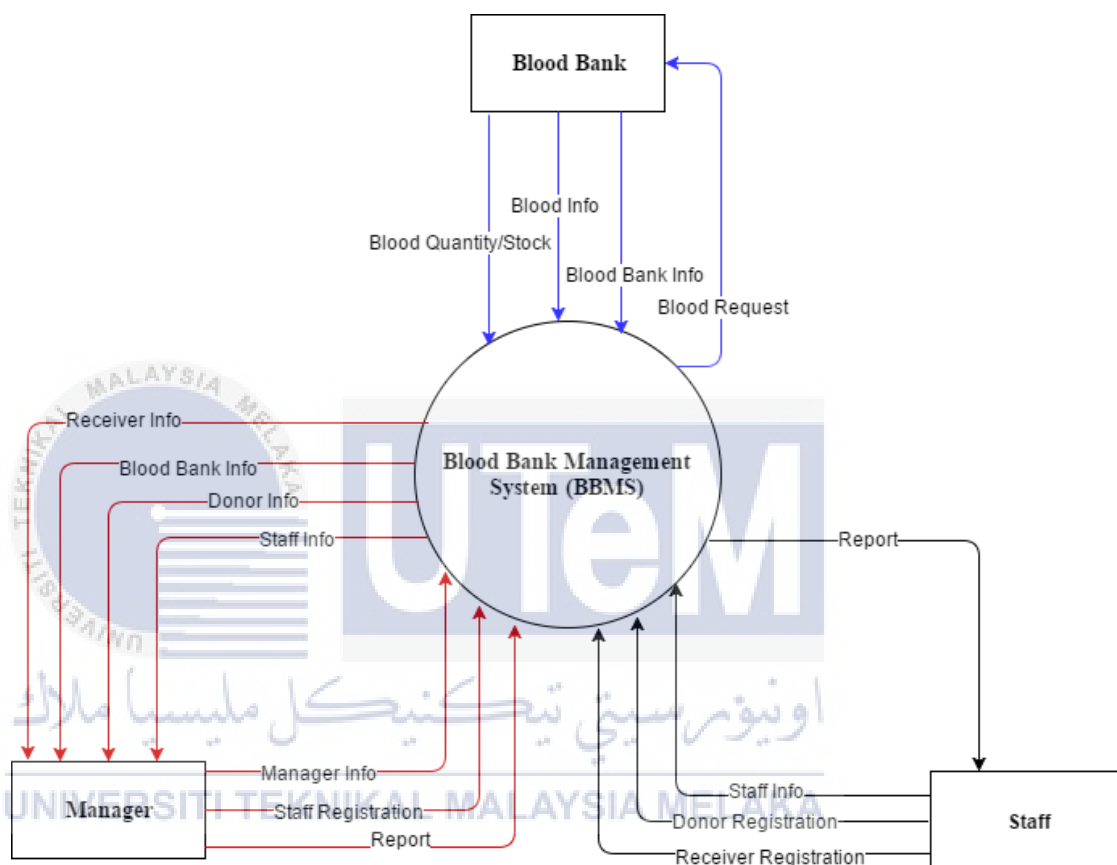
Figure 3.5 Context Diagram of BBMS

3.3.1.2 Data Flow Diagram (DFD)

Data Flow Diagram of BBMS will show the flow of information in this system. This DFD will provide an overview of the system. There are two level for BBMS which are Level 0 and Level 1.
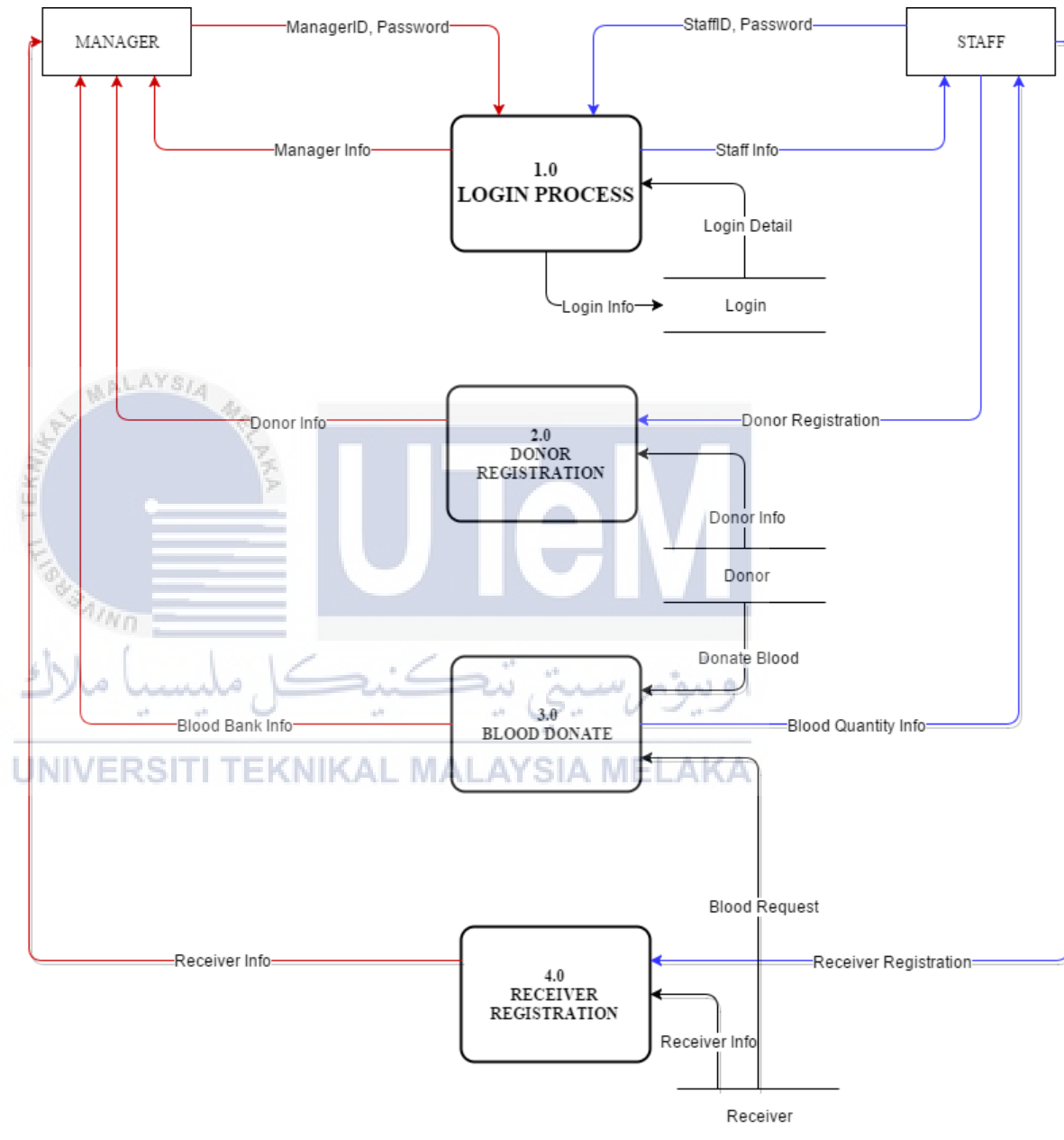
Figure 3.6 DFD of BBMS Level 0
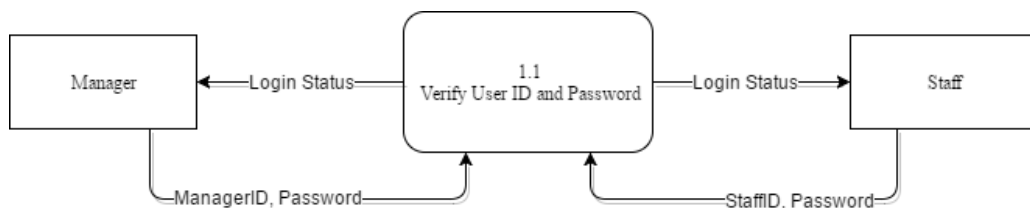
Figure 3.7 DFD for Login Process Level 1

Process Description

Table 3.1 Process Description of DFD for Login Process Level 1

Process	Login
Purpose	Process of login into the system
Description	This process has one sub-process involved which is the process of verification of user ID and password. This process involving inserting the user ID and password.
Responsibility	Manager and staff
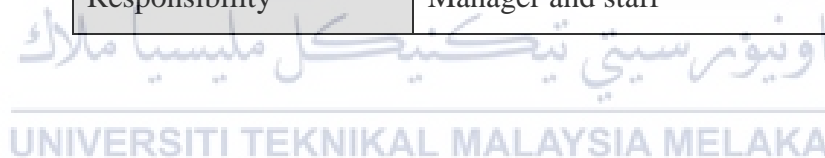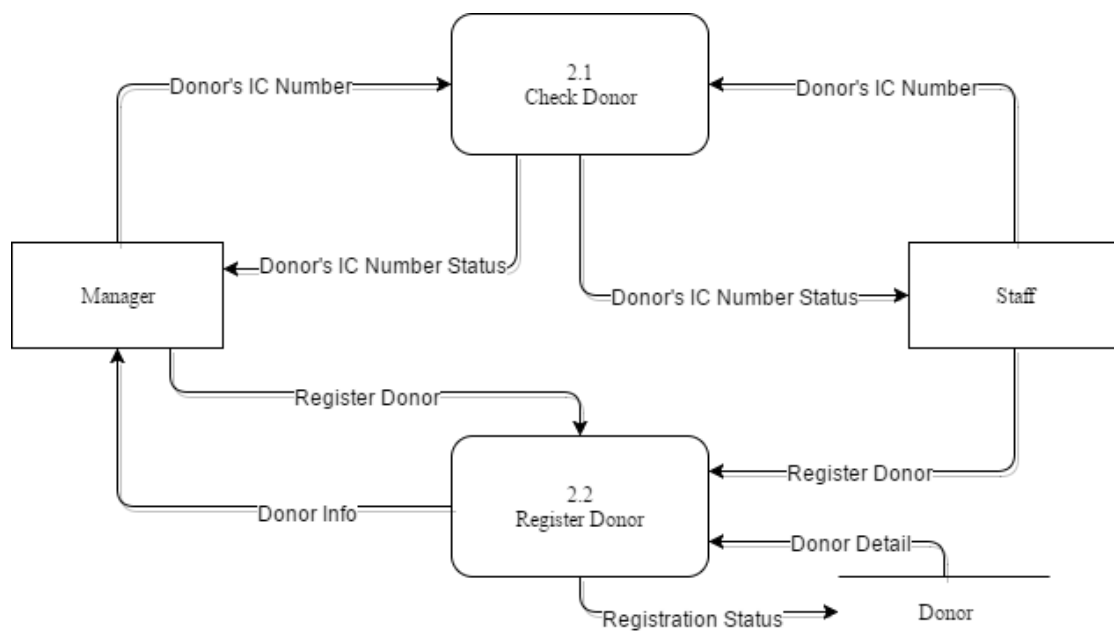
Figure 3.8 DFD for Donor Registration Level 1

Process Description

Table 3.2 Process Description of DFD for Donor Registration Level 1

Process	Donor Registration
Purpose	Process to register new donor
Description	This process have two sub-process involved: 1. Donor checking: to check either the donor has been registered as a donor or not. This process involve inserting the IC number of the donor. 2. Donor registration: register the non-registered donor as a new donor into the system. This process involve inserting the detail of the new donor.
Responsibility	Manager and staff
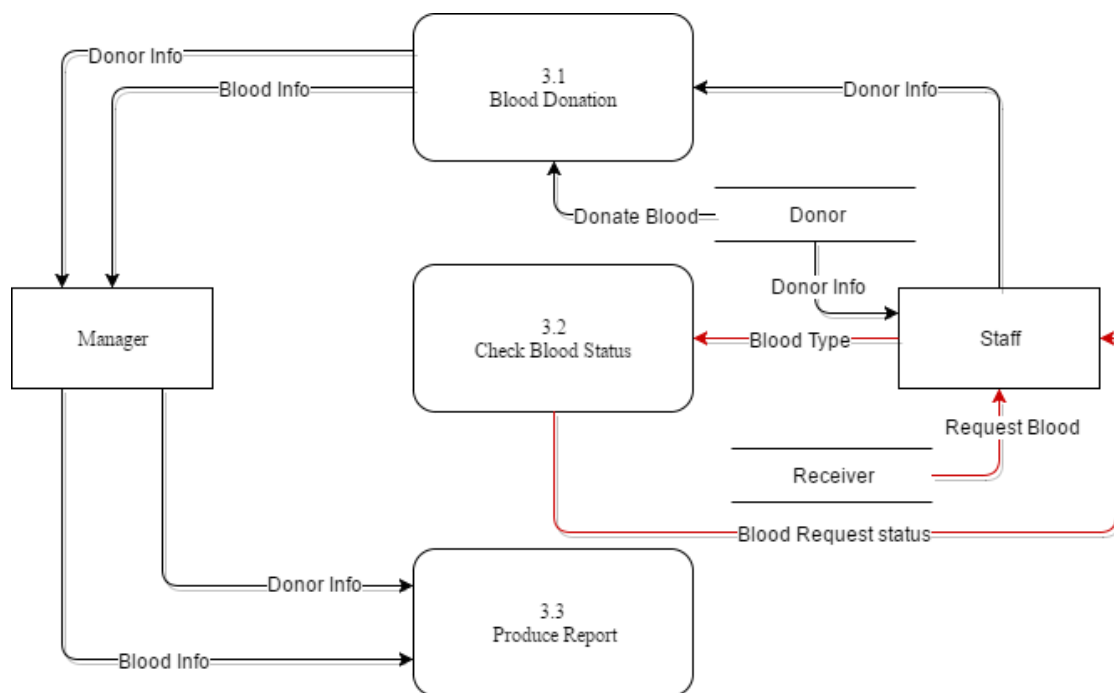
Figure 3.9 DFD for Blood Donation Level 1

Process Description

Table 3.3 Process Description of DFD for Blood Donation Level 1

Process	Blood Donation
Purpose	Process of blood donated by the donor
Description	This process involved three sub-process which are: 1. Blood donation: the process where the registered donor want to make a new blood donation. This process involve the activity of inserting donor IC number and assign the blood donated by them as their blood automatically by system. 2. Check blood status: this process is the process when there are request form the receiver for blood. This process will involve the process of inserting the blood type of the receiver and match with the current blood that have in the stock. 3. Produce report: this process will generate report for the blood donated by the donor and the blood received by the receiver.
Responsibility	Manager and staff

Figure 3.10 DFD for Receiver Registration Level 1

Process Description

Table 3.4 Process Description of DFD for Receiver Registration Level 1

Process	Receiver Registration
Purpose	Process of registration for a new receiver of blood
Description	This process involved two sub-process which are: 1. Checking receiver: this process is to check either the receiver had been registered in the system or not. This process involve inserting the receiver IC number to check. 2. Register receiver: process of registering the new receiver of blood. This process involve inserting the receiver detail.
Responsibility	Manager and staff

3.3.2 Non-Functional Requirement

Non-functional requirement is a group of requirement that describes a variety of system characteristics to attain user's higher satisfaction toward the system. The non-functional requirement for the BBMS are including the performance, security, usability and the safety of the system.

Table 3.5 Non-functional Requirement for BBMS

No	Non-functional Requirement	Description
1	Performance	• The system need to be reliable and fast. • If unable to process the request then pop out appropriate error message. • Web pages are loaded within few seconds. • Easy tracking of blood records and updating of blood quantity in the blood bank updated automatically.
2	Security	• The data can be backup and recovery. • The authorization process is done for the user account.
3	Usability	• The system is easy to use, user-friendly interface and efficient way to develop.
4	Safety	• The details of the user are maintained properly. • The database is kept backup. • User account is authorized.

3.3.3 Other Requirements

Other requirements will describe about the software and hardware used while develop the system.

3.3.3.1 Software Requirements

Table 3.6 Software Requirement

NFR No.	Software	Description
SR_01	Oracle 11g	For BBMS, Oracle 11g will be used as the database and Oracle SQL Server as the platform for the use to do database tasks while developing BBMS.
SR_02	XAMPP 3.2.2	XAMPP 3.2.2 is a collection of free software for installing and using the Apache Web server 2.0. Basically, it is an Apache 2.0 distribution that includes the Apache Web server 2.0, MySQL 5.0 and phpMyAdmin 2.10.3. XAMPP 3.2.2 is also to build an easy to install distribution for developers to get into the world of Apache 2.0.
SR_03	Oracle SQL Developer	Graphical user interface that allows database users and administrators to do the database tasks for BBMS.
SR_04	Microsoft Visio 2013	Used to design the Entity-Relationship Diagram (ERD) for the system.
SR_05	Notepad ++	The source code editor that support programming PHP language to develop the interface of BBMS.
SR_06	Microsoft Word 2013	Used to write the report and documentation for the BBMS.
SR_07	Microsoft PowerPoint	Used to prepare the slide show of the BBMS description for presentation.

3.3.3.2 Hardware Requirements

Table 3.7 Hardware Requirement

NFR No.	Hardware	Description
HR_01	Laptop Asus A550C	To be able to view the web based system.
HR_02	Processor	Processor used in the Laptop Asus A550C is i3. The processor is used to process and to avoid any corruption during development phase.
HR_03	Memory	Computer memory is any physical device capable of storing information temporarily or permanently. The memory need to be operated well to avoid slower performance of laptop and any corruption.
HR_04	Hard Dick Transcend (1Terabyte)	To develop this system, I need sufficient storage to store all the data, the GUI and coding for the system. The system also need to be back up the whole system to the hard disc if anything happen to the laptop.

3.4 Conclusion

As a conclusion, the Context Diagram explained the whole system to be developed in general. Meanwhile, the Data Flow Diagram explained more details about the input and output needed for the system and also the database.

The development of techniques of data analysis has helped to understand the structure and meaning of data in organizations. Data analysis techniques can be used as the first step of extrapolating the complexities of the real world into a model that can be held on a computer and be accessed by many users.

CHAPTER IV

4.0 DESIGN

4.1 Introduction

Every system to be developed must be design first. This is because all the requirements of the system can be improved after the analysis design. So that, this system can be more to the specified requirements because we can do the improvement of the system earlier during the design phase. There are three phases during the analysis which will include the conceptual design, physical design and logical design.

Conceptual database design is the first phase of the database design. The Entity-Relationship Diagram (ERD) is developed to portray the structure of the database. The entity relationship model is a way of graphically representing the logical relationship of entities in order to create a database. ERD will determine attribute domain, determine candidate and primary key attributes.

Logical database design is the second phase in the database design methodology. During the logical design phase, a logical architecture is designed showing the interrelationship of the logical components of the database. Data dictionary is a document used to describe a database structure, derive relations for local logical data, validate relations using normalization and validate relations against user transactions. At the same time, integrity constraints is defined and I make review local logical data model with user.

The last phase of database design methodology is physical database which 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.

4.2   System Architecture Design

This topic will define how the system architecture models describe the structure of the system. There are three types of model which are conceptual design, logical design and physical design.

4.3   Database Design

Database design is the process where the data model of the system database is produced in more detail. Data model will include all the physical and logical design to generate a design in a data definition language, which is it will be used in developing the database.
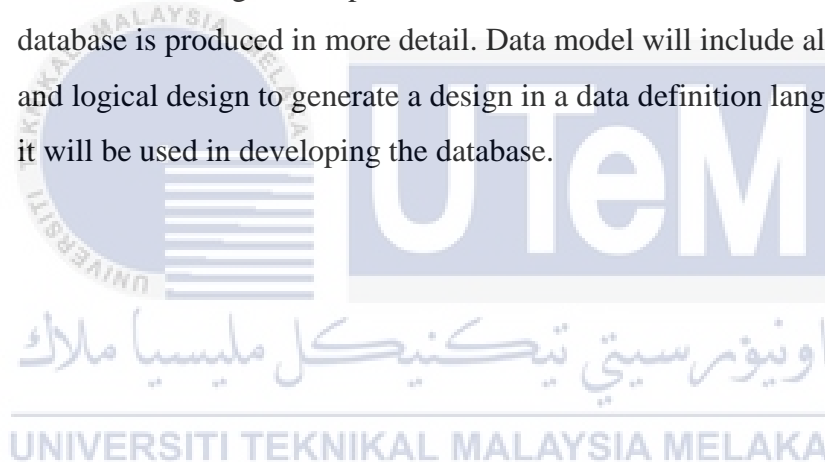
4.3.1 Conceptual Design

The purpose of the conceptual design phase is to build a conceptual model based upon the previously identified requirements, but closer to the final physical model. Entity-Relationship Diagram (ERD) for the BBMS is the first conceptual model that will be referred for the next stage of system development.
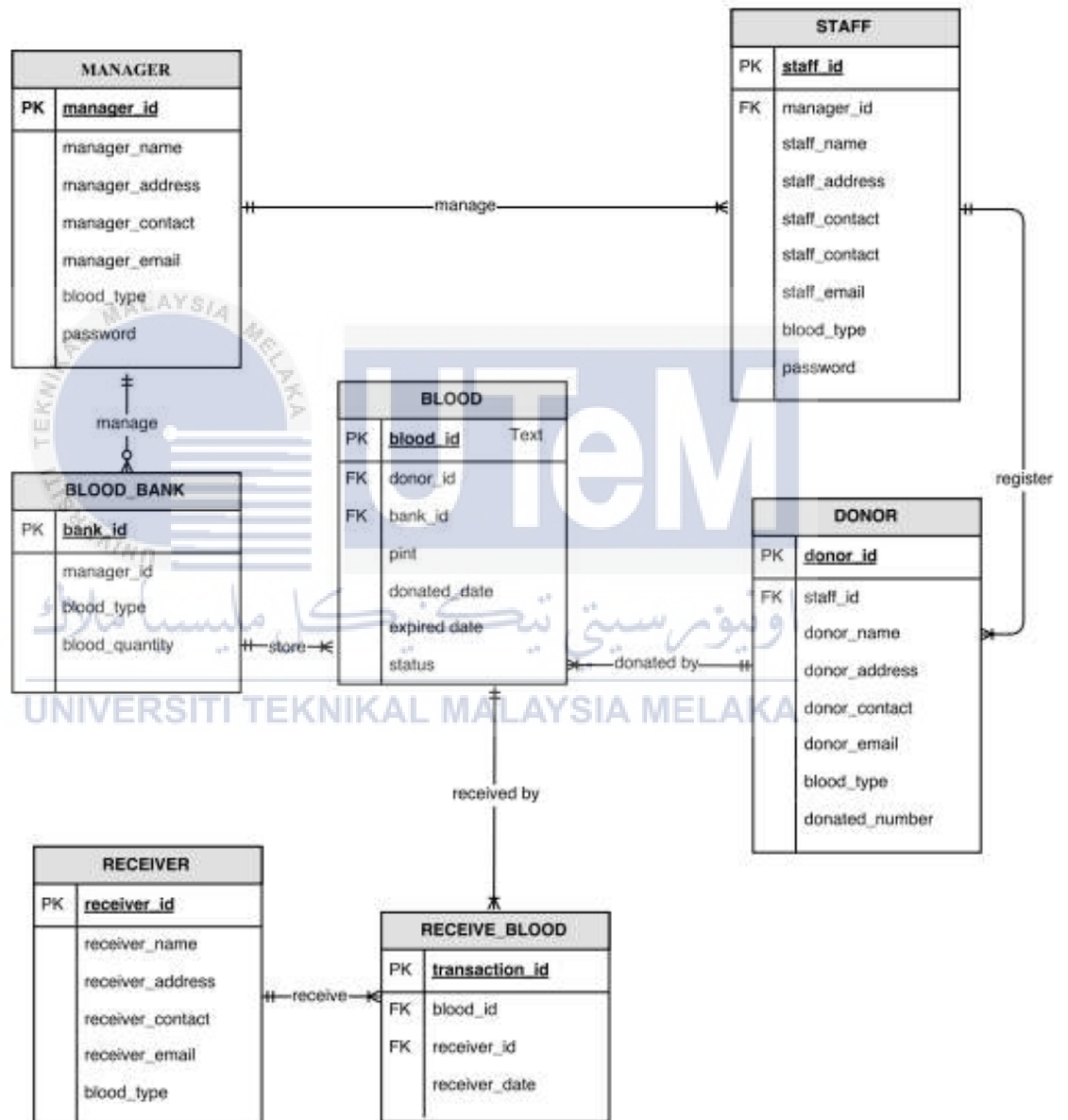
Figure 4.1 Entity-Relationship Diagram (ERD) of BBMS

4.3.1.1 Business Rules

Table 4.1 Business Rule of the BBMS

Role	Rules
Manager	- Manager can register one or many staffs. - Manager can register one or many donors. - Manager can register one or many receivers. - Manager can manage one or many blood bank.
Staff	- Staff only can be registered by one and only one manager. - Staff can register one or many donors. - Staff can register one or more receivers.
Donor	- Donor can be registered by one and only one manager or staff. - Donor can make one or many blood donation.
Blood	- One blood can be donated by one and only donor. - One blood can has one and only one receiver. - One blood must have in one and only one blood bank.
Blood Bank	- One blood bank can only be managed by one manager in one time. - One blood bank can store many blood.
Receiver	- One receiver can be registered by one and only one manager or staff. - One receiver can received one or more blood.

4.3.2 Logical Design

The process of logical design involves arranging data into a series of logical relationships called entities and attributes. An entity represents a chunk of information. In relational databases, an entity often maps to a table. Data dictionary will be describe the database structure, derive relations for local logical data, validate relations using normalization and validate relations against user transactions.

Table 4.2 Data Dictionary for Table Manager

Table	Attributes	Datatype	Constraint	Reference Table
Manager	manager_ID	varchar2(4)	Primary Key	
	manager_name	varchar2(50)		
	manager_address	varchar2(100)		
	manager _contact	varchar2(12)		
	manager _email	varchar2(100)		
	blood_type	varchar2(3)		
	password	varchar2(20)		

Table 4.3 Data Dictionary for Table Staff

Table	Attributes	Datatype	Constraint	Reference Table
Staff	staff_ID	varchar2(4)	Primary key	
	manager_id	varchar2(4)	Foreign key	Manager
	staff_name	varchar2(50)		
	staff_address	varchar2(100)		
	staff_contact	varchar2(12)		
	staff_email	varchar2(100)		
	blood_type	varchar2(3)		
	password	varchar2(20)		

Table 4.4 Data Dictionary for Table Donor

Table	Attributes	Datatype	Constraint	Reference Table
Donor	donor_ID	varchar2(12)	Primary key	
	staff_id	varchar2(4)	Foreign key	Staff
	donor _name	varchar2(50)		
	donor _address	varchar2(100)		
	donor _contact	varchar2(12)		
	donor _email	varchar2(100)		
	blood_type	varchar2(3)		
	donated_number	Number(7)		

Table 4.5 Data Dictionary for Table Blood

Table	Attributes	Datatype	Constraint	Reference Table
Blood	blood_ID	varchar2(4)	Primary key	
	donor_ID	varchar2(12)	Foreign key	Donor
	pint	varchar2(3)		
	donation_date	date		
	expired_date	date		
	status	Varchar2(6)		
	bank_ID	varchar2(6)	Foreign key	blood_bank

Table 4.6 Data Dictionary for Table Blood Bank

Table	Attributes	Datatype	Constraint	Reference Table
Blood_bank	bank_ID	varchar2(6)	Primary key	
	bood_type	varchar2(3)		
	blood_quantity	number(7)		
	manager_ID	varchar2(4)	Foreign Key	Manager

Table 4.7 Data Dictionary for Table Receiver

Table	Attributes	Datatype	Constraint	Reference Table
Receiver	receiver_ID	varchar2(12)	Primary key	
	receiver _name	varchar2(50)		
	receiver _address	varchar2(100)		
	receiver _contact	varchar2(12)		
	receiver _email	varchar2(100)		
	blood_type	varchar2(3)		

Table 4.8 Data Dictionary for Table Receive Blood

Table	Attributes	Datatype	Constraint	Reference Table
Receive_blood	transaction_ID	number(7)	Primary key	
	receiver _ID	varchar2(12)	Foreign key	Receiver
	blood_ID	Number(7)	Foreign key	Blood
	receive_date	date		

4.3.3 Physical Design

Physical database design translates the logical data model into a set of SQL statements that define the database. For relational database systems, it is relatively easy to translate from a logical data model into a physical database. Rules for translation is entities become tables in the physical database. The DBMS choose for this system is Oracle. This DBMS is supported by the phpMyAdmin. Oracle has a graphical user interface which easy for developer to perform database tasks. The DBMS used is to design a coding for database task as the planning system. The coding construct on the DBMS such as simple database coding (insert), trigger and stored procedure. The coding built in Oracle SQL Developer can show either have any error or problem at the interface. It also can show the success coding built by showing the output at the interface.

The function of stored procedure is to manage and maintain data easily. Stored procedure coding's built in phpMyAdmin is connected with the selected language that will make the language easy to gain data from the database by using a simple query. The trigger functions almost same as stored procedure but it will run as a unit. Moreover, trigger also function to atomicity generate derived column values.

4.4   Graphical User Interface (GUI) Design

This section will explain the interface of the system. It will include the detail explanation about the function of the system.

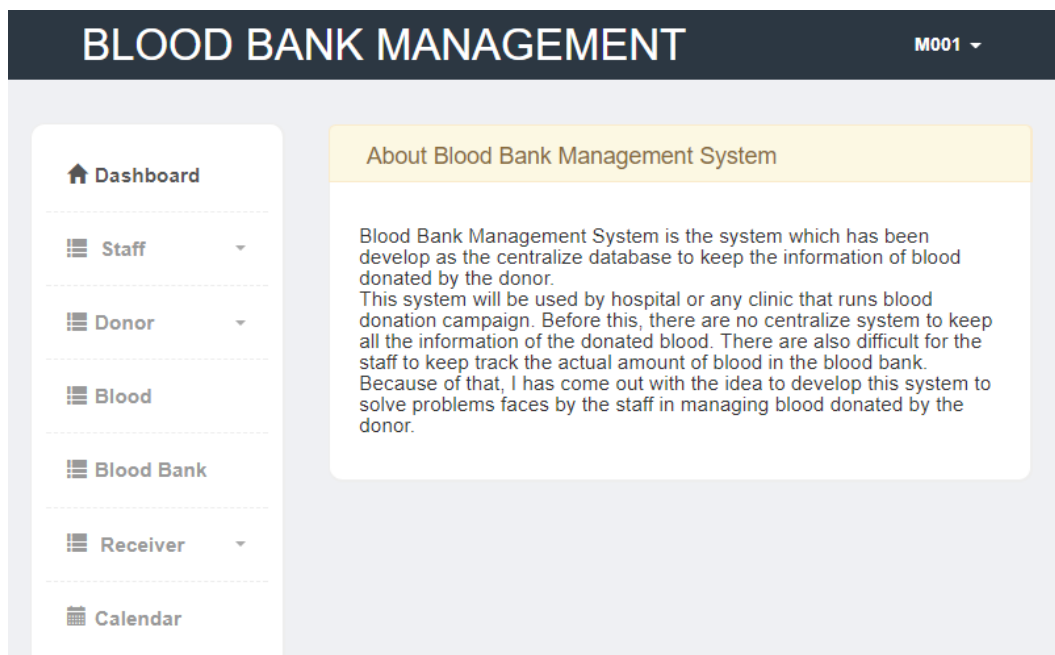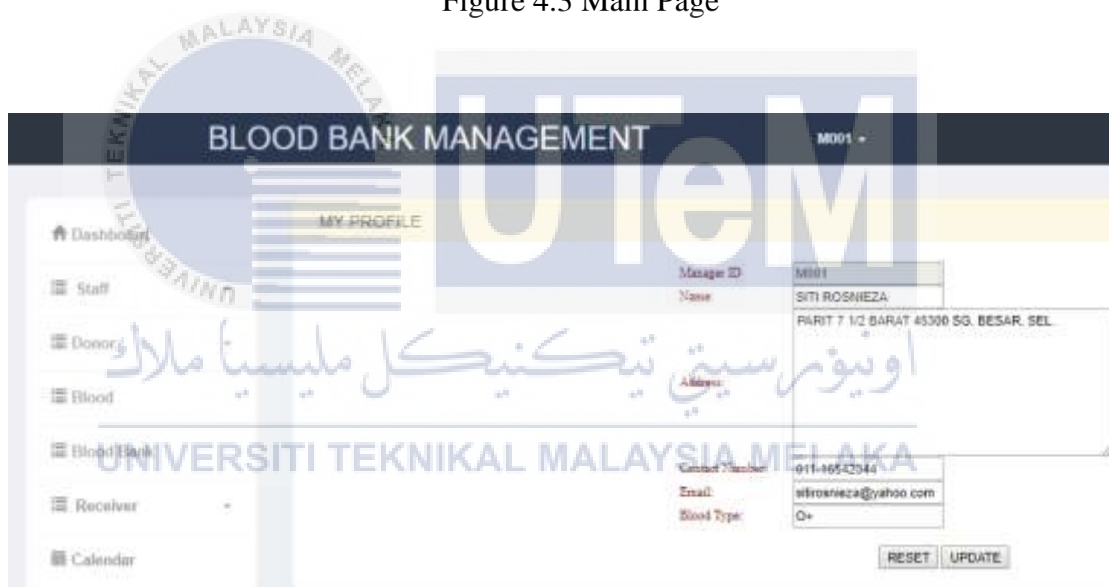Figure 4.2 Login Interface for Admin and Staff

Figure 4.3 Main Page

Figure 4.4 Profile Page

Figure 4.5 Registration Form for New Staff

Figure 4.6 List of Staff

Figure 4.7 Update Staff

Figure 4.8 Delete Staff

Figure 4.9 To Check Donor's IC Number

Figure 4.10 List of Donor

Figure 4.11 Update Donor's Details

Figure 4.12 List of Blood

Figure 4.13 Update Blood Details

Figure 4.14 To Transfer Blood to the Receiver

Figure 4.15 Blood Quantity in Each Blood Bank

Figure 4.16 Search for Blood
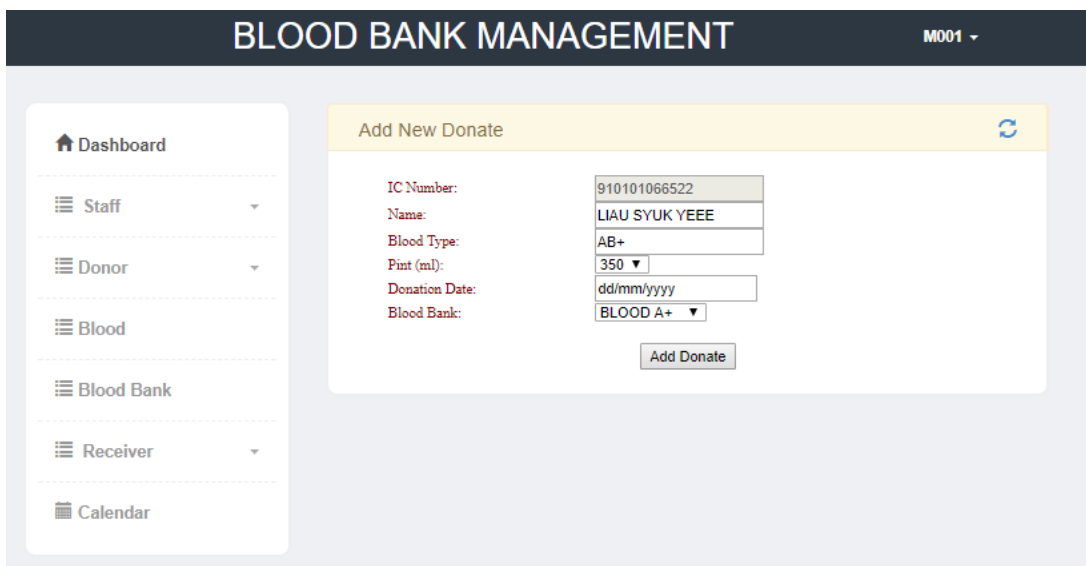
Figure 4.17 Add New Donation Blood
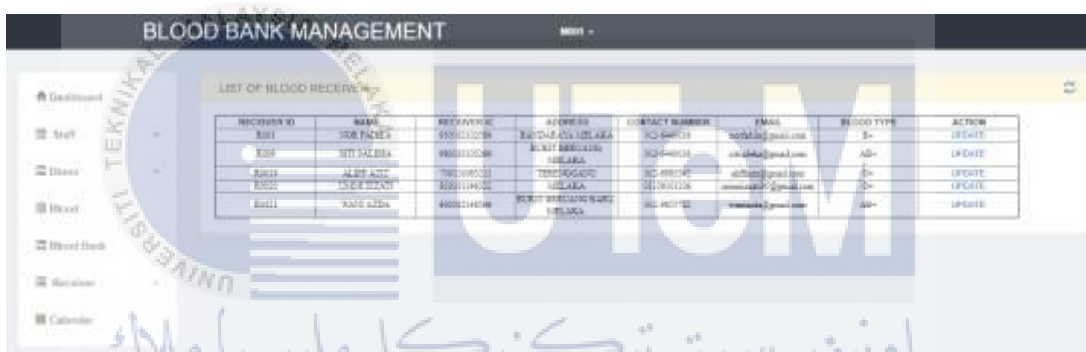
Figure 4.18 List of Blood Receiver

Figure 4.19 Update Receiver Details

4.5   Conclusion

As a conclusion, these three phases of database design is the first step of define how the design of database will be created before develop the system. From this chapter, all the needs for the development of the system can be identified clearly. The database that will be used for the BBMS system is Oracle and Oracle SQL Developer will be the platform of doing task for the database.

CHAPTER V

5.0 IMPLEMENTATION

5.1   Introduction

This chapter will discuss about the implementation of the system. Implementation is the realization of an application, or execution of a plan, idea, model, design, specification, standard, algorithm or policy. In this chapter will include the technical specification and the software components and system through computer programming and deployment.

There are two (2) sections will be discuss which are the software development environment setup and the database implementation. Software development environment setup is set of process and programming tools that will be used to create or develop system. It will include the installation step of database, assigning admin login and other software needed for the development the system including the start-up services. For the database implementation section, it will describe the database creation and database objects creation. To develop BBMS, I have used Oracle as the database and this section will includes Data Design Language (DDL) and Data Control Language (DCL).

5.2   **Software Development Environment Setup**

To develop BBMS, there are a set of software with the programming language will be used to write the program. It must meet with all the user requirement so that it will help the user easier to do the tasks with the system. Blood Bank Management System is identified as two levels. First level is the connection between the user and BBMS system. The connection between them is using user graphical interface (GUI) as the platform for the user to join together so as to provide access and communication with the system and doing their tasks. Notepad++ is the code editor that have been used for writing PHP codes before running to the Google Chrome and produce the GUI.

Level two in the development environment of Blood Bank Management System is between the system and database. It will describe how the system communicate with the database and allow the specific user to access and manipulate data in the database if the user is authorized by the administrator of the Blood Bank Management System. The database used for the Blood Bank Management System is Oracle 11g. Oracle SQL Developer is an Integrated Development Environment (IDE) will be used and act as a platform for working with PL/SQL in Oracle database.

5.2.1   Installation Step

      i.   **Windows 10**

During the phase of Blood Bank Management System development, Windows 10 with the latest version of operating system is used as the platform for the deployment of Blood Bank Management System. Hence, all installation and configuration of software packages are done based on the version of operating system.

     ii.   **XAMPP Server Provider**

To develop Blood Bank Management System, XAMPP Server is needed. The followings are the steps of how to install the XAMPP Server on the laptop. After the installation of XAMPP Server in the laptop, a folder named "BBMS", which signifies "Blood Bank

Management System", is created within the directory of "C:\xampp\htdocs\BBMS".

Step 1: Go to the website to download the XAMPP package.

Figure 5.0 Link to download the XAMPP Provider package.

Step 2: After choose the XAMPP package to download, there will be a pop-up warning shows in the screen about the User Account Control (UAC) being active on your system. Click "**OK**" to continue the installation of XAMPP package on the laptop.
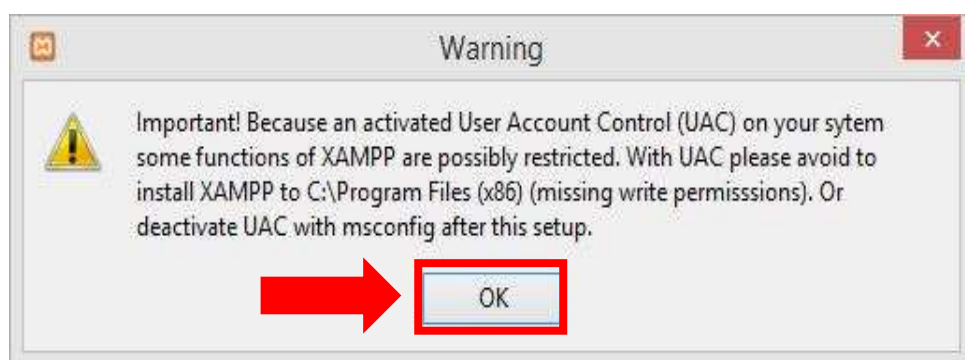
Figure 5.1 User Account Control (UAC)

Step 3: After click the button "**OK**", welcome page to the XAMPP Setup Wizard will show in the screen. Then, click "**Next**" button to continue the installation.

Figure 5.2 Welcome page of XAMPP Setup Wizard

Step 4: After click the "**Next**" button, the Select Components page will appear on the screen. We have to select which components of the software that we like to install. After select all the components that we need, click "**Next**" button on the page to continue. Figure 5.3 shows the components that have been selected.
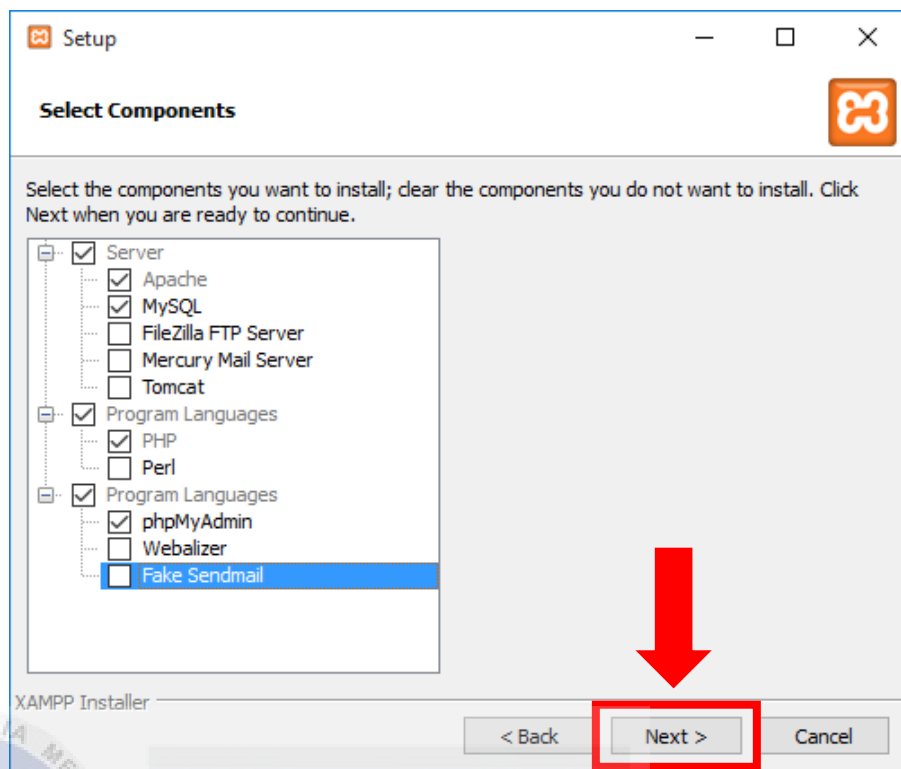
Figure 5.3 Welcome page of XAMPP Setup Wizard

Step 5: The next page in XAMPP Setup Wizard will show the Installation Folder page. In this page, the location folder for the XAMPP should selected. After the location have been selected, click "**Next**" button to continue. Figure 5.4 shows the location folder for the XAMPP Server.
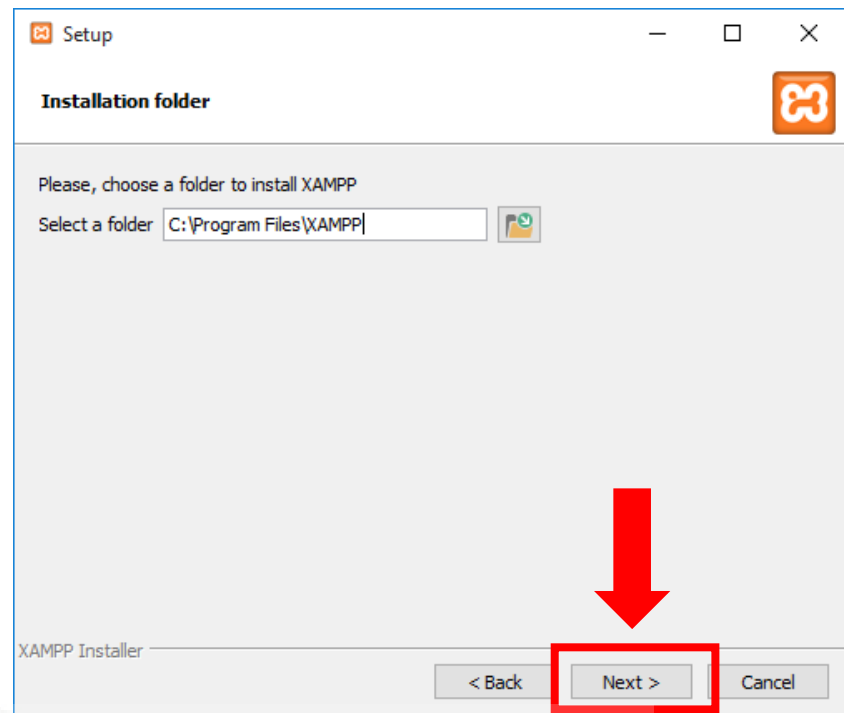
Figure 5.4 Installation Folder

Step 6: The next page will show as in the Figure 5.5. Untick
"*Learn more about Bitnami for XAMPP*" and click "**Next**"
button to continue.
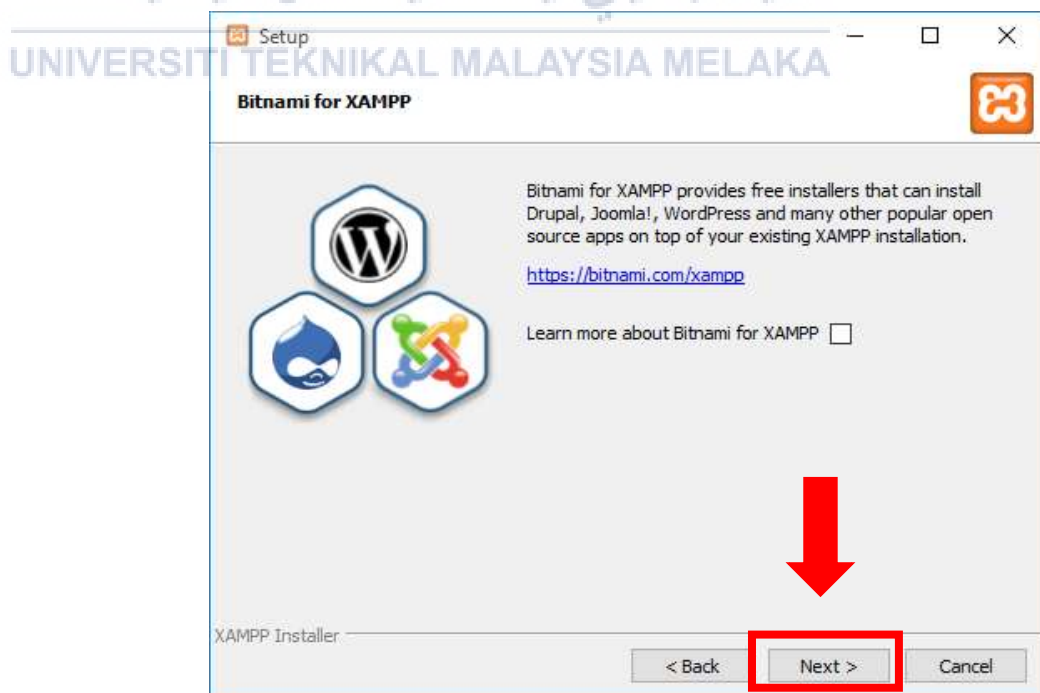
Figure 5.5 Bitnami for XAMPP

Step 7: Now the XAMPP is ready to install into the laptop.
Click "**Next**" button to start the installation.
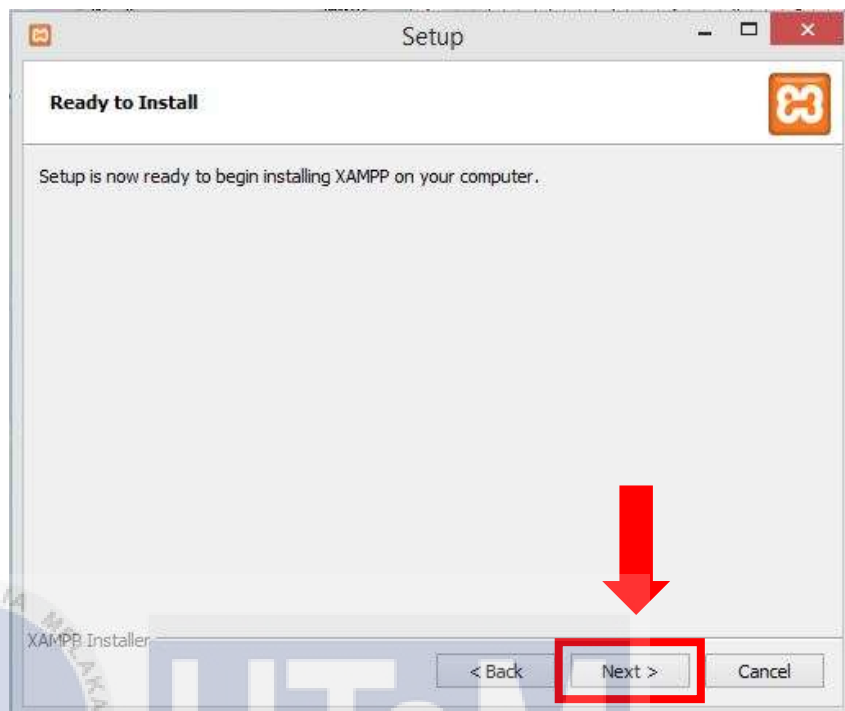
Figure 5.6 XAMPP is ready to install

Step 8: Now XAMP is begin to unpick all the files to the
location that has been selected on the previous step.
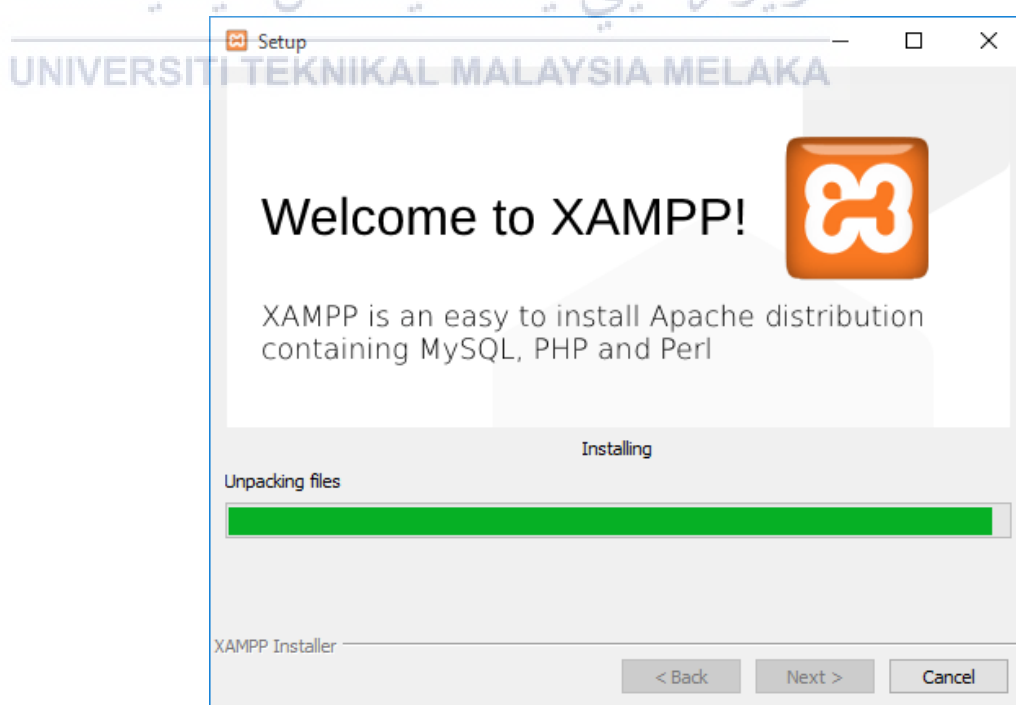
Figure 5.7 Installation process

Step 9: After the XAMPP is successful installed, the page as shown in Figure 5.8 will be display. There are an option that ask if we want to start the control panel. Tick the option and click "**Finish**" button.

Figure 5.8 XAMPP Setup Wizard successfully installed

Step 10: Now the Control Panel will automatically open on the screen. Click the "**Start**" button of Apache and MySQL.
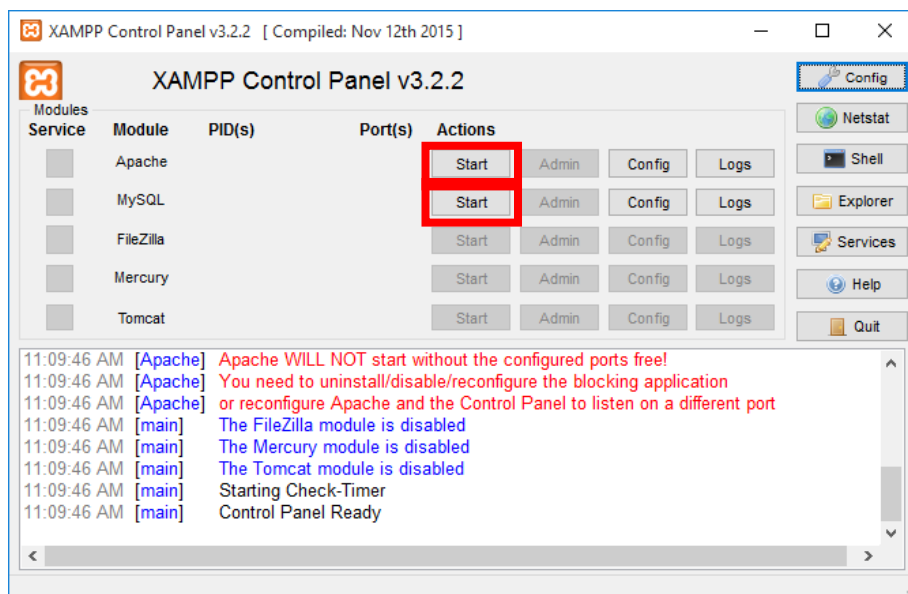
Figure 5.9 XAMPP Control Panel

Step 11: Now, the XAMPP is successfully start. Figure 5.10 show that the Apache is started.
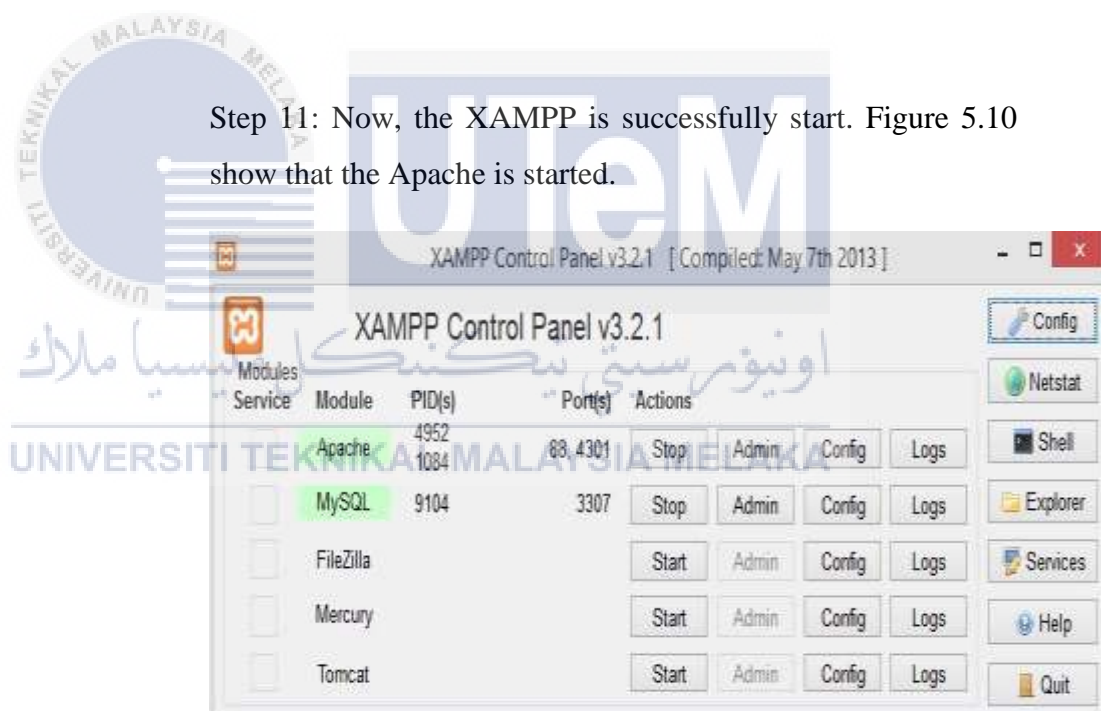
Figure 5.10 Setting of XAMPP

iii.   Oracle 11g

To develop Blood Bank Management System, Oracle Database 11g has been used as the system database. Oracle 11g is one of database type that has the capability of viewing tables back in time, stored a large number of data, and enhanced disaster recovery functions. The followings are the steps on how to install the Oracle 11g and it will used for the development of Blood Bank Management System.

Step 1: Go to the *http://www.oracle.com* and download Oracle Enterprise Edition. Unzipped the folder to start install the Oracle.

Step 2: The location of the folder now is in C:\oracle 11g\database. Click on "**Setup**" to start the installation.
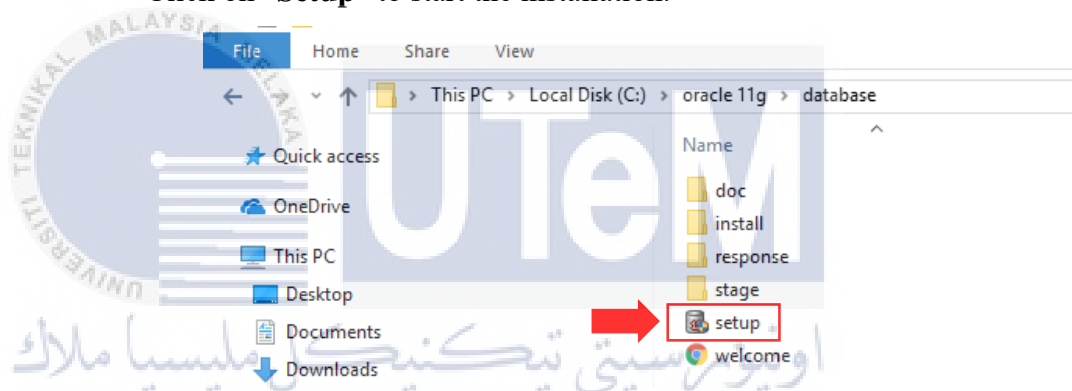
Figure 5.11 Location of Oracle Database setup

Step 3: Initiates the installation

Figure 5.12 Initiates the installation

Step 4: There will be a pop-up display while installing to tell
that the Oracle does not meet the minimum requirements.
Click "**Yes**" button to continue the installation.
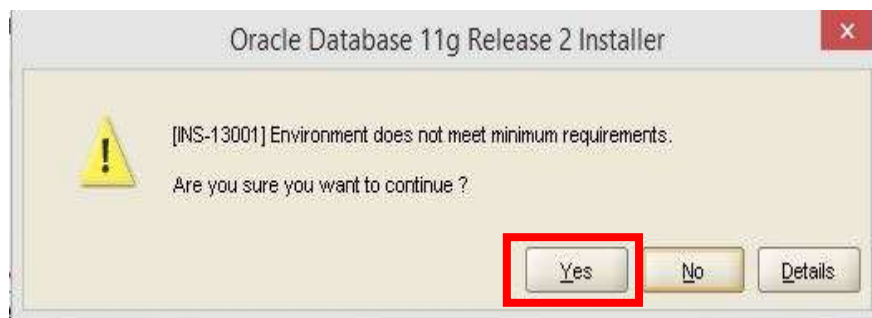
Figure 5.13 Environment variable

Step 5: Next, untick "**I wish to receive security updates via
My Oracle Support**" option. Then, insert email address to
continue the next step. Click "**Next**" to continue.

Figure 5.14 Configuration Security Updates

Step 6: For the first time installation of Oracle database in laptop or personal computer (PC), tick "**Create and configure a database**" option as in the Figure 5.14. Then, click "**Next**" button to continue.
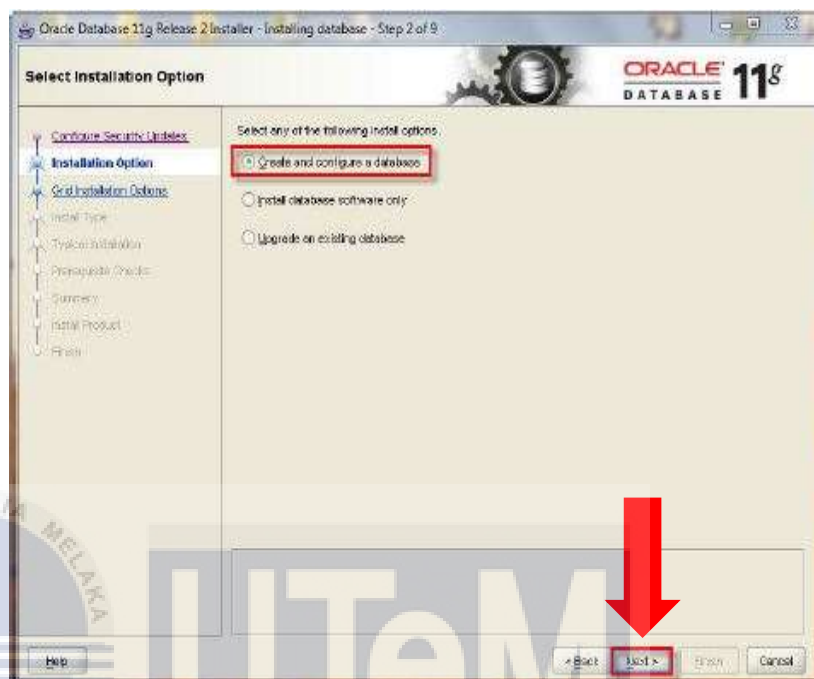
Figure 5.15 Installation Option

Step 7: Next, select "**Desktop Class**" to install the full features of Oracle Database 11g. Then click "**Next**" button to continue for the next step.

Figure 5.16 System Class

Step 8: In this step, set the Global Database Name as orcl as default. For Blood Bank Management System, "orcl123" has been set as the Global Database name. Administrator password also should be set and it will be used to help administrator to login into the database. After filling the details, click "**Next**" button.
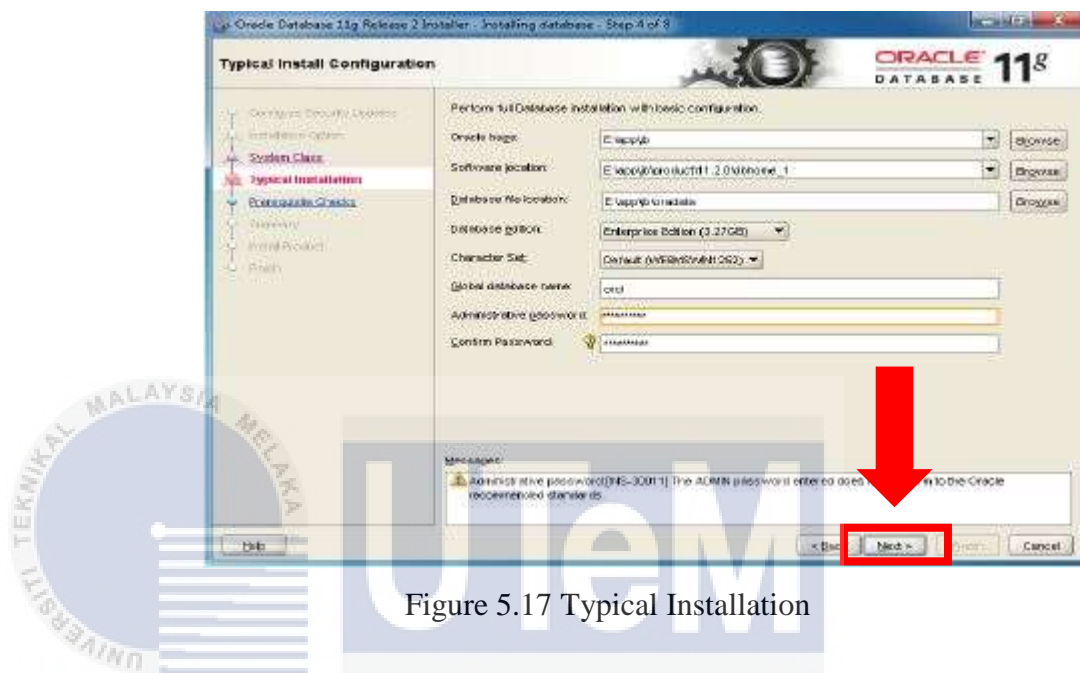
Figure 5.17 Typical Installation

Step 9: Next it steps checks all the prerequisites such as disk space, memory, environment variable and so on, so once it is done we will get next screen automatically.

Figure 5.18 Prerequisite Checks

Step 10: After finish the prerequisite checks, the summary will give the option to save response file to store all the installation related folder information in a text file. Click "**Finish**" to initiate the product installation.
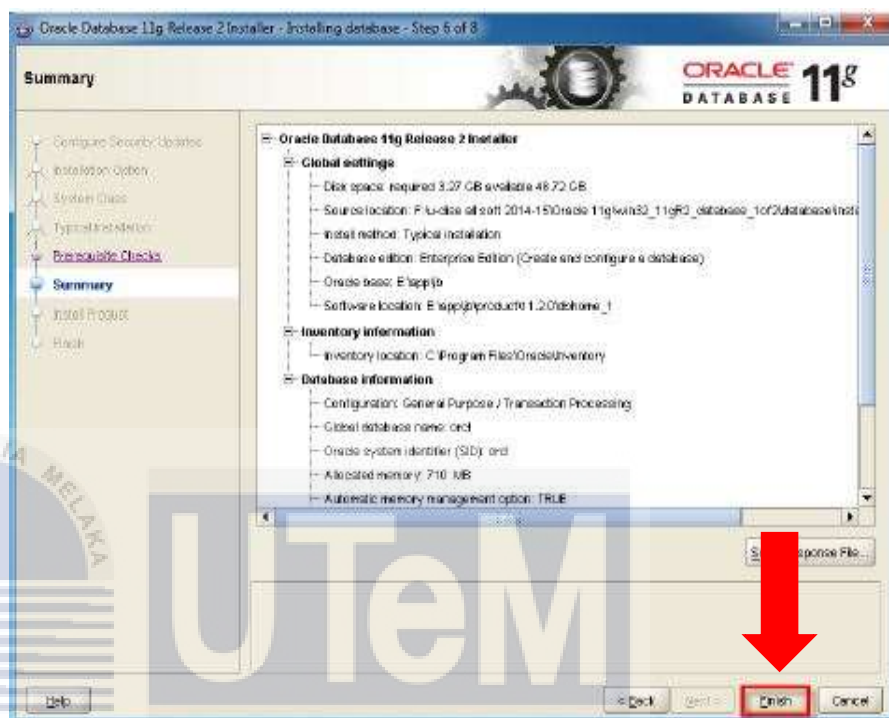
Figure 5.19 Summary

Step 11: Next, the database will start to be installed. It will show the progress of product installation including the copying files to the target folder and setup files.
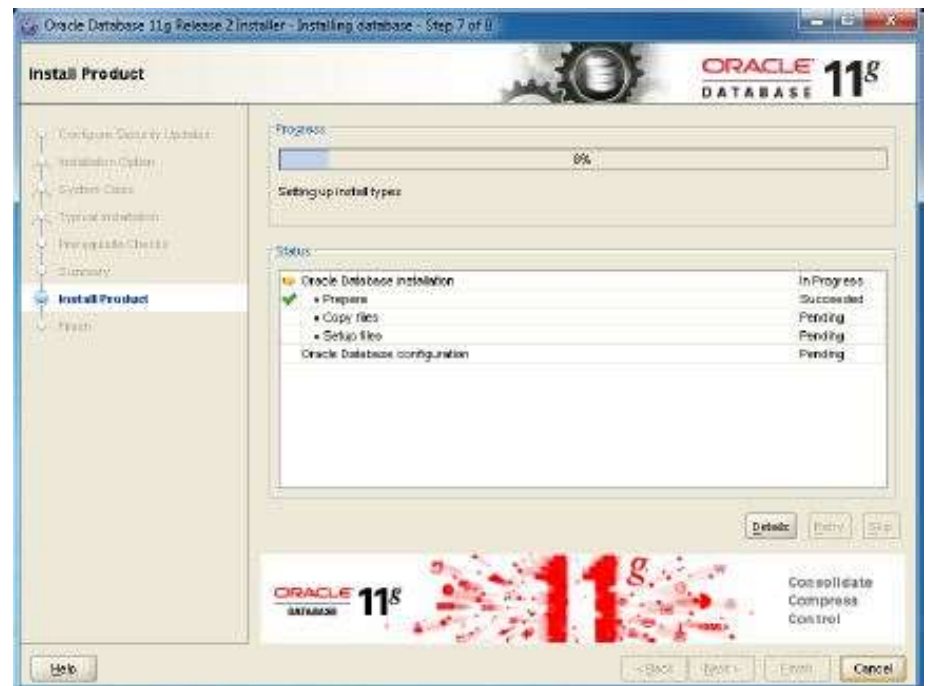
Figure 5.20 Install Product

Step 12: During the installation, there will be a pop-up on scree
as shown in Figure 5.20. Click "**Allow Access**" to continue the
installation. The progress of product installation will
automatically continue.
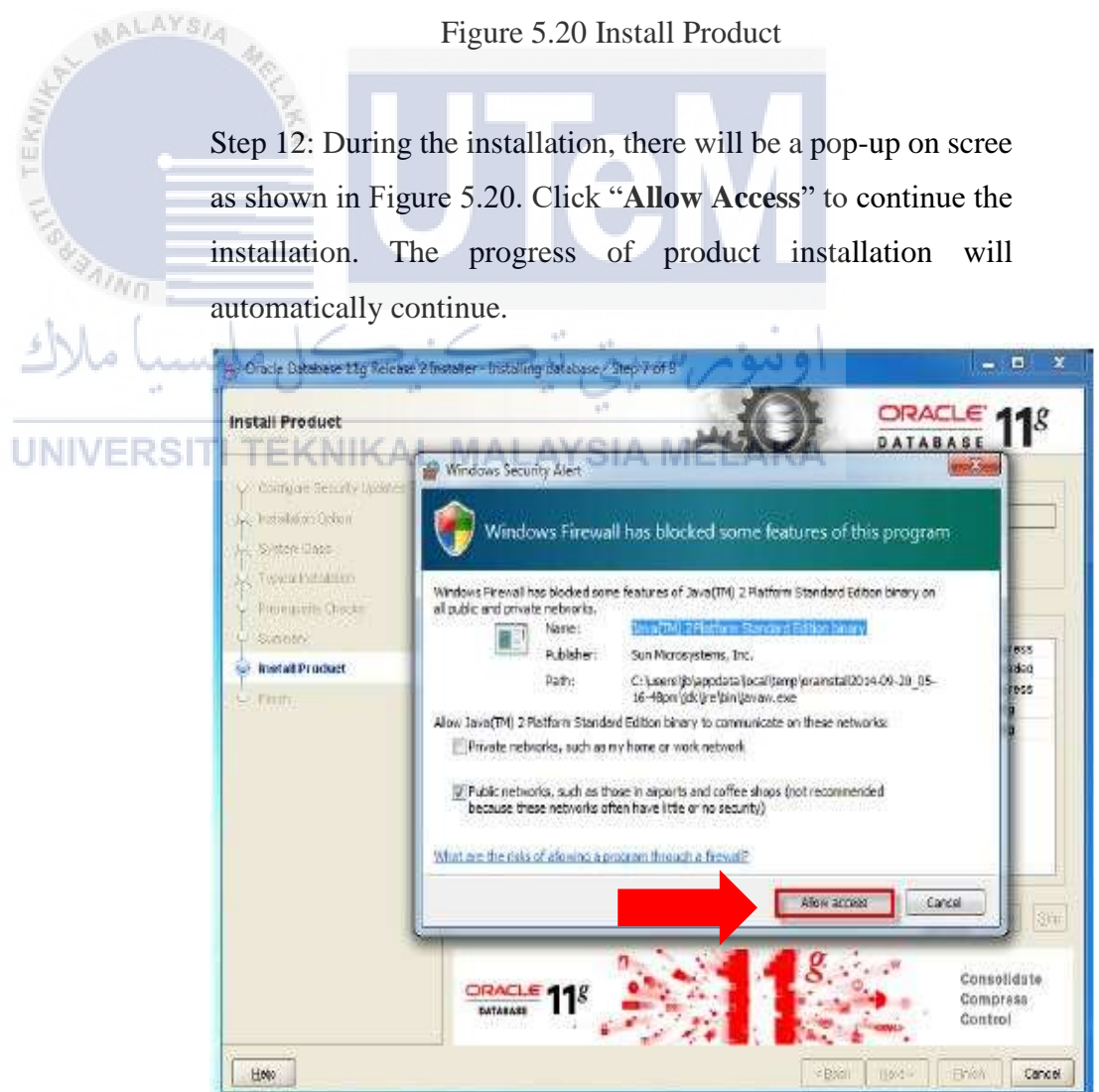
Figure 5.21 Window Security Alert

Step 13: After the progress is completed, it will start copying database files, creating and starting Oracle Instance and completing database creation.
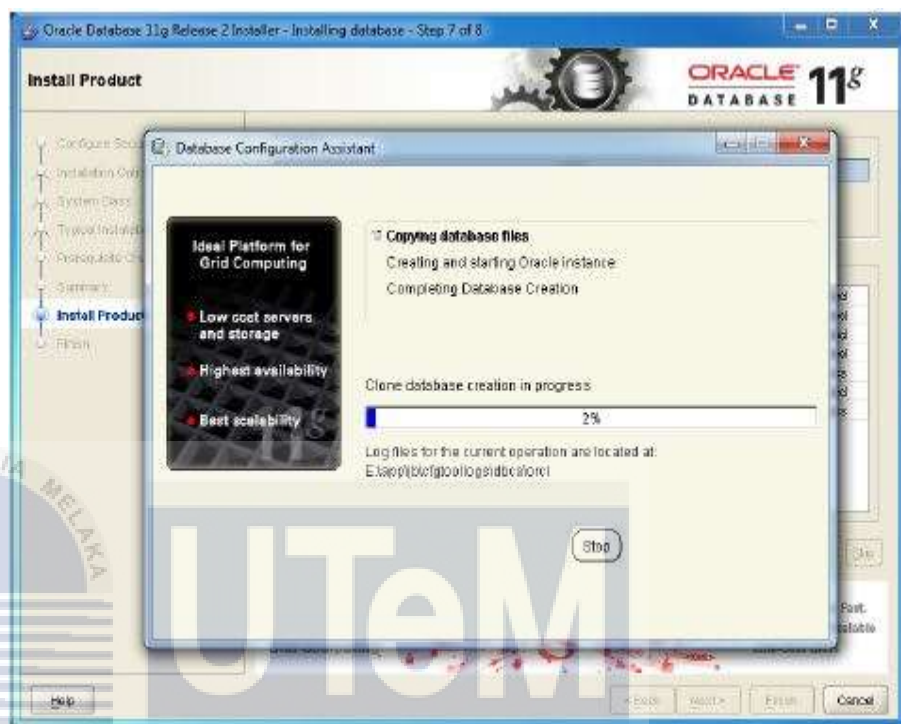
Figure 5.22 Database Configuration Assistant

Step 14: Click "**OK**" after select "**Password Management**".
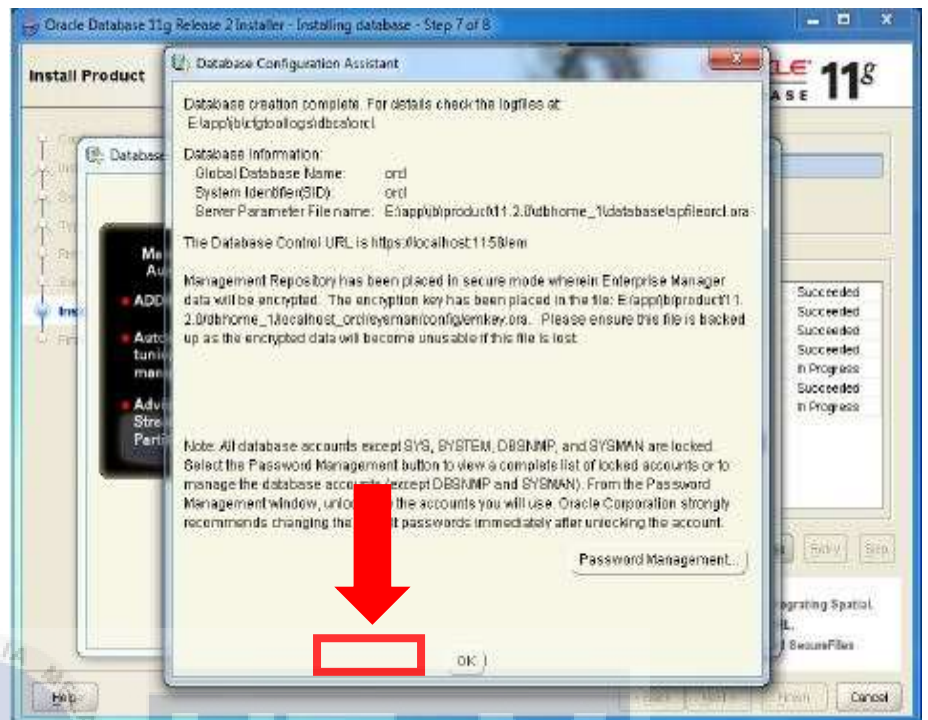
Figure 5.23 Password Management

Step 15: When the installation and configuration is completed, click "**Close**".
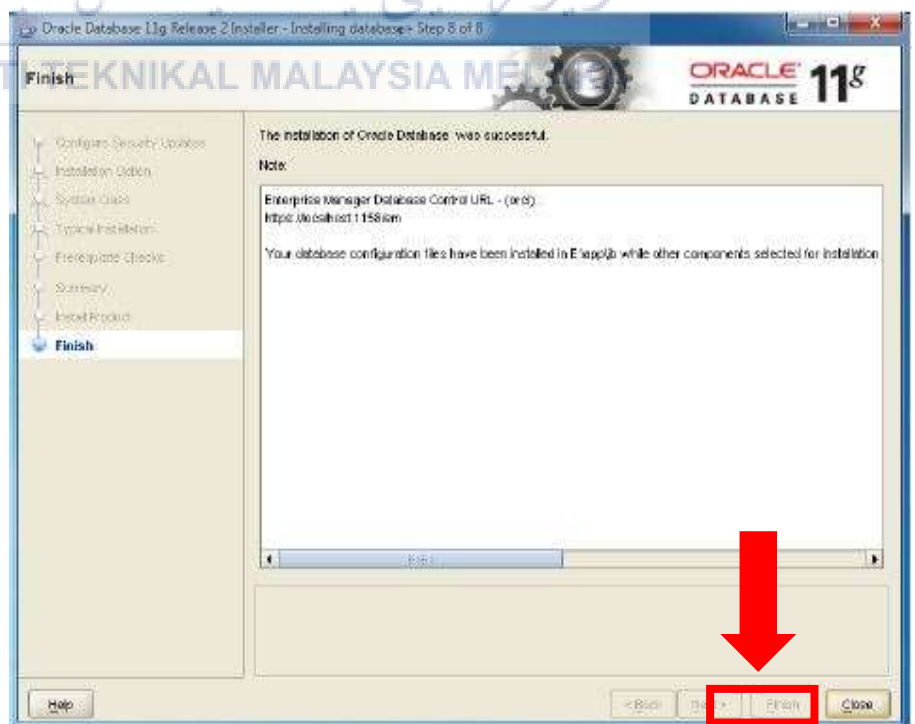
Figure 5.24 Complete Installation

iv. **Create Listener in Oracle 11g**

After finish the Installation of Oracle database, listener must be created. Oracle 11g need Oracle Net Listener which is it will runs on the database process computer. Without listener, database cannot receives incoming client connection requests to connect with the database. Listener also act to manage the traffic of client requests to the database server. The followings are step on how to configure the listener to accept client connections.

Step 1: Open Database Configuration Assistant in Oracle that has been installed and click "**Next**" to create the listener.
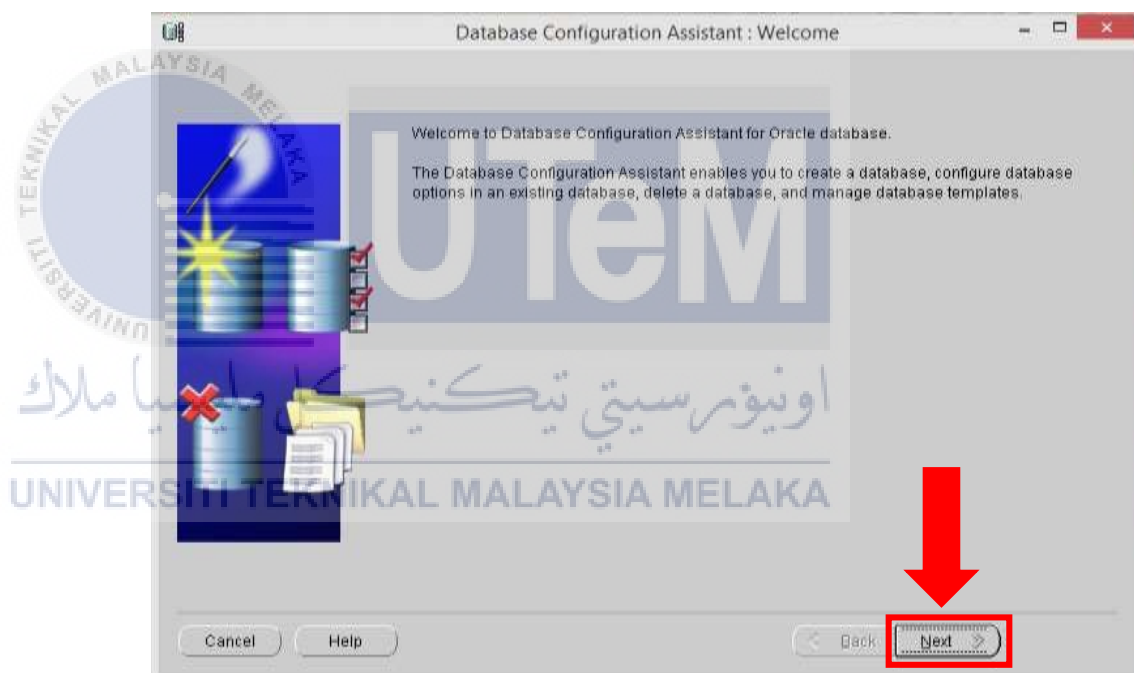
Figure 5.25 Database Configuration Assistant

Step 2: After that, select the operation that we want to perform. Since for Blood Bank Management System is to create the new database, tick "**Create Database**" and click "**Next**" to continue.
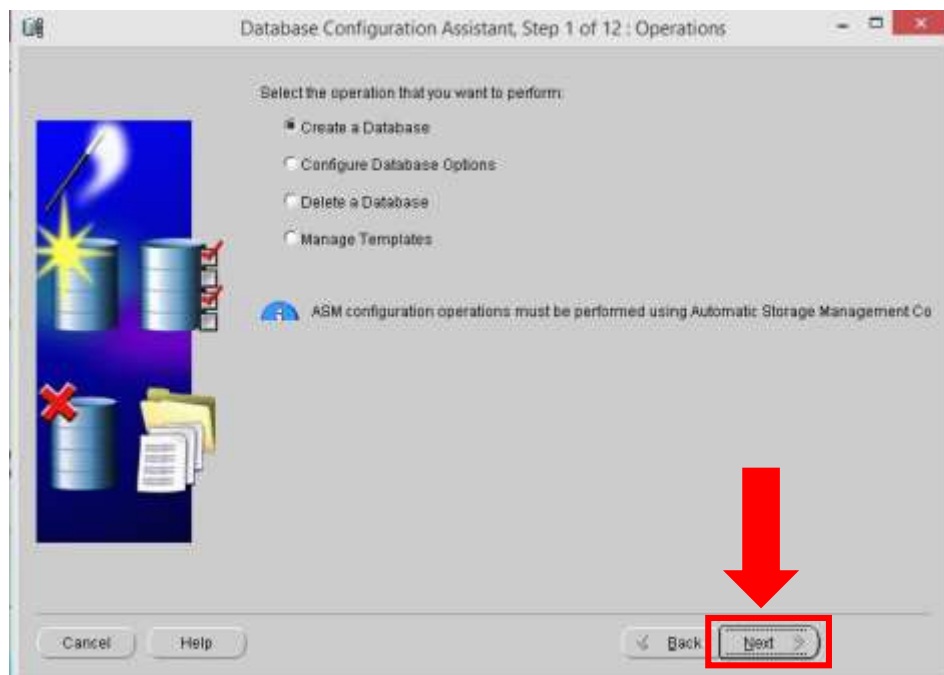
Figure 5.26 Create Database

Step 3: On the next page, it will show the templates that include data files contains pre-created database. Tick "**General Purpose or Transaction Processing**" as your template. Then, click "**Next**" button.
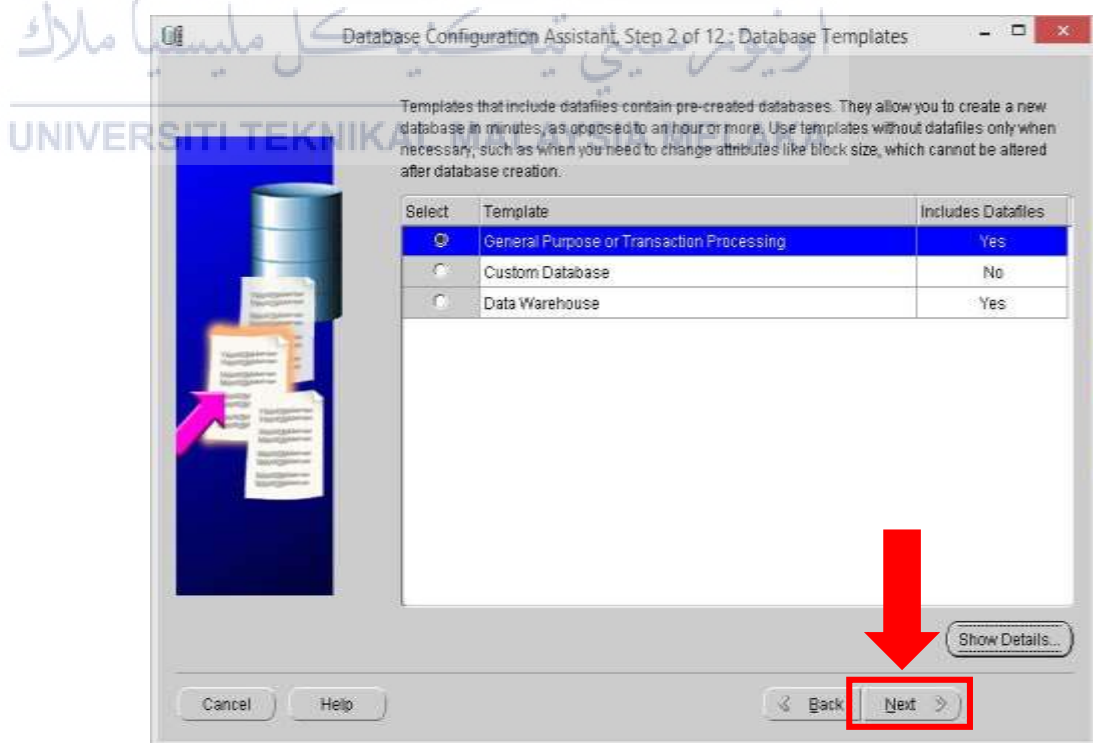
Figure 5.27 Database Template

Step 4: On this step, insert SID as same with the Global Database Name. The Global Database Name for Blood Bank Management System is "**orcl123**". Then, click "**Next**" to continue.
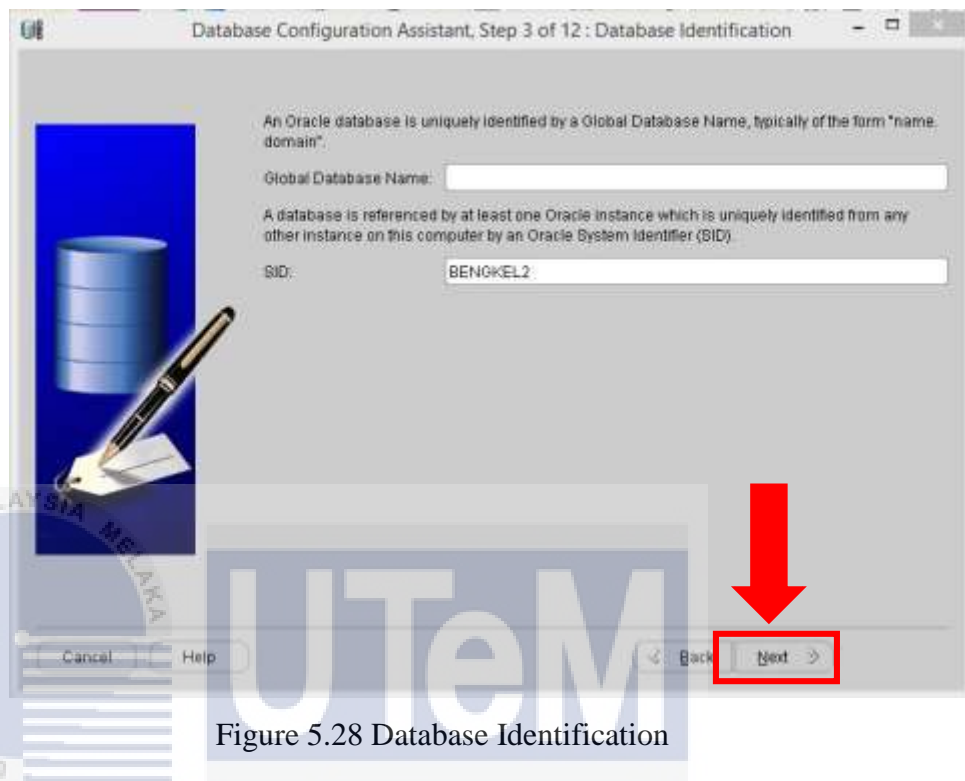
Figure 5.28 Database Identification

Step 5: This page is about the Management Option, click "**Next**" to continue.
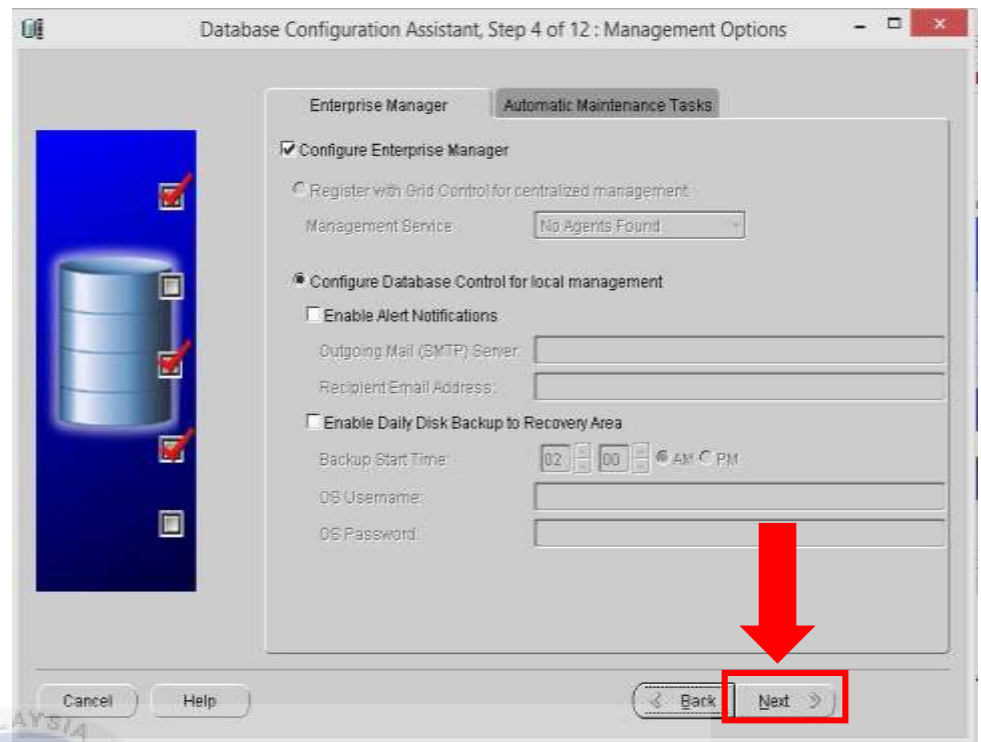
Figure 5.29 Management Option

Step 6: For security reasons, the password need to specify in this page. Select the **"User the Same Administrative Password for All Accounts***"* to insert the password. Then, click "**Next**" to continue.
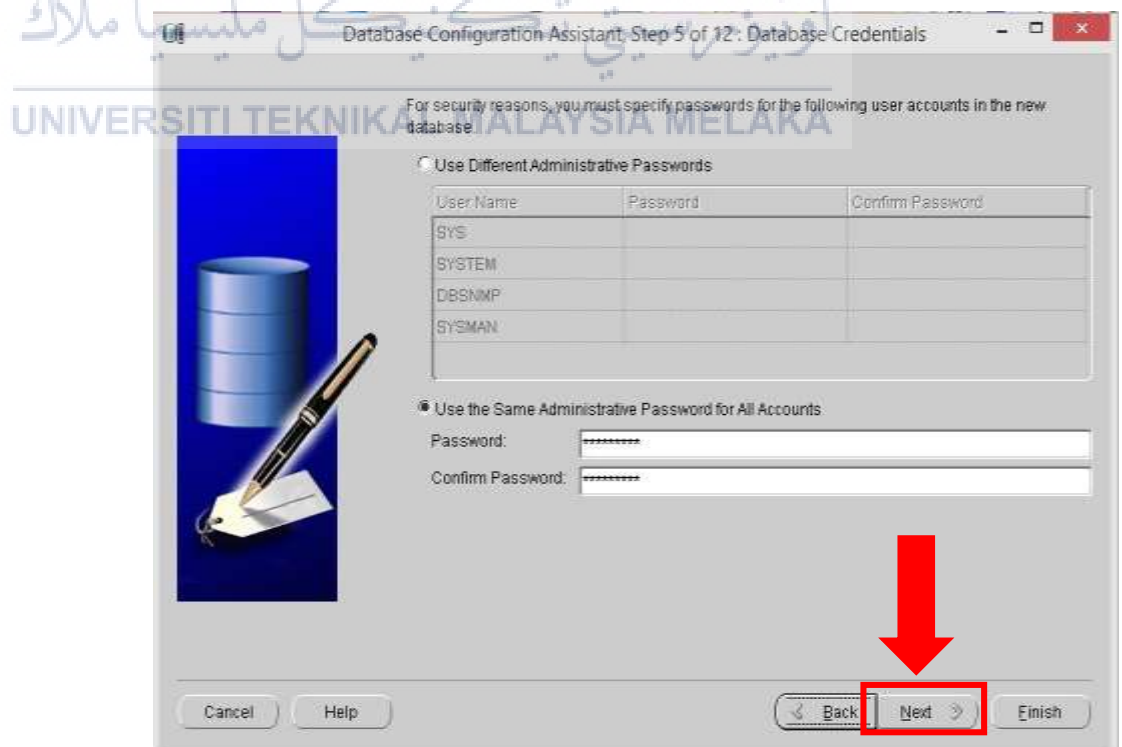
Figure 5.30 Database Credentials

Step 7: Next, tick "**Use Database File Location from Template**"
for the file location and then click "**Next**".
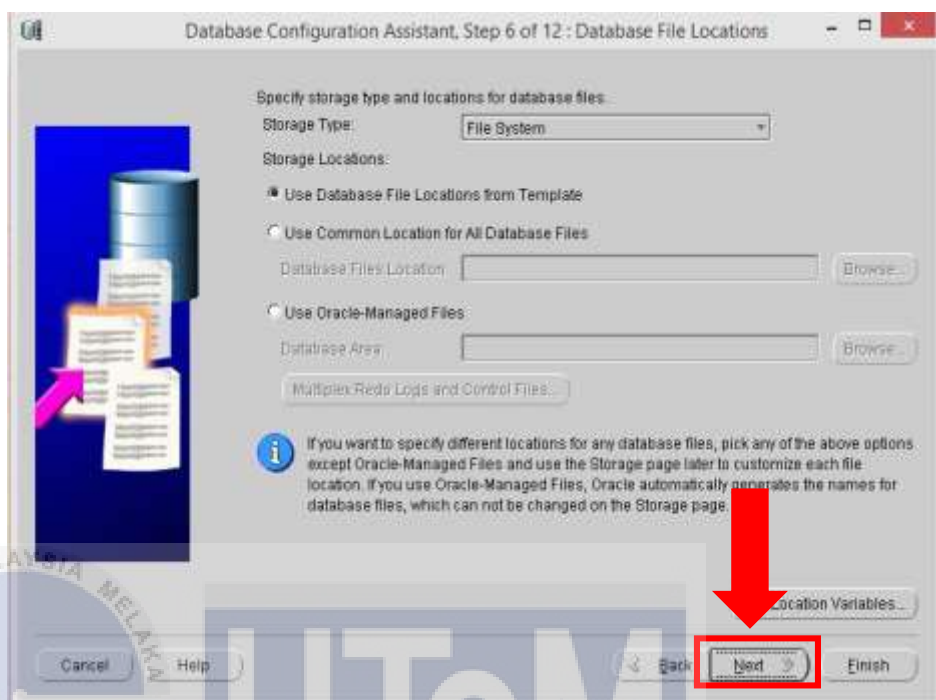
Figure 5.31 Database File Locations

Step 8: Next, on the recovery configuration, tick "**Enable Archiving**" and then click "**Next**" to continue.
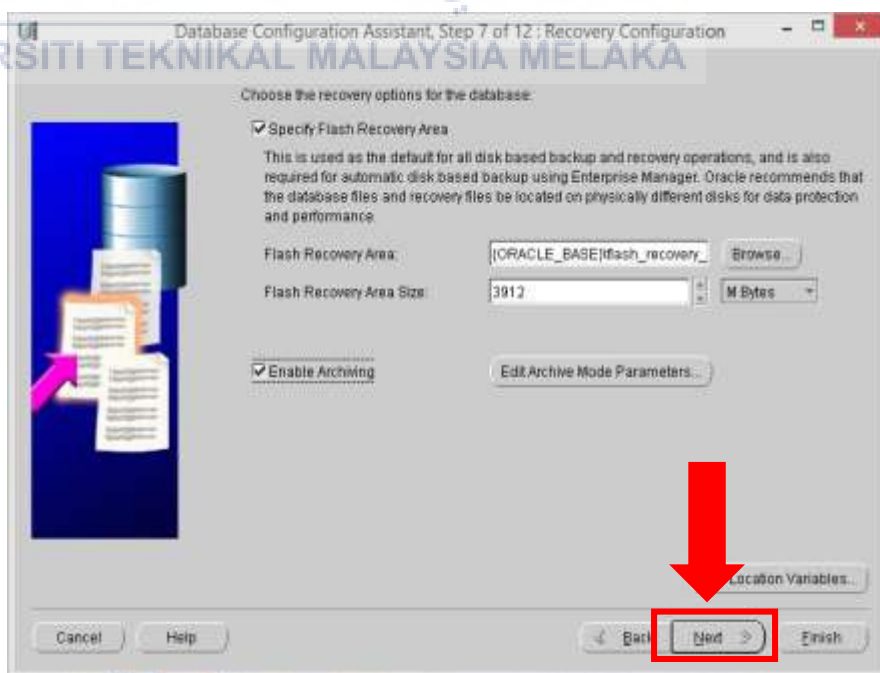
Figure 5.32 Recovery Configuration

Step 9: Next page will show the Database Content, as shown in Figure 5.32. Click "**Next**" to continue.
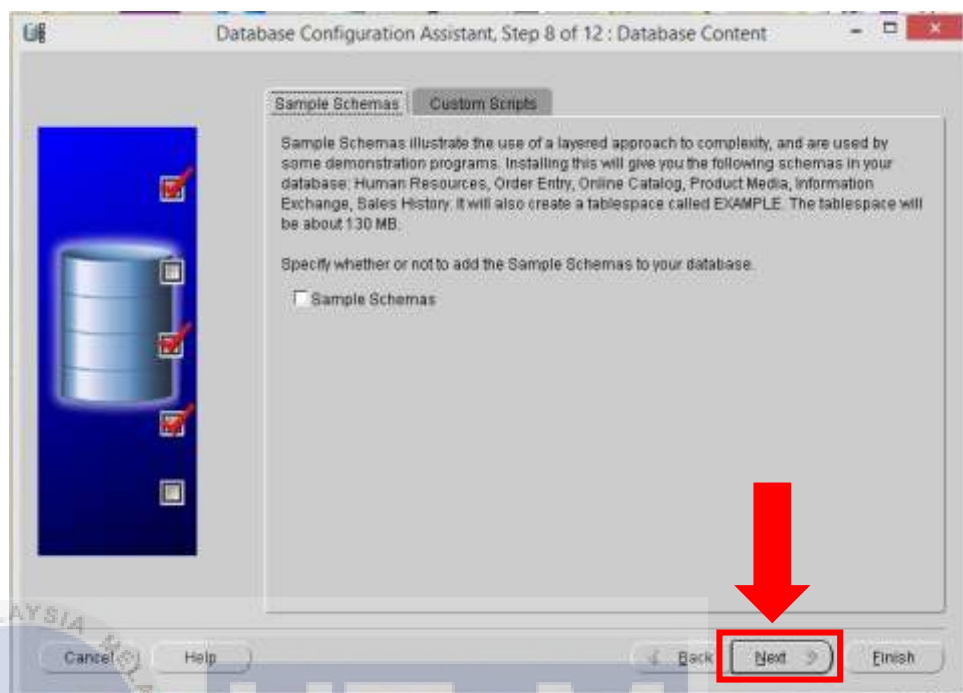
Figure 5.33 Database Content

Step 10: In the page of Initialization Parameters, click on the Characters Sets on the menu bars, and then tick "**Use Unicode (AL32UTF8)**" and then click "**Next**" to continue.
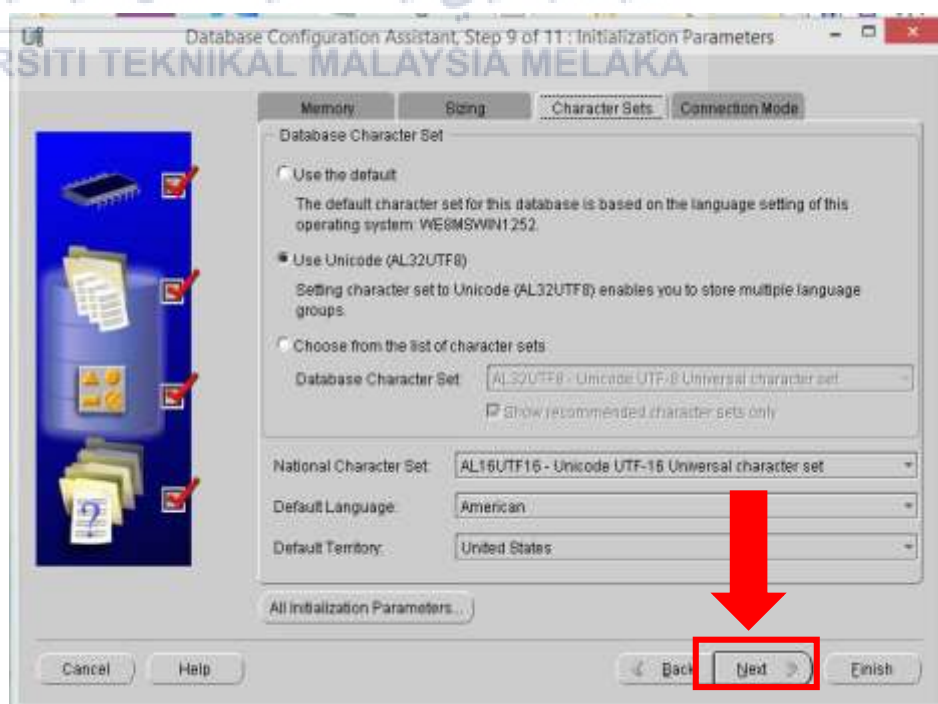
Figure 5.34 Initiation Parameters

Step 11: Click "**Next**" to continue.
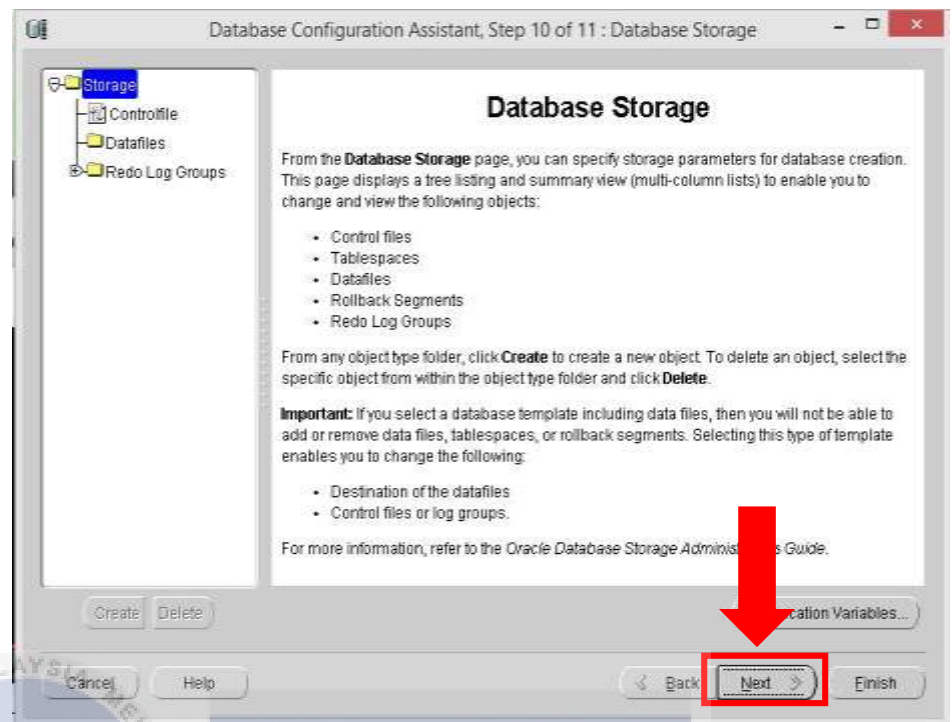
Figure 5.35 Database Storage

Step 12: Next, choose "**Create Database**" and then click "**Finish**".

Figure 5.36 Creation Option

Step 13: Then it will show the summary to view the details of database creation. Click "**OK**".
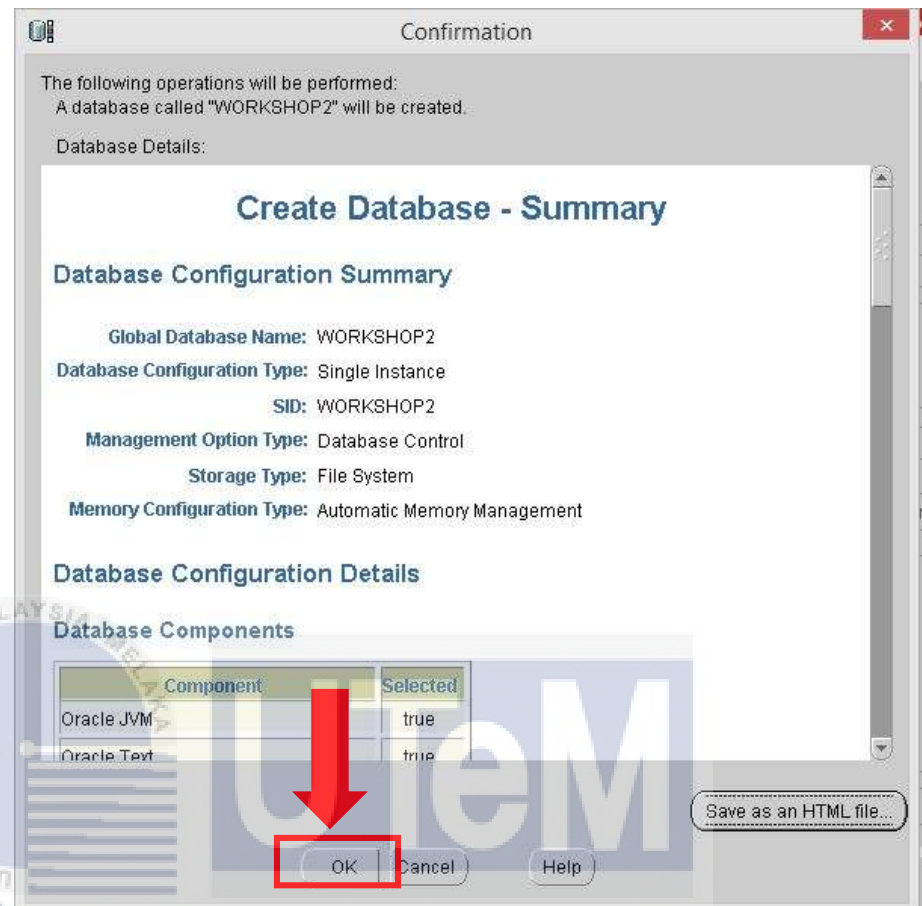
Figure 5.37 Confirmation

5.3 Database Implementation

In the database implementation phase, Database Definition Language (DDL), Database Control Language (DCL) will be explained in detailed manner in order to demonstrate the functionality of the system. Triggers and stored procedures also will be included and explained in this phase as to perform action through the interface Blood Bank Management System.

5.3.1 Data Definition Language (DDL)

Data Definition Language or DDL is used to define the different structures in a database. The standard command of DDL is used to create, modify or remove some database objects such as indexes, tables and users. CREATE, ALTER and DROP are the common DDL statements used in the deployment of Blood Bank Management System.

5.3.1.1 Create Database

Before creating the database objects such as the tables, indexes and users, there must be a database. For the Blood Bank Management System, a database named 'BBMS' have been created in the Oracle Database 11g. 'CREATE DATABASE' command has been used to create the database.
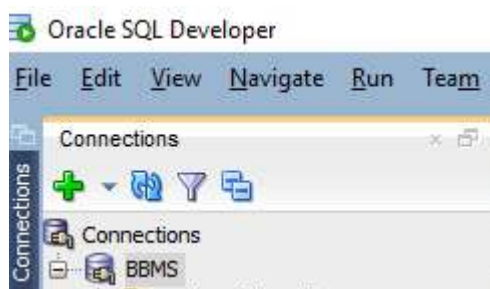
Figure 5.38 Database 'BBMS' in the Oracle SQL Developer has been created.

5.3.1.2 Database Connection

Every database that has been created must have a connection. The connection is used as a link between the Oracle 11g with the Oracle

SQL Developer. 'CREATE CONNECTION' command has been used to create the connection of database BBMS. Figure 5.2 shows the details of the connection.
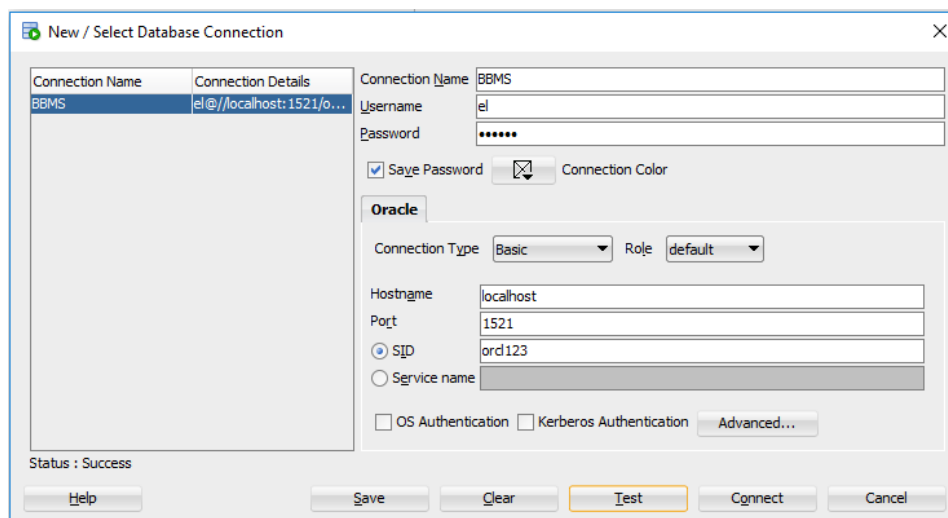
Figure 5.39 The Successful Connection of BBMS Database

5.3.1.3 Tables and Constraints

In BBMS database there are seven tables have been used to store all the data in the database. 'CREATE TABLE' command are used to create all the tables in the database including its attributes, set the primary key and foreign key of the table, defining the data type, the field lengths, default values and constraints.

i.   Table Manager

```
CREATE TABLE "EL"."MANAGER" (

"MANAGER_ID" VARCHAR2 (4 BYTE),
"MANAGER_NAME" VARCHAR2 (50 BYTE) NOT NULL
ENABLE,
"MANAGER_ADDRESS" VARCHAR2 (100 BYTE),
"MANAGER_CONTACT" VARCHAR2 (12 BYTE),
"MANAGER_EMAIL" VARCHAR2 (50 BYTE),
"BLOOD_TYPE" VARCHAR2 (4 BYTE),
"PASSWORD" VARCHAR2 (20 BYTE) NOT NULL ENABLE,

CONSTRAINT "PK_MANAGER" PRIMARY KEY
("MANAGER_ID")
);
```

ii.   Table Staff

```
CREATE TABLE "EL"."STAFF" (

"STAFF_ID" VARCHAR2 (10 BYTE),
"MANAGER_ID" VARCHAR2 (4 BYTE),
"STAFF_NAME" VARCHAR2 (50 BYTE) NOT NULL ENABLE,
"STAFF_ADDRESS" VARCHAR2 (100 BYTE),
"STAFF_CONTACT" VARCHAR2 (12 BYTE),
"STAFF_EMAIL" VARCHAR2 (50 BYTE),
"BLOOD_TYPE" VARCHAR2 (4 BYTE),
"PASSWORD" VARCHAR2 (20 BYTE) DEFAULT 1234 NOT
NULL ENABLE,

CONSTRAINT "PK_STAFF" PRIMARY KEY ("STAFF_ID"),
CONSTRAINT "FK_MANAGER" FOREIGN KEY
("MANAGER_ID") REFERENCES "EL"."MANAGER"
("MANAGER_ID") ENABLE
    );
```

iii.   Table Donor

```
CREATE TABLE "EL"."DONOR" (

"DONOR_ID" VARCHAR2 (12 BYTE),
"STAFF_ID" VARCHAR2 (10 BYTE),
"DONOR_NAME" VARCHAR2 (50 BYTE) NOT NULL
ENABLE,
"DONOR_ADDRESS" VARCHAR2 (100 BYTE),
"DONOR_CONTACT" VARCHAR2 (12 BYTE),
"DONOR_EMAIL" VARCHAR2 (50 BYTE),
"BLOOD_TYPE" VARCHAR2 (4 BYTE),
"DONATED_NUMBER" NUMBER (*, 0) DEFAULT 0,

CONSTRAINT "PK_DONOR" PRIMARY KEY ("DONOR_ID"),
CONSTRAINT "FK_STAFF" FOREIGN KEY ("STAFF_ID")
REFERENCES "EL"."STAFF" ("STAFF_ID") ENABLE
);
```

iv.   Table Receiver

```
CREATE TABLE "EL"."RECEIVER" (

"RECEIVER_ID" VARCHAR2 (10 BYTE),
"RECEIVER_NAME" VARCHAR2 (50 BYTE) NOT NULL
ENABLE,
"RECEIVER_ADDRESS" VARCHAR2 (100 BYTE),
"RECEIVER_CONTACT" VARCHAR2 (12 BYTE),
"RECEIVER_EMAIL" VARCHAR2 (50 BYTE),
"BLOOD_TYPE" VARCHAR2 (4 BYTE),
"RECEIVER_IC" VARCHAR2 (20 BYTE) NOT NULL ENABLE,

PRIMARY KEY ("RECEIVER_ID")
);
```

v.   Table Blood Bank

```
CREATE TABLE "EL"."BLOOD_BANK" (

"BANK_ID" VARCHAR2 (6 BYTE),
"MANAGER_ID" VARCHAR2 (4 BYTE),
"BLOOD_TYPE" VARCHAR2 (4 BYTE),
"BLOOD_QUANTITY" NUMBER (*, 0),

CONSTRAINT "PK_BANK" PRIMARY KEY ("BANK_ID"),
CONSTRAINT "FK_BANKMANAGER" FOREIGN KEY
("MANAGER_ID") REFERENCES "EL"."MANAGER"
("MANAGER_ID") ENABLE
);
```

vi.   Table Blood

```
CREATE TABLE "EL"."BLOOD" (

"BLOOD_ID" NUMBER (*, 0) NOT NULL ENABLE,
"DONOR_ID" VARCHAR2 (12 BYTE) NOT NULL ENABLE,
"PINT" VARCHAR2 (5 BYTE),
"DONATION_DATE" DATE,
"EXPIRED_DATE" DATE,
"STATUS" VARCHAR2 (15 BYTE) DEFAULT 'IN PROCESS',
"BANK_ID" VARCHAR2 (6 BYTE),

CONSTRAINT "PK_BLOOD" PRIMARY KEY ("BLOOD_ID")
CONSTRAINT "FK_DONOR" FOREIGN KEY ("DONOR_ID")
REFERENCES "EL"."DONOR" ("DONOR_ID") ENABLE,
CONSTRAINT "FK_BANK" FOREIGN KEY ("BANK_ID")
REFERENCES "EL"."BLOOD_BANK" ("BANK_ID") ENABLE
);
```

vii.   Table Receive Blood

```
CREATE TABLE "EL"."RECEIVE_BLOOD" (

"RECEIVER_ID" VARCHAR2 (10 BYTE) NOT NULL
ENABLE,
"BLOOD_ID" NUMBER (38, 0) NOT NULL ENABLE,
"RECEIVE_DATE" DATE,
CONSTRAINT "FK_BLOOD" FOREIGN KEY ("BLOOD_ID")
REFERENCES "EL"."BLOOD" ("BLOOD_ID") ENABLE,
CONSTRAINT "FK_RECEIVER" FOREIGN KEY
("RECEIVER_ID")
REFERENCES "EL"."RECEIVER" ("RECEIVER_ID") ENABLE
);
```

5.3.1.4 Sequence Table

Sequence is used in the tables of the BBMS database. Sequence create unique values automatically to act as primary key, used to increments a value and returns it. 'CREATE SEQUENCE' command are used to create a sequence in a table. In BBMS database, three tables use sequences to generate the primary key for ID attributes.

i.   Sequence ID for table Staff

```
CREATE SEQUENCE "EL"."SEQ_STAFFID"
MINVALUE 1
MAXVALUE 999
INCREMENT BY 1
START WITH 21
CACHE 20
NOORDER
NOCYCLE;
```

ii.   Sequence ID for table Receiver

```
CREATE SEQUENCE "EL"."SEQ_RECEIVERID"
MINVALUE 0
MAXVALUE 99999
INCREMENT BY 1
START WITH 41
CACHE 20
NOORDER
NOCYCLE;
```

iii.   Sequence ID for table Blood

```
CREATE SEQUENCE "EL"."SEQ_BLOODID"
MINVALUE 0
MAXVALUE 99999
INCREMENT BY 1
START WITH 41
CACHE 20
NOORDER
NOCYCLE;
```

5.3.1.5 Trigger

Trigger is a kind of stored procedure that will automatically executes or enable the function when an event occurs in the database server. Usually, trigger will be execute when there is a modification through a Data Manipulated Language (DML). DML events that occur are when INSERT, UPDATE or DELETE statements on a table. 'CREATE TRIGGER' command is used to create trigger function in the database. The followings are the triggers that have been used in the BBMS database. Other triggers refer to Attachment 2.0.

Trigger to generate ID for new registered staff

```
create or replace trigger TRIG_STAFFID
before insert on staff
for each row

begin
if :NEW.STAFF_ID is NULL then
select 'S00'||seq_staffID.nextval into :NEW.STAFF_ID from dual;
end if;
END;
```

5.3.1.6 Stored Procedure

Stored procedure is a set of SQL statements which can be used to access or modify data in the database. Stored procedures used in the BBMS database will help to reduce network traffic and increase database performance. 'CREATE PROCEDURE' command will be used to create stored procedure in the database. The followings are the stored procedures that have been used in the BBMS database. Other stored procedures refer to Attachment 1.0.

Procedure to insert new registered staff into the table staff.

```
create or replace PROCEDURE procINSERTSTAFF (

  vManagerID IN STAFF.MANAGER_ID%TYPE,
  vStaffName IN STAFF.STAFF_NAME%TYPE,
  vStaffAdress IN STAFF.STAFF_ADDRESS%TYPE,
  vStaffContact IN STAFF.STAFF_CONTACT%TYPE,
  vStaffEmail IN STAFF.STAFF_EMAIL%TYPE,
  vBloodType IN STAFF.BLOOD_TYPE%TYPE
  )
IS
BEGIN
  INSERT INTO STAFF ("MANAGER_ID", "STAFF_NAME",
"STAFF_ADDRESS", "STAFF_CONTACT", "STAFF_EMAIL",
"BLOOD_TYPE")
  VALUES (vManagerID, vStaffName, vStaffAdress, vStaffContact,
vStaffEmail, vBloodType);
  COMMIT;
END;
```

5.3.2 Data Control Language (DCL)

Data Control Language (DCL) is one of the SQL statements that is used to control access of the data stored in a database. DCL is an operation to grant or to revoke privileges from a user or role. This will allow the user to do DDL and DML in the database. In order to make the BBMS database is privileged on the table, the grant command is used and show as the following SQL.

SQL > CREATE user 'el' identified by 'abc123';

SQL > conn /as sysdba;

SQL > GRANT all privileges to el;

SQL > conn el/abc123;

SQL > GRANT resource TO el;

SQL > GRANT CREATE SESSION GRANT ANY PRIVILEGE TO el;

5.4 Conclusion
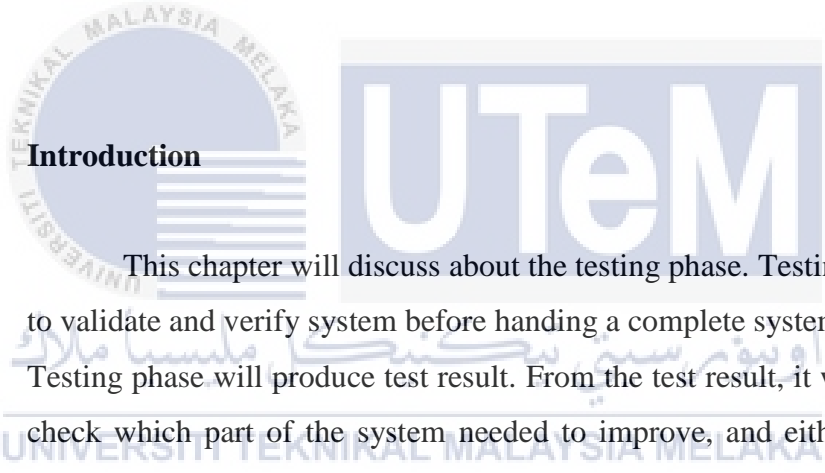
BBMS database has been created for the Blood Bank Management System. Oracle database has been used and XAMPP server act as the server-side script language to keep and store all the data. In the other hand, sequences, triggers and stored procedures have been used and explained in the implementation phase for the deployment of Blood Bank Management System.

Next chapter is the testing phase which is the Blood Bank management System will be test and verify by the user to validate all the requirements. Testing phase will include the test plan, test strategy, test design, test results and analysis.

CHAPTER VI

6.0 TESTING

6.1   Introduction

This chapter will discuss about the testing phase. Testing is a method to validate and verify system before handing a complete system to the client. Testing phase will produce test result. From the test result, it will be used to check which part of the system needed to improve, and either the system meets specified requirements.

System testing is a very important phase. From the testing phase, it will ensure that the system should not result into any failures so that it can reduce the costing of system deployment in future or in the later stages of development. For Blood Bank Management System, the activities that involved in the testing phase are test plan, test strategy, test design, test results and analysis of the testing.

6.2   Test Plan

Test plan is a document that serve as a communication tool to describe scope, approach, resources and schedule of testing activities. It can also help to track changes to the testing project. As changes to the test plan are made on the items to be tested, the test plan document should be updated to reflect those decisions.

There are some submodules of test plan involved on the testing phase of Blood Bank Management System which are test organization, test environment and test schedule.

6.2.1   Test Organization

Test organization defines as the person who is responsible for what activity in the test phase. The organization defines the test functions, test facilities and activities. For the Blood Bank Management System, test organization will include two users who are reacted to the system which are manager who are the administrator and staff who worked in blood department. All the user will be tested based on their roles. Table 6.1 describe the roles of the users in Blood Bank Management System.

Table 6.1 User's Role

Users	Role
Manager (Administrator)	• Testing the system based on the given script • Testing of admin's module • Check the fault and defect detection
Staff	• Testing the system based on the given script • Testing of staff's module • Check the fault and defect detection

6.2.2 Test Environment

Test environment consist of the testing of hardware and software of the newly built software or system. It also supports test execution of network configuration.

6.2.2.1 Hardware Requirement

Table 6.2 shows the hardware that have been used in the development of Blood Bank Management System.

Table 6.2 Hardware Requirements

Item	Description
Processor	Processor is the logic circuit that responds and processes the basic instructions that drive a computer. The four primary functions of processor are fetch, decode, execute and write back. For this project, we need a very good processor to avoid any corruption during development phase.
Memory	Computer memory is any physical device capable of storing information temporarily or permanently. The memory need to be operated well to avoid slower performance of laptop and any corruption.
Hard disc	Hard disk drive (HDD) is the mechanism that controls the positioning, reading and writing of the hard disk, which furnishes data storage. To develop this system, we need sufficient storage to store all the data, the GUI and coding for the system. We also need to back up the whole system to the hard disc if anything happen to the laptop.
Monitor	Monitor is computer display and related parts packaged in physical unit that is separate from other parts of the computer but for laptop, the monitor and keyboard are attach together. To

	display the output for the system, we need the monitor to show how Integrated Accommodation Agent System works.
Graphic Card	The graphics components are the part of your computer that control and enhance how graphics are displayed on your computer screen. Without graphic card, we are not able to display the system to the monitor.

6.2.2.2 Software Requirement

Table 6.3 shows software that have been used in the development of Blood Bank Management System.

Table 6.3 Software Requirements

Item	Description
Software Design	• Using Microsoft Visio 2013
Server Provider	• Using XAMPP 3.2.2
Operating System	• Using Windows 10
Database Installer	• Using Oracle 11g

6.2.3 Test Schedule

Test schedule will include the time taken to do the testing tasks on the system. Test schedule will have the target start date end date of some testing activities or tasks. For Blood Bank Management System, there will have three (3) tasks of task schedule which are unit testing, integration testing and acceptance.

Table 6.4 Task Schedule

Task	Start Date	End Date
1. Unit Testing	05 June 2017	18 June 2017
2. Integration Testing	19 June 2017	02 July 2017
3. Acceptance	14 August 2017	18 August 2017

6.3   Test Strategy

To do the testing on the Blood Bank Management System, it must have the test strategy. Test strategy is an approach to start deal with the testing tasks in a certain way. It will defines on how testing would be carried out and to make sure if the system will run smoothly without any detection of error. There have some classes of test included that have mentioned in Section 6.3.1.

6.3.1   Classes of Test

Table 6.5 Classes of Test and Detailed Description

Classes of Test	Description
i. Accessibility Test	• This test is to the system can be accessed or viewed by users with related data
ii. Graphical User Interface Test	• This test is to the GUI is suitable use to the user
iii. Security Test	• This test is to check the data are shown to authenticated and authorized users correctly
iv. Usability Test	• Application flow, user experience and usability of using the system are tested in this test to ensure users are able get hold and understand the system well and easily.

6.4 Test Design

Test design is a set of test cases that will be used as a documentation to do the testing for a system or software. It will do the analysis and identifying test conditions and give a generic idea for testing which covers a large range of possibilities. There will be two (2) designs involved in the Blood Bank Management System which are the test description and test data. Each of the test are based on the two (2) users: manager (administrator) and staff.

6.4.1 Test Description

The followings are test description for the Blood Bank Management System.

i. **Login**

It will describe the test description for the login of the manager (administrator) and staff. The purpose of the test is to ensure that the authorized user of the system which are the manager and staff can login into the system successfully. Test ID for Login is A001.

Table 6.6 Login Test Description

Test Case ID	Test Cases	Test Data	Expected Result
A001_01	Validate that user not enter username or password.	Username: - Password: -	There will be a pop-up message display on the screen showing that specify field cannot be empty.
A001_02	Validate that user insert incorrect username or password.	Username: M001 Password: M001 or Username: A001 Password: manager1	Error message display on the screen "Wrong Username or

			Password! Please insert again."
A001_03	Validate that user insert correct username and password.	Username: M001 Password: manager1	Successfully login into the system.
A001_04	Validate that log in button is enable when clicked.		Successfully login into the system.
A001_05	Validate that log in button is disable when clicked.		Unsuccessfully login into the system.
A001_05	Validate that character inserted in the username field is same as the data type value.	Username: M001	Successfully insert username.
A001_06	Validate that character inserted in the username field is not same as the data type value.	Username: M00001	Unsuccessfully insert username.

ii.   **Registration**

It will describe the testing of register new staff, donor and receiver test description. The purpose of this test is to ensure that an authorized user will be able to register new staff, donor and receiver into the database system successfully. Test ID for the registration is A002.

Table 6.7 Registration Test Description (New Staff)

Test Case ID	Test Cases	Test Data	Expected Result
A002_1	Validate that register new staff option is available.		Registration option is display.
A002_2	Validate that register new staff option is unavailable.		Registration option not display.
A002_3	Validate that user not insert all staff information in the registration field.	Name: Siti Address: Melaka Contact: 012-6440019 Email:	There will be a pop-up message display on the screen showing that specify field cannot be empty.
A002_4	Validate that user insert all staff information registration field.	Name: Siti Address: Melaka Contact: 012-6440019 Email: siti@gmail.com	Successfully inserting the data information.
A002_5	Validate that register button is enable.		New staff is successfully registered.

A002_6	Validate that register button is disable.		New staff is unsuccessfully registered.
A002_7	Validate that character inserted in the contact phone field is not same as the format value.	Contact:01264400125	Unsuccessfully insert contact phone.
A002_8	Validate that character inserted in the contact phone field is same as the format value.	Contact: 012-6440019	Successfully insert contact phone.
A002_9	Validate that character inserted in the email field is not same as the format value.	Email: siti.salwa.com	Unsuccessfully insert email.
A022_10	Validate that character inserted in the email field is same as the format value.	Email: siti@gmail.com	Successfully insert email.

Table 6.8 Registration Test Description (New Donor)

Test Case ID	Test Cases	Test Data	Expected Result
A002_11	Validate that register new donor option is available.		Registration option is display.
A002_12	Validate that register new donor option is unavailable.		Registration option not display.
A002_13	Validate that user not insert all donor information in the registration field.	Name: Mohd Address: Selangor Contact: 011-1234567 Email:	There will be a pop-up message display on the screen showing that specify field cannot be empty.
A002_14	Validate that user insert all donor information registration field.	Name: Mohd Address: Selangor Contact: 011-1234567 Email: mohd@yahoo.com	Successfully inserting the data information.
A002_15	Validate that register button is enable.		New donor is successfully registered.
A002_16	Validate that register button is disable.		New donor is unsuccessfully registered.
A002_17	Validate that character inserted in the contact phone field is not	Contact:011123456789	Unsuccessfully insert contact phone.

A002_18	Validate that character inserted in the contact phone field is same as the format value.	Contact:011-1234567	Successfully insert contact phone.
	same as the format value.		
A002_19	Validate that character inserted in the email field is not same as the format value.	Email: mohd.yahoo.com	Unsuccessfully insert email.
A002_20	Validate that character inserted in the email field is same as the format value.	Email: mohd@yahoo.com	Successfully insert email.

Table 6.9 Registration Test Description (New Receiver)

Test Case ID	Test Cases		Expected Result
A002_21	Validate that register new receiver option is available.		Registration option is display.
A002_22	Validate that register new receiver option is unavailable.		Registration option not display.
A002_23	Validate that user not insert all receiver information in the registration field.	Name: Ali Address: Selangor Contact: 011-1234567 Email:	There will be a pop-up message display on the screen showing that specify field cannot be empty.
A002_24	Validate that user insert all receiver information registration field.	Name: Ali Address: Selangor Contact: 011-1234567 Email: ali@yahoo.com	Successfully inserting the data information.
A002_25	Validate that register button is enable.		New receiver is successfully registered.
A002_26	Validate that register button is disable.		New receiver is unsuccessfully registered.
A002_27	Validate that character inserted in the contact phone field is not same as the format value.	Contact:0123456789	Unsuccessfully insert contact phone.

A002_28	Validate that character inserted in the contact phone field is same as the format value.	Contact:012-3456789	Successfully insert contact phone.
A002_29	Validate that character inserted in the email field is not same as the format value.	Email: ali.yahoo.com	Unsuccessfully insert email.
A002_30	Validate that character inserted in the email field is same as the format value.	Email: ali@yahoo.com	Successfully insert email.

iii. Updating Information

It will define the test cases of testing to update information. The purpose of this test is to ensure that user will be able to update the information successfully. Test ID for the registration is A003.

Table 6.10 Updating Information Test Description

Test Case ID	Test Cases	Test Data	Expected Result
A003_01	Validate that update option is available.		Update option is display.
A003_02	Validate that update option is unavailable.		Update option is not display.
A003_03	Validate that user not filling any data in the required field.		There will be a pop-up message display on the

A003_04	Validate that user change new information in the required field.		Successfully updating the data information.
A003_05	Validate that update button is enable.		Pop up message shows that data successfully updated is display on the screen. New information is successfully updated.
A003_06	Validate that update button is disable.		New information is unsuccessfully updated.

| | | | screen showing that specify field cannot be empty. |

iv. **Deleting Information**

This section will define the test cases of testing to delete information. The purpose of this test is to ensure that user will be able to delete the information successfully. Test ID for the registration is A004.

Table 6.11 Deleting Information Test Description

Test Case ID	Test Cases	Expected Result
A004_01	Validate that delete option is available.	Delete option is display.
A004_02	Validate that delete option is unavailable.	Delete option is not display.
A004_03	Validate that delete button is enable.	Pop up message shows that data successfully deleted is display on the screen. Information is successfully deleted.
A004_04	Validate that delete button is disable.	Information is unsuccessfully deleted.

v.   **Check Donor**

This section will define the test cases of testing to check the information. The purpose of this test is to ensure either the donor has been registered as a donor or not. Test ID for the registration is A005.

Table 6.12 Check Donor Test Description

Test Case ID	Test Cases	Test Data	Expected Result
A005_01	Validate that check option is available.		Check option is display.
A005_02	Validate that check option is unavailable.		Check option is not display.
A005_03	Validate that user do not insert IC number of donor in the IC number field.	IC Number: -	There will be a pop-up message display on the screen showing that specify field cannot be empty.
A005_04	Validate that user do insert IC number of registered donor in the IC number field.	IC Number: 950423105728	Donor information display on the screen.
A005_05	Validate that user insert IC number of unregistered donor in the IC number field.	IC Number: 950206106347	There will be a pop-up message display that no data for the IC Number inserted.
A005_06	Validate that character inserted in the IC Number field is same as the format value.	IC Number: 950423105728	Successfully insert IC Number.

A005_07	Validate that character inserted in the IC Number field is do not same as the format value.	IC Number: Alia950423	Unsuccessfully insert IC Number.

6.5 Test Results and Analysis

This section will describe about the validation of the data over Blood Bank Management System. This test document is to see whether the result in test design is the same with the test results and analysis document. Table below shows the details of the test document.

Table 6.13 Test Results

Test Case Identification	Test Identification	Result (Passed / Failed)
A001: Login Test Description		
A001_01	OK	Passed
A001_02	OK	Passed
A001_03	OK	Passed
A001_04	OK	Passed
A001_05	OK	Passed
A001_06	OK	Failed
A001_07	OK	Failed
A002: Registration Test Description		
A002_01	OK	Passed
A002_02	OK	Passed
A002_03	OK	Passed
A002_04	OK	Passed
A002_05	OK	Passed
A002_06	OK	Passed
A002_07	OK	Failed
A002_08	OK	Failed

A002_09	OK	Failed
A002_10	OK	Failed
A002_11	OK	Passed
A002_12	OK	Passed
A002_13	OK	Passed
A002_14	OK	Passed
A002_15	OK	Passed
A002_16	OK	Passed
A002_17	OK	Failed
A002_18	OK	Failed
A002_19	OK	Failed
A002_20	OK	Failed
A002_21	OK	Passed
A002_22	OK	Passed
A002_23	OK	Passed
A002_24	OK	Passed
A002_25	OK	Passed
A002_26	OK	Passed
A002_27	OK	Failed
A002_28	OK	Failed
A002_29	OK	Failed
A002_30	OK	Failed
A003: Updating Information Test Description		
A003_01	OK	Passed
A003_02	OK	Passed
A003_03	OK	Passed
A003_04	OK	Passed
A003_05	OK	Passed
A003_06	OK	Passed
A004: Deleting Test Description		
A004_01	OK	Passed
A004_02	OK	Passed
A004_03	OK	Passed

A004_04	OK	Passed
A005: Check Donor Test Description		
A005_01	OK	Passed
A005_02	OK	Passed
A005_03	OK	Passed
A005_04	OK	Passed
A005_05	OK	Passed
A005_06	OK	Passed
A005_07	OK	Passed

6.6   Conclusion

System testing is one of the most important phase in the system development. Testing is done to ensure that system will function correctly and properly. All the modules of testing phase had been explained in this chapter. From this phase, Blood Bank Management System has been improved and fixed the bug and errors.

CHAPTER VII

7.0 CONCLUSION

7.1   Introduction

This chapter will describe about overall conclusion for the Blood Bank Management System. It will include the strengths and weaknesses of the system. For the overall conclusion, Chapter I has been describe about background project of Blood Bank Management System which consists of the problem statements,   objectives, project scope and project significance. Next, Chapter II has been discussed about the project methodology and planning. This chapter is also include the project schedule and milestone for the deployment of Blood Bank Management System. Chapter III is about the analysis phase of system. In this chapter, all the functional and non-functional requirements have been identified. Next, chapter IV is about design which discuss about the conceptual, logical and physical design of the Blood Bank Management System. This chapter also include the Graphical User Interface (GUI) design. The last chapter for development of Blood Bank Management System is testing phase which is one of the most important phase. All the chapter have been discussed.

7.2   Observation on Weakness and Strengths

Every system has its own strengths and weaknesses including Blood Bank Management System. Blood Bank Management System is developed based on the needs and requirements of manager and staff. This is a computer-based system that is designed for these two users. As mentioned earlier, some of the strengths and weaknesses of the system have been summarized as following.

a) Strengths

i.   **Efficiency Improvement**

Compare to the old system, this system can improve in term of the efficiency of managing data information of donor, blood and receiver in the blood department. With this system, all the data are stored in the centralized database. It will be easier for the user to stored and retrieve data when needed.

ii.   **Reduce Human Error**

This system is totally change from the older system which is manually calculating total number of blood for each blood in the blood bank. With this system, staff can know the total number of exact blood in the blood bank for each type of the blood. Beside, this system also provide alert function which can give notification to the user if there are any blood that will be expired within a week. So, user do not need to check by manually the date of blood expired and in the same time user also know how many days left for the blood before its expired for further action.

iii.   **Usability**

There have not much to learn how to use the system as the system can be easily be used by the administrator, staffs and customers. The system is developed based on the requirements and the interfaces of the system are user friendly. Hence, the users are able to use the system easily.

b) Weaknesses

i. Lack of Graphical Features

This system has the report feature for each part but does not provide the graphical features to overview the results.

ii. Testing with dummy data

This system is tested with dummy data and values. This is one of the most important step to really test and detect any defects or fault existed in the system.

7.3 Propositions of Improvement

As mentioned in weaknesses section of this chapter, the system requires some improvement to ensure the system can be truly implemented in order for administrator, staff and customers to use. Following are the propositions for improving the system to a better state and use.

i. Enhancing Graphical Features

The system should consist of better graphical features allowing administrator to have a better overview of the results for each part such as receiving the payment for daily, population of customers are making the appointment in a week and others. It is more obvious and easily to be viewed.

ii. Testing with Real Data

Before implementing this system, it is suggested that Edward Hair Salon Management System is tested on the server with real data to test the true performances of the system before letting to the users to develop and use it.

7.4 Project Contribution

Blood Bank Management System is designed and developed based on the requirements for two users which are the manager as the administrator and staff.

a) Manager (administrator)

Manager is the administrator to this system. This system will allow the manager to do the registration for the new staff so that the staff can use this system. For all the function of this system, manager is able to do it.

b) Staff

Staff is one of the user to use this system. Staff can update their profile, do the registration for the new donor and receiver and also can check the expired date of the blood in the blood bank.

7.5 Conclusion

Finally, for the conclusion of this chapter and the rest of the chapters, Blood Bank Management System has successfully meet the objectives of the project. Both functional and non-functional requirements of project have been fulfilled.

In a nutshell, this project has been successfully completed in partial fulfilment of the requirements for the Bachelor of Computer Science (Database Management).

ATTACHMENT 1.0

APPENDIX A

i.   Trigger to generate ID for new registered of blood receiver

```
create or replace trigger TRIG_RECEIVERID
before insert on receiver
for each row

begin
if :NEW.RECEIVER_ID is NULL then
select 'R00'||seq_receiverID.nextval into :NEW.RECEIVER_ID
from dual;
end if;
END;
```

ii.   Trigger to generate ID for blood

```
create or replace trigger TRIG_BLOODID
before insert on BLOOD
for each row

begin
if :NEW.BLOOD_ID is NULL then
select seq_bloodID.nextval into :NEW.BLOOD_ID from dual;
end if;
END;
```

iii.   Trigger to update the number of blood donated by the donor every time when they make a new donation. This trigger will be execute in the donor's table.

```
create or replace trigger UPDATE_DONATED_NUMBER
after insert on BLOOD
for each row

BEGIN
  UPDATE DONOR
  SET DONATED_NUMBER=DONATED_NUMBER + 1
  WHERE DONOR_ID=:NEW.DONOR_ID;
END;
```

iv.   Trigger to update the number of blood quantity for each type of blood in the blood bank. This trigger will be execute in the blood bank's table when there is new blood registered in the table blood.

```
create or replace trigger UPDATE_QUANTITY
after insert on BLOOD
for each row
declare
v_type DONOR.BLOOD_TYPE%TYPE;

BEGIN
  SELECT BLOOD_TYPE
  INTO v_type
  FROM DONOR
  WHERE DONOR_ID=:NEW.DONOR_ID;

  UPDATE BLOOD_BANK
  SET BLOOD_QUANTITY=BLOOD_QUANTITY + 1
  WHERE BLOOD_TYPE=v_type;
END;
```

iv.   Trigger to update the status of blood when there is a receiver received the blood.

```
create or replace trigger UPDATE_STATUS
after insert on RECEIVE_BLOOD
for each row

BEGIN
  UPDATE BLOOD
  SET STATUS='NOT AVAILABLE'
  WHERE BLOOD_ID=:NEW.BLOOD_ID;
END;
```

ATTACHMENT 2.0

APPENDIX B

i.   Procedure to insert new registered donor into the table donor.

```
create or replace PROCEDURE procINSERTDONOR (

 vDonorIC IN DONOR.DONOR_ID%TYPE,
 vDonorName IN DONOR.DONOR_NAME%TYPE,
 vDonorAddress IN DONOR.DONOR_ADDRESS%TYPE,
 vDonorContact IN DONOR.DONOR_CONTACT%TYPE,
 vDonorEmail IN DONOR.DONOR_EMAIL%TYPE,
 vBloodType IN DONOR.BLOOD_TYPE%TYPE
 )
IS
BEGIN
 INSERT INTO DONOR ("DONOR_ID", "DONOR_NAME",
"DONOR_ADDRESS", "DONOR_CONTACT",
"DONOR_EMAIL", "BLOOD_TYPE")
 VALUES (vDonorIC, vDonorName, vDonorAddress,
vDonorContact, vDonorEmail, vBloodType);

 COMMIT;
END;
```

ii.   Procedure to insert new registered receiver into the table receiver.

```
create or replace PROCEDURE procINSERTRECEIVER (
 vreceiverName IN RECEIVER.RECEIVER_NAME%TYPE,
 vreceivefAdress IN RECEIVER.RECEIVER_ADDRESS%TYPE,
 vreceiveContact IN RECEIVER.RECEIVER_CONTACT%TYPE,
 vreceiveEmail IN RECEIVER.RECEIVER_EMAIL%TYPE,
 vbloodType IN RECEIVER.BLOOD_TYPE%TYPE
 )
IS
BEGIN
 INSERT INTO RECEIVER ("RECEIVER_NAME",
"RECEIVER_ADDRESS", "RECEIVER_CONTACT",
"RECEIVER_EMAIL", "BLOOD_TYPE")
 VALUES (vreceiverName, vreceivefAdress, vreceiveContact,
vreceiveEmail, vbloodType);
 COMMIT;
END;
```

iii.   Procedure to insert new blood donated by the donor into the table blood.

```
create or replace PROCEDURE procINSERTBLOOD (
  vdonorID IN BLOOD.DONOR_ID%TYPE,
  vpint IN BLOOD.PINT%TYPE,
  vdonationDate IN BLOOD.DONATION_DATE%TYPE,
  vbankID IN BLOOD.BANK_ID%TYPE
  )
IS
BEGIN
  INSERT INTO BLOOD ("DONOR_ID", "PINT",
"DONATION_DATE", "BANK_ID")
  VALUES (vdonorID, vpint, vdonationDate, vbankID);

  COMMIT;
END;
```

iv.   Procedure to update staff information.

```
create or replace PROCEDURE procUPDATESTAFF
  (
  vstaffID IN STAFF.STAFF_ID%TYPE,
  vName IN STAFF.STAFF_NAME%TYPE,
  vAddress IN STAFF.STAFF_ADDRESS%TYPE,
  vContact IN STAFF.STAFF_CONTACT%TYPE,
  vEmail IN STAFF.STAFF_EMAIL%TYPE
  )
IS
BEGIN
UPDATE STAFF
SET STAFF_NAME = vName, STAFF_ADDRESS = vAddress,
STAFF_CONTACT = vContact, STAFF_EMAIL = vEmail
WHERE STAFF_ID = vstaffID;
END;
```

v.   Procedure to update receiver information.

```
create or replace PROCEDURE procUPDATERECEIVER
  (
  vReceicerID IN RECEIVER.RECEIVER_ID%TYPE,
  vName IN RECEIVER.RECEIVER_NAME%TYPE,
  vAddress IN RECEIVER.RECEIVER_ADDRESS%TYPE,
  vContact IN RECEIVER.RECEIVER_CONTACT%TYPE,
  vEmail IN RECEIVER.RECEIVER_EMAIL%TYPE,
  vBloodType IN RECEIVER.BLOOD_TYPE%TYPE)
IS
BEGIN
UPDATE RECEIVER
SET RECEIVER_NAME = vName, RECEIVER_ADDRESS =
vAddress, RECEIVER_CONTACT=vContact,
RECEIVER_EMAIL = vEmail, BLOOD_TYPE = vBloodType
WHERE RECEIVER_ID = vReceicerID;
END;
```

vi.   Procedure to update donor information.

```
create or replace PROCEDURE procUPDATEDONOR
  (
  vDonorID IN DONOR.DONOR_ID%TYPE,
  vName IN DONOR.DONOR_NAME%TYPE,
  vAddress IN DONOR.DONOR_ADDRESS%TYPE,
  vContact IN DONOR.DONOR_CONTACT%TYPE,
  vEmail IN DONOR.DONOR_EMAIL%TYPE,
  vBloodType IN DONOR.BLOOD_TYPE%TYPE)
IS
BEGIN
UPDATE DONOR
SET DONOR_NAME = vName, DONOR_ADDRESS = vAddress,
DONOR_CONTACT=vContact,
DONOR_EMAIL = vEmail, BLOOD_TYPE = vBloodType
WHERE DONOR_ID = vDonorID;
END;
```

vii. Procedure to update the blood if there is a receiver received the blood. Data that will be updated is the date of receiving blood.

```
create or replace PROCEDURE procRECEIVEBLOOD (
  vbloodID IN RECEIVE_BLOOD.BLOOD_ID%TYPE,
  vreceiverID IN RECEIVE_BLOOD.RECEIVER_ID%TYPE,
  vreceiveDate IN RECEIVE_BLOOD.RECEIVE_DATE%TYPE
  )
IS
BEGIN
  INSERT INTO RECEIVE_BLOOD ("BLOOD_ID",
"RECEIVER_ID", "RECEIVE_DATE")
  VALUES (vbloodID, vreceiverID, vreceiveDate);

  COMMIT;
END;
```

viii. Procedure to update the blood information.

```
create or replace PROCEDURE procUPDATEBLOOD
  (
  vbloodID IN BLOOD.BLOOD_ID%TYPE,
  vexpiredDate IN BLOOD.EXPIRED_DATE%TYPE,
  vstatus IN BLOOD.STATUS%TYPE)
IS
BEGIN
UPDATE BLOOD
SET EXPIRED_DATE = TO_DATE(vexpiredDate,
'DD/MM/YYYY'), STATUS = vstatus WHERE BLOOD_ID =
vbloodID;
END;
```

ix. Procedure to delete staff that has not working in the blood department anymore.

```
create or replace PROCEDURE DELETE_STAFF(H_ID IN
VARCHAR2)
IS BEGIN
DELETE FROM STAFF
WHERE STAFF_ID=H_ID;
COMMIT;
END;
```

REFERENCES

Abdul Aziz K., Nurul Ain Najihah Yusri, Sumazly Sulaiman. (July 2015). Procedia-Social and Behavioral Sciences. *Development of a Blood Bank Management System*. 195 (2015), 2008-2013.

Frankfurter, George & E. Jelmert, A & Pegels, C & P. Seagle, J & L. Wallace, E. (1970). A Regional Computer-based Blood Inventory Information System. *Transfusion*. 10. 203 - 214.

P. Connelly, D & A. Boraas, B & C. Leavitt, W & McCullough, J. (1976). A Regional Blood Bank Automated Information System. *Laboratory Medicine*. 7. 15-20.

"Online Blood Bank Management System | TANSACS." Online Blood Bank Management System | TANSACS. Retrieved from www.tngovbloodbank.in/.