MALAYSIA AUXILIARY POLICE ASSOCIATION (APA) MANAGEMENT SYSTEM (TRANSPORT MODULE)

UMMI IZZATI BINTI MUSTAPHA

UNIVERSITI TEKNIKAL MALASIA MELAKA

BORANG PENGESAHAN STATUS TESIS*

JUDUL: **MALAYSIA AUXILIARY POLICE ASSOCIATION (APA) MANAGEMENT SYSTEM (TRANSPORTATION MODULE)**

SESI PENGAJIAN: **2017**

Saya **UMMI IZZATI BINTI MUSTAPHA**

(HURUF BESAR)

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 RASMI 1972)
____	TERHAD	(Mengandungi maklumat TERHAD yang telah ditentukan oleh organisasi/badan di mana penyelidikan dijalankan)
/	TIDAK TERHAD	

(TANDATANGAN PENULIS)

Alamat tetap: 4149-10
Kg Bukit Mentok,
24000,Kemaman,
Terengganu

Tarikh: 24/8/2017

(TANDATANGAN PENYELIA)

PUAN NOR MAS AINA MD BOHARI
Nama Penyelia

Tarikh: 24/8/2017

MALAYSIA AUXILIARY POLICE ASSOCIATION (APA) MANAGEMENT SYSTEM (TRANSPORT MODULE)

UMMI IZZATI BINTI MUSTAPHA

This report is submitted in partial fulfillment of the requirement for the

Bachelor Degree of Computer Science (Database Management)

FACULTY OF INFORMATION AND COMMUNICATIO TECHNOLOGY

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2017

DECLARATION

DECLARATION

I hereby declare this project report entitled

MALAYSIA AUXILIARY POLICE ASSOCIATION (APA) MANAGEMENT SYSTEM (TRANSPORT MODULE)

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

STUDENT : _____ Date: 24/8/2017

(UMMI IZZATI BINTI MUSTAPHA)

SUPERVISOR : _____ Date: 24/8/2017

(PUAN NOR MAS AINA BINTI MD BOHARI)

DEDICATION

I dedicated this thesis for who I greatly appreciated.

To my beloved parents, Mustapha and Haliza who always support me through my upside down life as a student.

To my special ones, Nur Azwani and Nur Juwana Izzati, who always support me and guidance me for three years as a student.

To my supervisor, Puan Nor Mas Aina in giving suggestions and encouragement throughout my project.

To my supportive and helpful friends who always give support to me.

Thank You.

ACKNOWLEDGEMENT

Alhamdulillah, thanks to God for gratitude and blesses upon me in my Final Year Project entitled Malaysia Auxiliary Police Association (APA) Management System (Transport Module). I really grateful because managed to complete this final year project although I slightly miss much of the time given.

I would like to express my gratitude and appreciation to all who gave me the responsibility to complete this report. A big thank you to my supervisor, Puan Nor Mas Aina Md Bohari for the continuous support and giving suggestions and encouragement throughout my project.

I also would like to give my deepest thanks and appreciation to my parents and family for their endless support to complete this project, from beginning till the end. Also a special thanks and appreciation to my friend that has been helping me and guidance during my study.

Last but not least, thanks to people who help me directly or indirectly for the support and guidance given to improve the report produced and give more confidence in presentation skills by the comments and tips.

ABSTRACT

Auxiliary Police Association(APA) Management System (Transportation Module) is a web-based system that provides efficient management to manage transportation in this association. This system replaces the existing manual system that uses the form to register the transportation and distribute the transportation to the agencies. The objectives for this system are to change the manual system of transportation management into web-based management system. Besides this system also to provide a systematic management for the transportation. It also develop to save time during the registration of transport with computerized system. Project methadology that will be use is waterfall model. This system is very beneficial for Auxiliary Police Association (APA) because it will develop to ease them and make the system more efficient.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLE

List

LIST OF ABBREVIATIONS

DBMS	-	Database management system
DCL	-	Digital common language
DDL	-	Data definition language
DML	-	Data Manipulation Language
GUI	-	Graphical user interface
PSM	-	Projek Sarjana Muda
SDLC	-	System development life cycle
SQL	-	Structured Query Language

LIST OF ATTACHEMENTS

CHAPTER I

INTRODUCTION

1.1 Project Background

Malaysia Auxiliary Police Association was established in 1948 when the country was facing the emergency. In 1956, the Police Force was disbanded after the end of emergency aid and set back under the Police Act 1967. Now the Malaysia Auxiliary Police Association was growing and they have many agencies registered with them. Unfortunately, the management of this association is written in a book and form where the risk of data loss increases.

A proposed system, Malaysia Auxiliary Police Association (APA) Management System (Transportation Module) is a web-based system that provides efficient management to manage transport in this association. With this system, users

(admin, agency) can manage the transport from the association. This system replaces the existing manual system that uses the form to register the transportation and distribute the transportation to the agencies. Before this, agency use the manual system which is form based to register the transport that they got from the association. The manual processes cause many problems like data loss, damage and also it was waste the time to manage all the data. So with this system, the association and agency can easily manage the data without wondering the damaged and data loss.

Malaysia Auxiliary Police Association (APA) Management System (Transportation Module) is helpful for the agency to register and manage the transport that they got from Malaysia Auxiliary Police Association (APA). They can update the status of that transport. They can monitor where the transport are located. The agency can see the report of the transport either it active or not. Also, they can register their branch and manage it. For the admin of this system, they have full privilege in this system. Admin can control all the basic data in this system.

1.2 Problem Statements

1) Malaysia Auxiliary Police Association (APA) Management's current manual system is an inefficient use by the association. They use the paper to fill the form for every registration of transport. It is wasting resource.

2) Manual process need to go through multiple roles that are not so efficient and easily caused human errors which will lead to the difficult in handling the transportation. It also cause many problems like data loss, damage and also it was waste the time to manage all the data.

3) Difficult to keep track of the transport that they had and non-systematic data management.

1.3 Objective

1. To change the manual system of transportation management into web-based management system.
2. To provide a systematic management for the transportation.
3. To save time during the registration of transport with computerized system.
4. To provide a system that is user-friendly suitable for the targeted audience.
5. To produce report for transport in each agency.

1.4 Project Scope

There are three (3) parts of the scope of Auxiliary Police Association Management System which are user, module and software.

a) Scope for User

This system will be used by admin of the system and agency's auxiliary police association. Admin have the full privilege of this system. Agency used this system to register and manage the transport that they received from association.

b) Scope for Module

The focus on Malaysia Auxiliary Police Association (APA) Management System is to change the manually system to the web-based system. This system is used by the agency of auxiliary police. Agency need to register first to use this system and also register the branch that they have. All details about transport must be register in this system. This system also keeps the information about agency and its branch.

c) Scope for Software

This system will be created by using PHP language and Oracle 11g for it database.

1.5 Project Significant

Malaysia Auxiliary Police Association (APA) Management System (Transportation Module) is a web-based system that provides efficient management to manage transportation in this association. This system replaces the existing manual system that uses the form to register the transportation and distribute the transportation to the agencies. This system provides a systematic management for the transportation. It also developed to save time during the registration of transport with computerized system.

1.6 Conclusion

In conclusion, by developing this Malaysia Auxiliary Police Association (APA) Management System (Transportation Module), all the data of transport and agency will be recorded. So it can prevent from data loss and damaged. There is no more handwritten and printed data that user need to be passed up or keep. Users do not need to fill up the form register the transport that they received.

In the next chapter, it will discuss about project methodology and planning. We will see the database development methodology and what model of methodology used for this system.

CHAPTER II

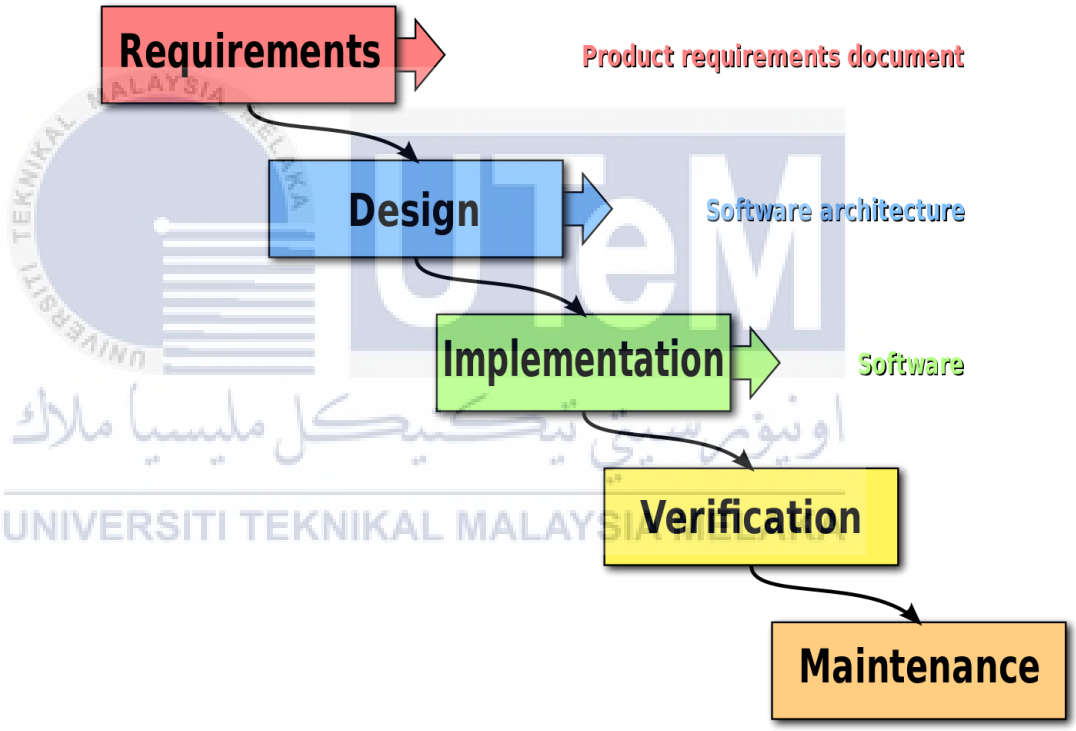
PROJECT METHODOLOGY AND PLANNING

2.1 Introduction

Methodology is the systematic, theoretical analysis of the methods applied to a field of study. It comprises the theoretical analysis of the body of methods and principles associated with a branch of knowledge. This chapter will explain the details of project methodology and planning that will use as a guide to manage the project from start to finish. In order to evaluate this project, the methodology used based on System Development Life Cycle (SDLC).

2.2 Database Development Methodology

For this system, Waterfall Model is being used as a System Development Life Cycle (SDLC) approach. . It is very simple to understand and use. Each phase must be completed before the next phase can begin and there is no overlapping in the phases. . The next phase is started only after the defined set of goals are achieved for the previous phase and it is signed off, so the name "Waterfall Model". In this model, phases do not overlap.



2.2.1 Database Planning

The database planning includes the activities that allow the stages of the database system development lifecycle to be realized as efficiently and effectively. Effective database planning means that the system is capable of managing and consolidating all the data generated and relied upon by the business plan. This phase must be integrated

with the overall Information System of the organization. The very first step in database planning is to define the mission statement and objectives for the database system.

2.2.2 System Definition

In the system definition phase, the scope and the boundaries of the database application are described. The description includes to what we planned for the system in the future and include who is the stakeholders for the system.

2.2.3 Database Design

Database design consists of three phase which are conceptual database design, logical database design and physical database design. In the conceptual database design phase, the model is based on the requirements specification of the system. In the logical database design phase, the model of the data to be used based on a specific data model. In the physical database design phase, data definition language (DDL) statement will be design, for the better implementation.

2.2.4 Implementation and Loading

In this phase, which is implementation and loading stage, the database management system (DBMS) was been chosen. There are a few types of DBMS like Access, SQL Server, MySQL and Oracle. The criteria for the used DBMS are defined. Then the DBMS will be installed and the database will be created. The data that are required will be loaded into database. Design of GUI will be defined and designed.

2.2.5 Testing

Before the new system is going to live, it should be thoroughly tested. That is to ensure that the system can performs and be used by the users as expected without any error. This phase is also to find the errors of the system, so the developer can fix it before it will be used by the users.

2.2.6 Operation Maintenance

After testing and evaluation stage, it will move to operational stage. The operational maintenance is the process of monitoring and maintaining the database system. Maintaining and upgrading the database system means that, when new requirements arise, the new development lifecycle will be done.

2.3 Project Schedule and Milestones

Table 2.1: Project Schedule and Milestone

Milestones	**Expected Documents**	**Dates(Weeks)**
Proposal correction and submission	Proposal	13/2/2017 – 26/2/2017 (Week 1 & 2)
Chapter 1	Objectives, Scope, Project Significance	20/2/2017 – 12/3/2017 (Week 2 – 4)
Chapter 2	Project Methodology, Project Schedule and Milestones	6/3/2017 – 26/3/2017 (Week 4 – 6)
Chapter 3	Functional Requirements, Non-	20/3/2017 – 2/4/2017

	Fucntional Requirements	(Week 6 – 7)
Project Demo	-	27/3/2017 – 2/4/2017 (Week 7)
Chapter 4	ERD, Data Dictionary, Normalization, Queries, Graphical User Interface (GUI)	27/3/2017 – 23/4/2017 (Week 7 – 10)
Project Demo	-	17/4/2017 – 21/5/2017 (Week 9 – 14)
PSM Report	Chapter 1 – Chapter 4	1/5/2017 – 21/5/2017 (Week 12 – 14)
Final Presentation	-	22/5/2017 – 28/5/2017 (Week 15)
Correction Draft Report	Chpater 1 – Chapter 4	29/5/2017 – 4/6/2017 (Week 16)

2.4 Conclusion

As a conclusion, every project will have a different methodology that is being used to make the project successful and working well. Selecting System Development Life Cycle (SDLC) approach could be tricky if it is not suitable. Thus, this SDLC is the most suitable for my project as it has to be test out several times.

Based on Database Development Life Cycle(DBLC) , the next phase after development methodology and planning is analysis phase. Analysis phase will be discuss in the next chapter.

CHAPTER III

ANALYSIS

3.1 Introduction

After planning phase is done, system analysis will take part for this project documentation. To proceed with this phase, collecting data from the user and investigation technique be used in order to simplify work on identify required data.

Data requirement is important in collecting data that is required in develop the system. Functional requirement is included the functions that available in the system while non-functional requirement is included other than functional requirement.

As for the analysis part, it will cover on problem and requirement analysis. Requirement analysis on this chapter will cover on data requirement, functional

requirement, non-functional requirement and others. The purpose of this chapter is to identify the requirements for Auxiliary Police Association (APA) System.

3.2 Problem Analysis

Based on analysis, current situation shows the problem of agency's Auxiliary Police Association to register the transport by used manual system. So, it difficult to know each data of transport that they have. They also have difficulties in maintaining the data of transport. By developing this system, they only need to register their own transport by using online system. Besides that, they can know how many transport they have and all data about the transport. They also can update the status of transport and view the report of transport.

3.3 The proposed improvements/solutions

The proposed flow system for Malaysia Auxiliary Police Association Management System (Transport Module) is divided into two users which are the admin of the system and agency of Auxiliary Police Association.

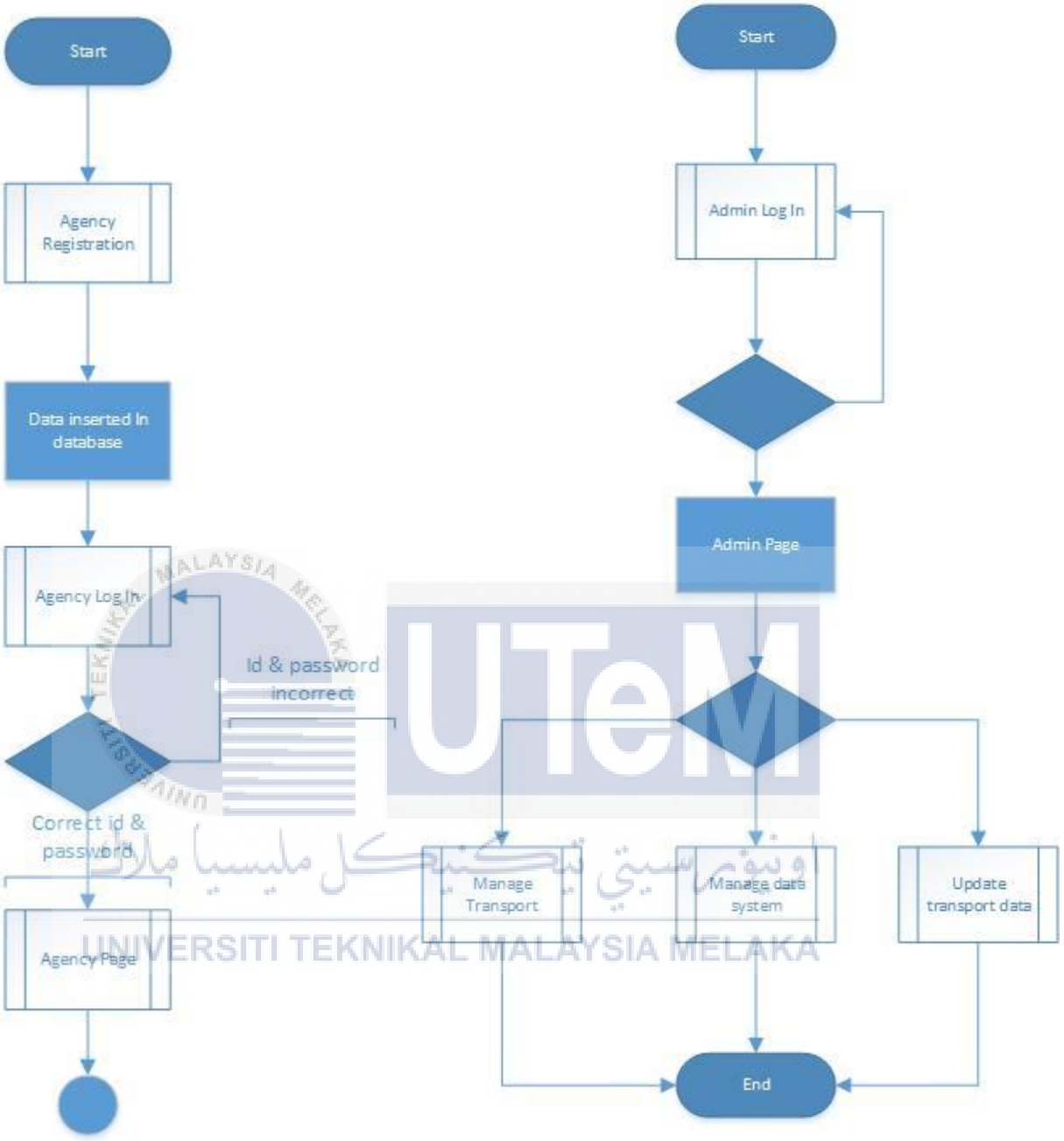


Figure 3.2: Agency Page

Figure 3.1: Admin Page

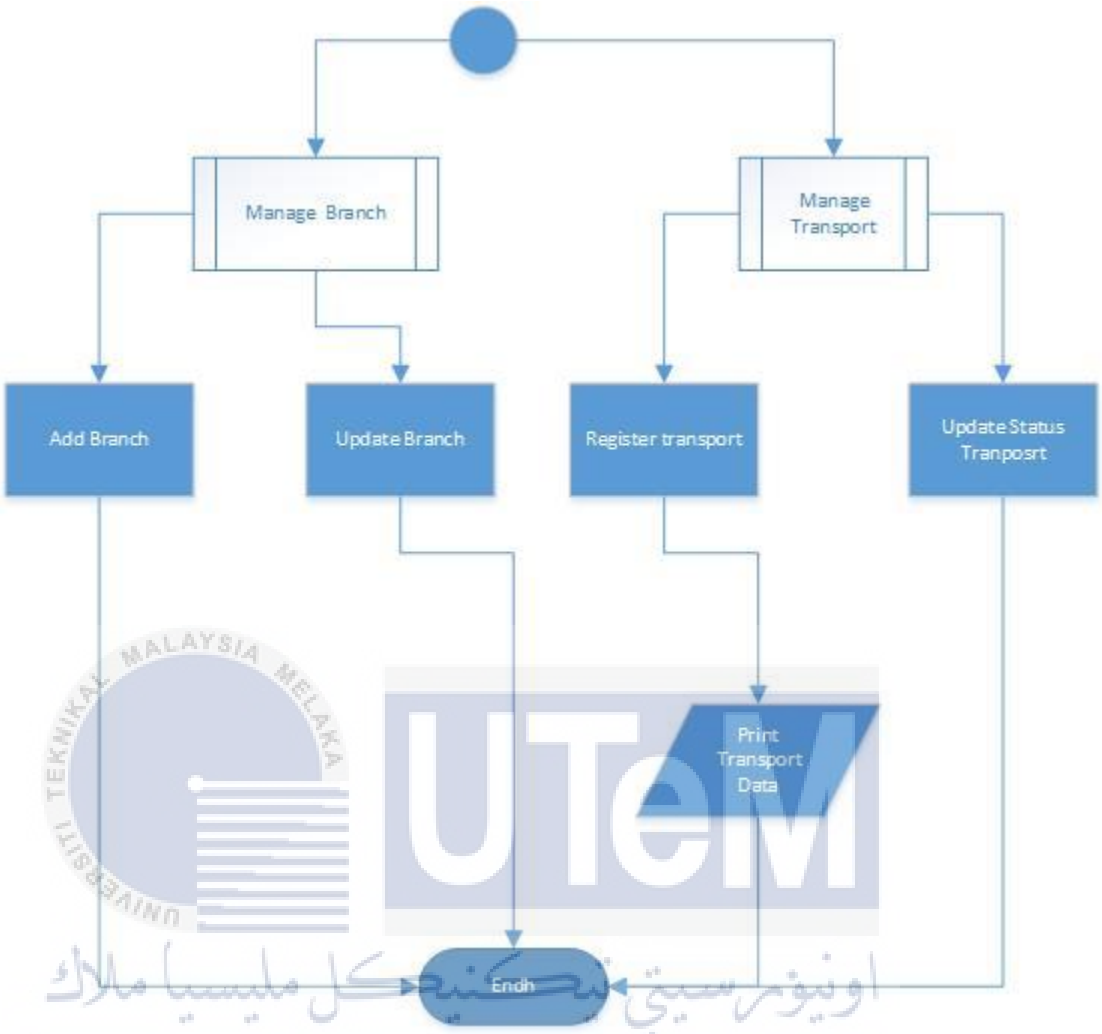


Figure 3.3: Agency Page2

3.4 Requirement Analysis

Data requirement, functional requirement, non-functional requirement and another requirement will be cover in requirement analysis. Contact diagram and data flow diagram of APA Management System will be included in functional requirement. In non-functional requirement, there are 3 main topics will be discuss which are product requirements, organizational requirement and network requirement will be state in other requirement.

3.4.1 Functional Requirement

In this section, it will describe about interaction of the component system. It will describe what the user functionality in this system and how the system should operate. The UML diagram is being used to explain the whole function of this system.

Use case diagram will represent the whole function of this system. Figure below shows the use case diagram for Malaysia Auxiliary Police Association Management System (Transport Module). This system consists of two external entities which are admin and agency of auxiliary police.

3.4.2 Non-Functional Requirement

In this part, non-functional requirements specified the system quality attributes or characteristics such as below:

1. Only the authorized users can access in this system.
2. There is no limit time for users to use the system
3. This system can be functional using web-browser.

3.4.3 Others Requirement

3.4.3.1. Software Requirement

Table 3.1: Software Requirement

SOFTWARE	PURPOSE
Sublime Text3	To write a coding for the system
Oracle 11g	As a database used for the system

Microsoft Visio 2016	To build Entity-Relationship Diagram(ERD) and flow chart
Start UML	To create use case for the system
Adobe Illustrator CS6	To create a poster for PSM showcase

3.4.3.2. Hardware Requirement

Table 3.2: Hardware Requirement

HARDWARE	PURPOSE
Laptop	To keep all information regarding project
Printer	To print the documents and reports

Programming Language, Operating System and Database Management System (DBMS)

Table 3.3: Operating System

PROGRAMMING LANGUAGE	OPERATING SYSTEM	DATABASE MANAGEMENT SYSTEM (DBMS)
PHP	Windows 7 Ultimate	Oracle

3.5 Conclusion

In conclusion, this chapter explains how the system operates. Besides that, it explains how the data flow for each process in this system including requirement need such as functional, non-functional and others requirement.

As for the next chapter which design, this chapter will explain briefly about the proposed design for the system.

CHAPTER IV

DESIGN

4.1 Introduction

Design is the most crucial phase of system development included conceptual design, logical design and physical design. This chapter is divided into three (3) designs which are database design, conceptual design and physical interface. Each of this design process is important in order to develop a system according to the user requirements. It will help in develop an interactive system.

Conceptual design will explain about the Entity Relationship Diagram (ERD) and the business rule for the diagram. As for the logical design, data dictionary will be included and validate the conceptual design using transaction pathway. It also will describe about the types of query used in the Auxiliary Police Association System.

Whereas for the physical design will describe briefly on the selection of Database Management System (DBMS). In this physical design also explain about the usage of stored procedure, trigger and security mechanism in this system.

4.2 Database Design

Database design is the process of producing a detailed data model of database. This data model contains all the needed conceptual, logical and physical design choice. The architecture of the system will help in understanding the function of the system. The developer of a development system has to make user that the system architecture meets user requirement and provide an interactive interface for the system.

4.2.1 Conceptual Design

The conceptual design will explain on how the system should work based on the requirements. **Figure 4.1** shows the Entity-Relationship Diagram (ERD) for Malaysia Auxiliary Police Association Management System (Transportation Module).

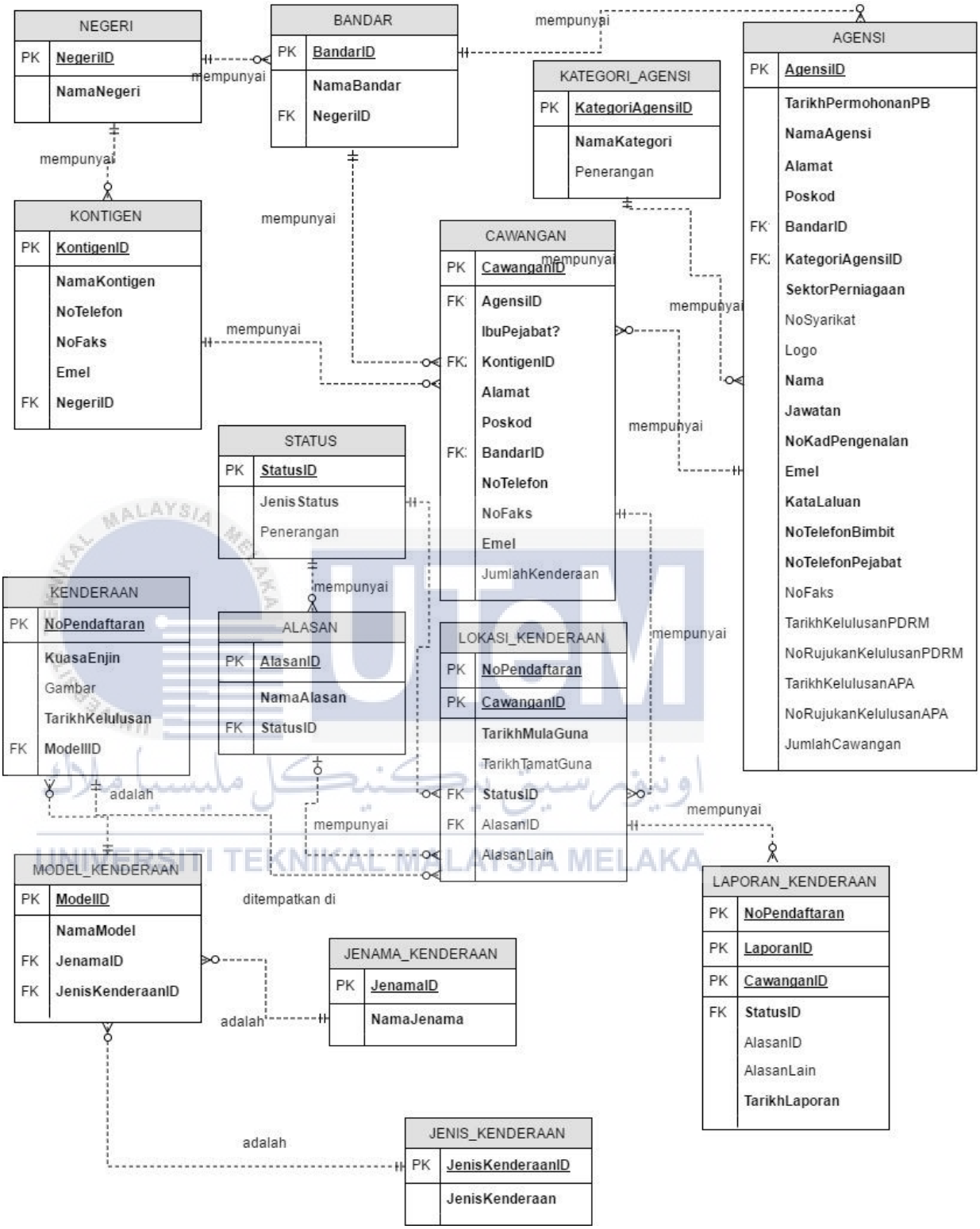


Figure 4.1: Entity-Relationship Diagram (ERD)

Figure 4.1 will be described in this business rule:

Business Rule

1. An agency can have one category and category of agency can have many agencies.
2. An agency can have many branches and a branch can only have one and only one agency.
3. Each city has zero or many agency and each agency has one and only one city.
4. A branch can have one and only one contingency and a contingency can have zero or many branches.
5. Each branch is located at only one city and each city has zero or many branches.
6. A branch has zero or many transports at a location and a transport for each location located at only one branch.
7. At a location located zero or many transport and the transport are located at one location at one time.
8. A transport can only have one model and a model can have zero or many transports.
9. Each type of transport has zero or many models and each model has one and only one type of transport.
10. Each transport brand has zero or many models and each model can only have one and only one transport model.
11. Each transport at a location must have only one status at one time and each status can be used by zero or many transport at one time.
12. Each transport at a location can have one or no reason at one time and each reason are used by zero or many transport.
13. Each status has zero or many reason and each reason has one and only one status.
14. Each transport at a location has zero or many report and each report has one and only one transport at a location

15. Each city has one and only one state and each state has zero or many city.

16. Each state has zero or many contingency and each contingency is located at one and only one state.

4.2.2 Logical Design

In this section, it will describe data dictionary and query design of this system. The validation of the conceptual design is also shows in this section. There are two types of query design use in this logical design phase. There are in the form of relational algebra or SQL statement.

The data dictionary will describe more details in the table below. The rest of table can be refers to **Appendix A.**

4.2.2.1 Data Dictionary

Table 4.1: State

Attribute	Content	Type	Domain	Required	PK/FK	Referred Table
NegeriID	State unique ID	INT (10)		Y	PK	
NamaNegeri	State name	VARCHAR(40)		Y		

Table 4.3: Contingency

Attribute	Content	Type	Domain	Required	PK/FK	Referred Table
KontingenID	Contingency unique ID	INT (10)		Y	PK	
NamaKontingen	Contingency name	VARCHAR(50)		Y		
NoTelefon	Telephone number	VARCHAR(15)		Y		
NoFaks	Faks number	VARCHAR(15)		Y		
Emel	Email of contingency	VARCHAR(40)				
NegeriID	State ID	INT(10)		Y	FK	State

Table 4.4: Brand

Attribute	Content	Type	Domain	Required	PK/FK	Referred Table
JenamaID	Transport's brand unique ID	INT (10)		Y	PK	
NamaJenama	Transport's brand name	VARCHAR(50)		Y		

Table 4.5: Transport model

Attribute	Content	Type	Domain	Required	PK/FK	Referred Table
ModelID	Transport model unique ID	INT (10)		Y	PK	
NamaModel	Model name of transport	VARCHAR(50)		Y		
JenamaID	Brand of transport Unique ID	INT (10)		Y	FK	Brand
JenisKenderaanID	Type of transport Unique ID	INT (10)		Y	FK	Type of transport

Table 4.6: Transport Location

Attribute	Content	Type	Domain	Required	PK/FK	Referred Table
NoPendaftaran	Type of transport unique ID	VARCHAR(10)		Y	PK/FK	Transport
CawanganID	Branch ID	INT(15)		Y	PK/FK	Branch
TarikhMulaGuna	Start date of use the transport	DATE		Y		
TarikhTamatGuna	End date of use the transport	DATE				
StatusID	Status of the transport			Y	FK	Status
AlasanID	Reason of the status				FK	Reason
AlasanLain	Another reason of the status	VARCHAR(50)				

4.2.2.2 Transaction Pathway

The user transaction is used to validate the conceptual design. It is using transaction pathway. The list of query below is to build a transaction pathway to validate the conceptual design.

1) List all the branch that agency have.
2) How many transport that branch ‘A1021’ have?
3) List all the transportation having by each agency.
4) List all type of transport.
5) List all model of transport.

4.2.2.2 Query Design

A query is a request for information from a database. Queries are very useful tools when it comes to databases and they are often called by the user through a form. They can be used to search for and grab data from one or more of your tables, perform certain actions on the database and even carryout a variety of calculations depending on your needs. There are several type of queries in database which are basic queries, aggregate queries, subqueries and join queries.

a) Basic Queries

List all type of transport in ascending order.

SQL Statement

```
SELECT *
FROM JENIS_KENDERAAN
ORDER BY JENISKENDERAAN ASC;
```

Relational Algebra

```
Π jeniskenderaanid, jeniskenderaan(JENIS_KENDERAAN)
```

List all the transport.

SQL Statement

```
SELECT *
FROM KENDERAAN
```

Relational Algebra

```
Π(KENDERAAN)
```

b) Aggregate queries

Calculate total of transport in an agency A1021

Sql Statement

```
SELECT sum(jumlahkenderaan) as Jumlah_Kenderaan

FROM cawangan

WHERE agensiid='A1021';
```

Relational Algebra

```
F SUM(jumlahkenderaan) (σ
agensiid='A1021')(cawangan)).
```

List type of transport that have name like 'Mo'

Sql Statement

```
SELECT *

FROM JENIS_KENDERAAN

WHERE JENISKENDERAAN LIKE '%'||JnsKenderaan||'%';
```

Relational Algebra

```
Πjeniskenderaanid,jeniskenderaan (σ agensiid like 'Mo'
(jenis_kederaan)).
```

c) Join Queries

List transport plate number, transport brand, transport model, start date of use transport from agency.

Sql Statement

```
SELECT L.NOPENDAFTARAN AS PLATENO, P.NAMAJENAMA AS
JNM, M.NAMAMODEL AS MDL, L.TARIKHMULAGUNA AS
MULAGUNA ,I.NAMACAWANGAN ,A.NAMAAGENSI

FROM LOKASI_KENDERAAN L, JENAMA_KENDERAAN P,
MODEL_KENDERAAN M, KENDERAAN C, CAWANGAN I, AGENSI A

WHERE L.NOPENDAFTARAN = C.NOPENDAFTARAN AND
C.MODELID=M.MODELID AND M.JENAMAID=P.JENAMAID AND
L.CAWANGANID= I.CAWANGANID

AND I.AGENSIID = A.AGENSIID AND A.AGENSIID='A1021';
```


4.2.3 Physical Design

In the physical design process, the data gathered during logical design phase are convert into a description of the physical design including tables and constraints. Physical design decisions, such as the type of index or partitioning have a large impact on query performance.

The DBMS choose for the APA system is Oracle. This DBMS is perceived to be able to support enormous databases and it is more reliable. Oracle database can handle a huge amount of data and perform well under load. It also helps the developer to create

customized database applications. The DBMS is used to design a coding for database task as the planning system. The coding constructs in DBMS such a simple coding, trigger and stored procedure.

A trigger is a special kind of stored procedure that automatically executes when an event occurs in the database server. In this system, we used before insert and after insert trigger. Before insert trigger is used for the primary key for each table as example the username for the agency are auto insert when an agency register to this system. For the after insert trigger are used in calculated the total transport that agency have. Below some example of trigger that has been used in this system.

1) Before Insert Trigger

Table: Transport brand

This before insert trigger is used for auto increment the transportation type unique id.

Table: Agency

This before insert trigger is used for auto increment the agency unique id by adding 'A' in front of the id. For example 'A1001'.

2) After Insert Trigger

Table: Branch and agency

This after insert trigger is used for update the total transport for each agency after register the transport in each branch for the agency.

A stored procedure is a group of SQL statements that form a logical unit and perform a particular task, and they are used to encapsulate a set of operations or queries to execute on a database server. Stored procedure can manage and maintain data easily. The benefits of using stored procedure are they allow modular programming, faster execution and can reduce network traffic. Stored procedure also can be used as a

security mechanism. The example of stored procedure that has been used in this system are shows below.

1) Stored Procedure for Insert

Table: Transport Type

Function: The stored procedure used to insert the transport type into its table when admin register the transport type.

Table: Transport Model

Function: The stored procedure used to insert the transport model during the registration of transport brand.

2) Stored Procedure for Select

Table: Transport Type

Function: This stored procedure is to select transport type where the type is like the input that inserted.

Table: Branch

Function: This stored procedure is to list all the branches from their agency.

3) Stored Procedure for Update

Table: Transport Type

Function: This stored procedure is to update the transport type for each transport.

4.3 Graphical User Interface (GUI) Design

The module that will be used in the interface will explain in this section. Some of the function on this system such as register, login and others will be explain. The other graphical user interface can refer to **Appendix A.**

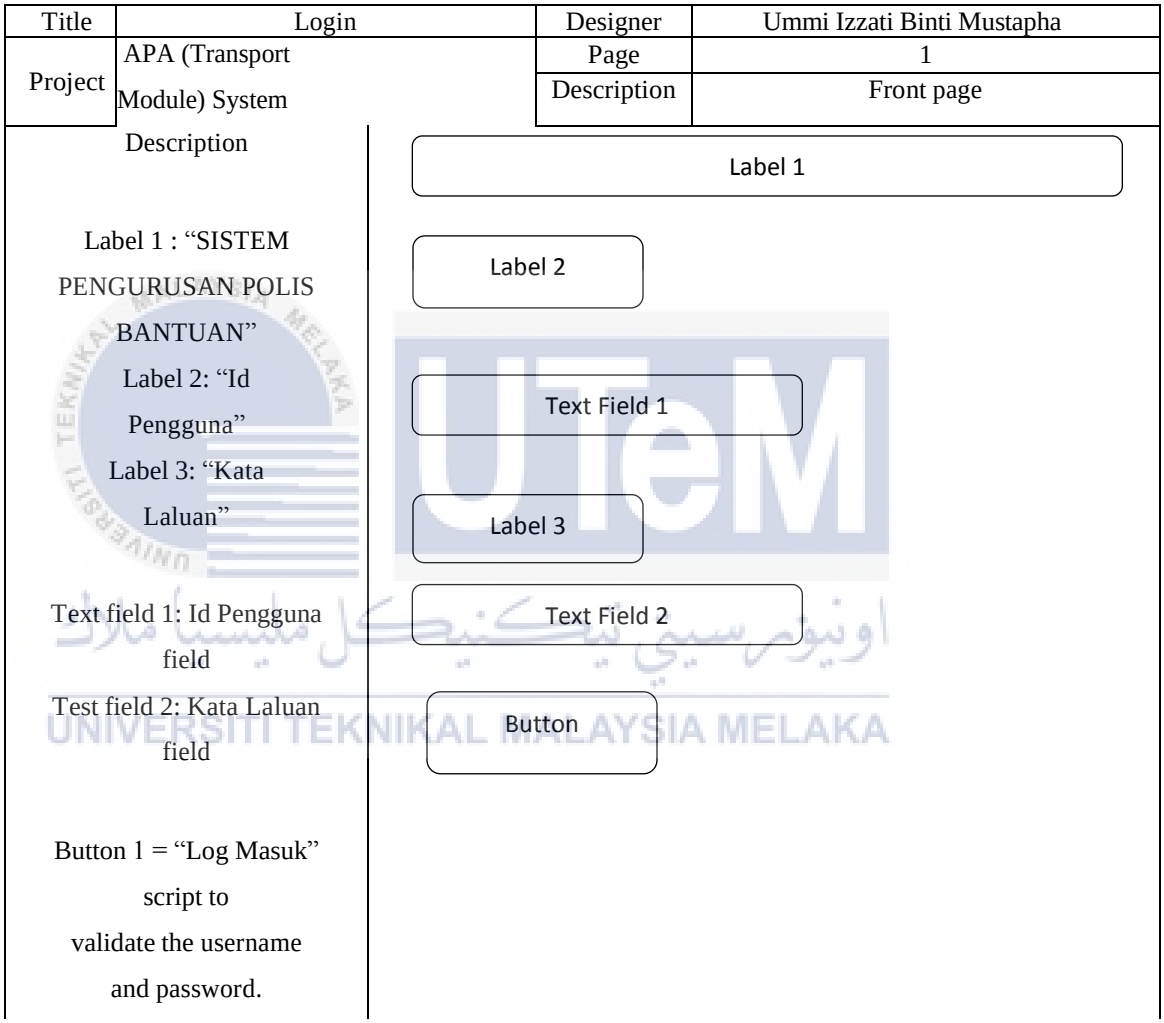


Figure 4.2: Login interface

Title	Admin Interface	Designer	Ummi Izzati Binti Mustapha
Project	APA (Transport Module) System	Page	2
		Description	Admin page

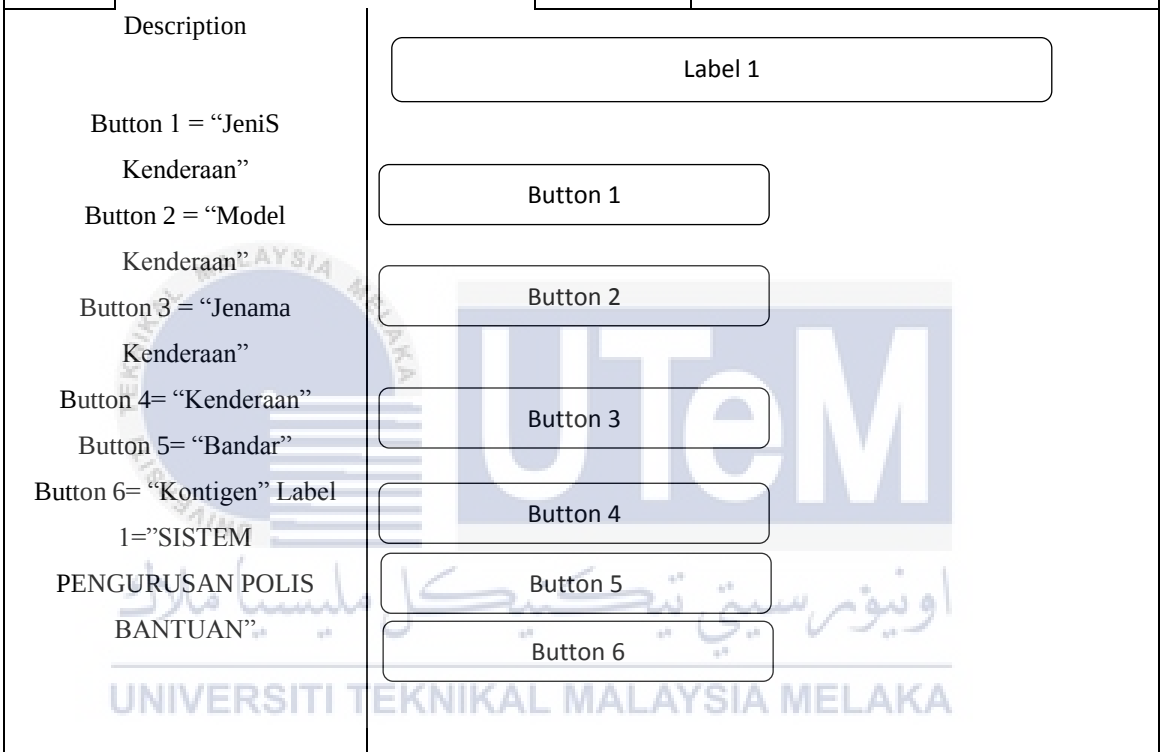


Figure 4.3: Admin interface

Title	Transport Type	Designer	Ummi Izzati Binti Mustapha
Project	APA (Transport Module) System	Page	3
		Description	Jenis Kenderaan page

Description

Label 1 : “JENIS KENDERAAN”

Text field 1: Jenis Kenderaan field

Button 1 = “Cari” button to Search the jenis kenderaan.

Label 2: “Jenis Kenderaan Id”

Label 3: “Jenis Kenderaan”

Button 2= “Kemaskini” script to update the “Jenis Kenderaan”.

Data=the data of “Jenis Kenderaan”

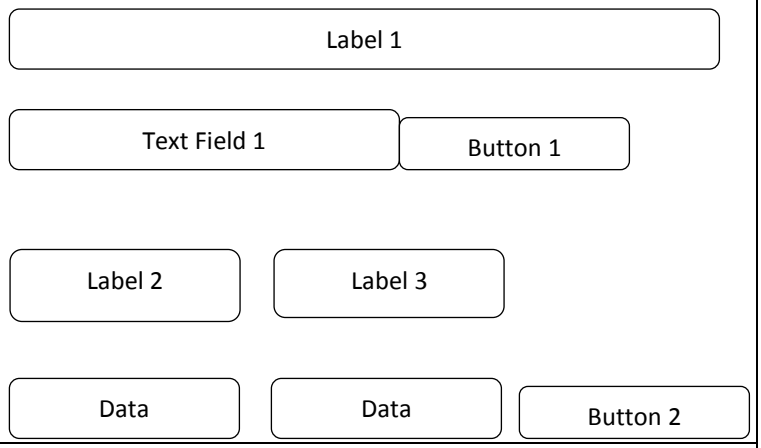


Figure 4.4: Transport Type interface

4.4 Conclusion

In conclusion, this chapter discussed more about how design of the system is being built which include all type of design such as conceptual design, logical, physical and user interface. Each type of design provided different structure. An Entity

Relationship Diagram (ERD), business rule, data dictionary, query design and GUI are used to help in order to create syntax and figures in creating the database structure of Auxiliary Police Association Management System (Transportation Module). The ERD has been validated by using transaction path. The business rule will explain more details about the diagram. For the trigger and stored procedure, the example of them will be show in the testing chapter.

Chapter V will handle about implementation phase. It involves installation steps-by-steps of the database and software requirements for using PHP programming language. Implementation step is the third step in DBLC based on waterfall model.

CHAPTER V

IMPLEMENTATION

5.1 Introduction

This chapter is about implementation phase. In this phase, the system is installed and made operational in the production environment after the system and users' acceptance testing. Activities in this phase include efforts required for implementation including the installation step and starting the web application server and database services. Besides that, this chapter will explain on the database implementation which includes the DDL statements in the selected DBMS which Oracle database. The main processes such as the use of stored procedures and triggers in this system will be show in this chapter.

5.2 System Development Environment Setup

This chapter will explain the initial setup of the Malaysia Auxiliary Police Association Management System (Transportation Module). There are many types of web server that can be used as server to a system. For this system it will be using Apache Web Server services and runs it as localhost mode. It is because Apache is the most widely used web server software and it is an open source software available for free. Meanwhile, Oracle is used as the database server. All the data needed by the system are stored in the database.

5.2.1 System Environment Setup

The software has been used to setup the system environment is XAMPP. XAMPP is stand for Cross-Platform (X), Apache (A), MariaDB (M), PHP (P) and Perl (P). XAMPP can be downloaded freely from the *https://www.apachefriends.org/*. The step of installation XAMPP are as shown.

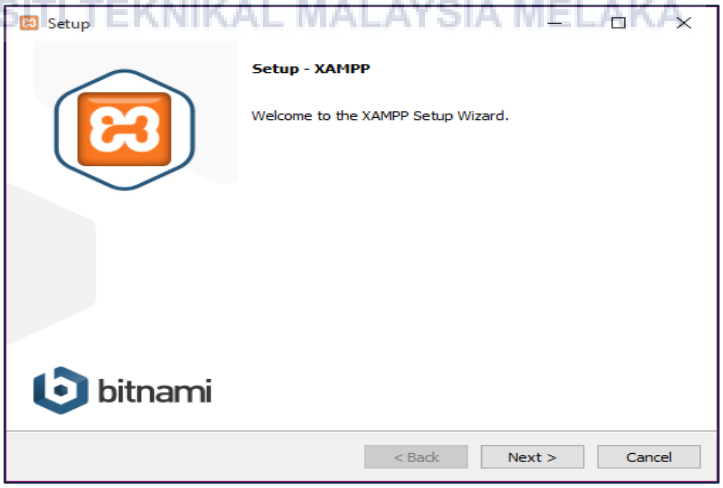


Figure 5.1: XAMPP Setup

In the **Figure 5.1**, it can be seen right after the XAMPP installer has been executed. Users have to click on ‘Next’ button to start the installation.

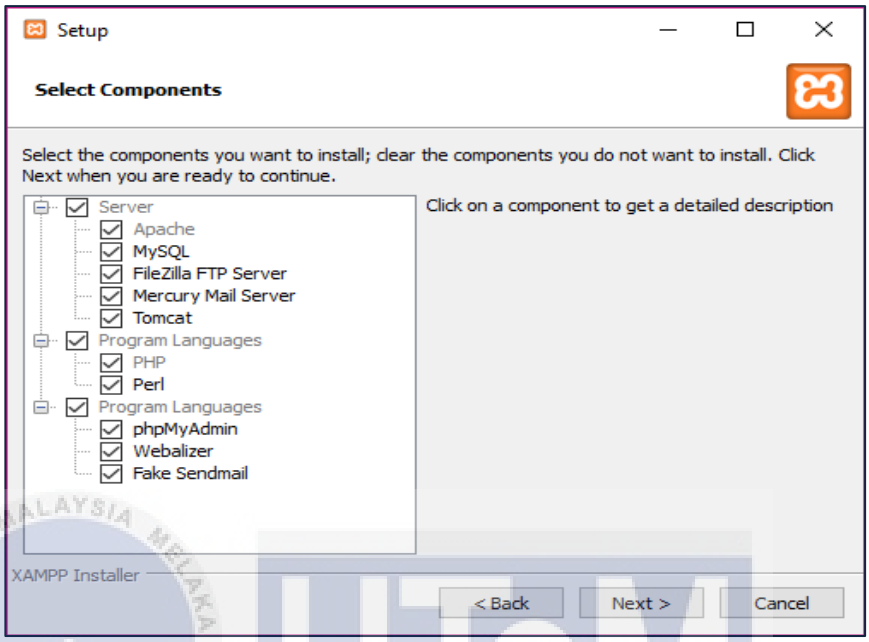


Figure 5.2: XAMPP Components

Prompt shown in **Figure 5.2** will appear. This will show the component you want to install. Click next when the component were selected.

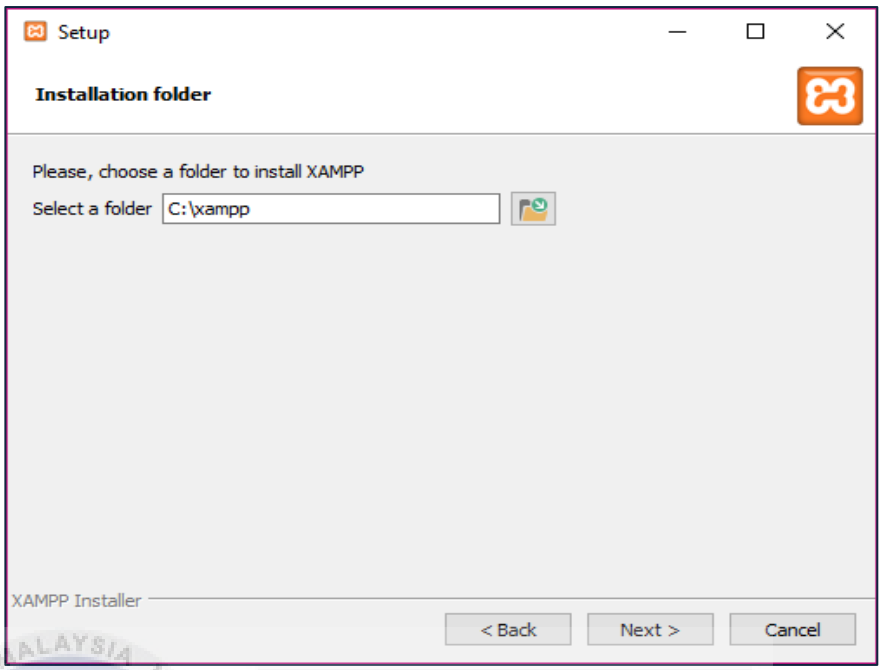


Figure 5.3: Installation Folder

From **Figure 5.3,** the installation is prompting to choose a folder for the XAMPP should be installed. As default, the installation will be placed on C:/xampp folder location.

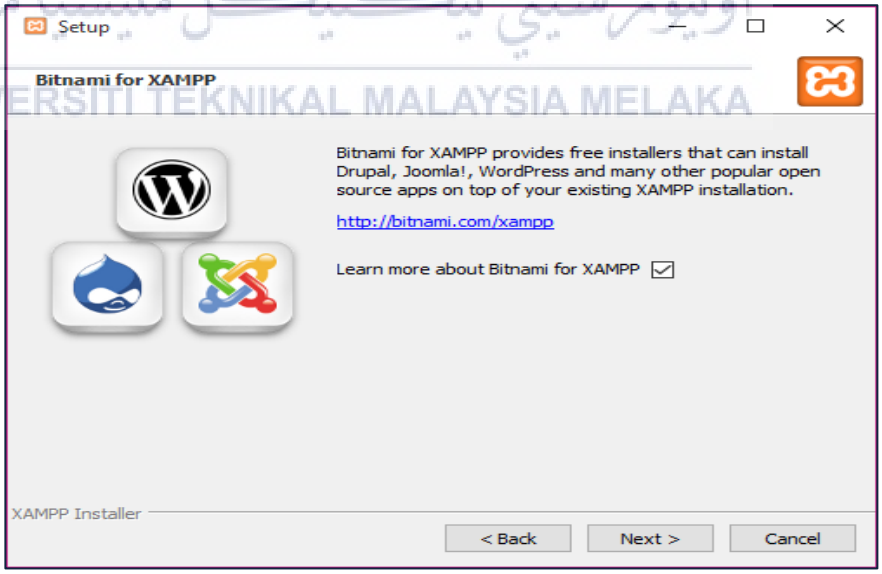


Figure 5.4: Bitnami for XAMPP

In the **Figure 5.4**, it is the step to continue the installation after all the required settings have been done. It will take several minutes to completely install the XAMPP application.

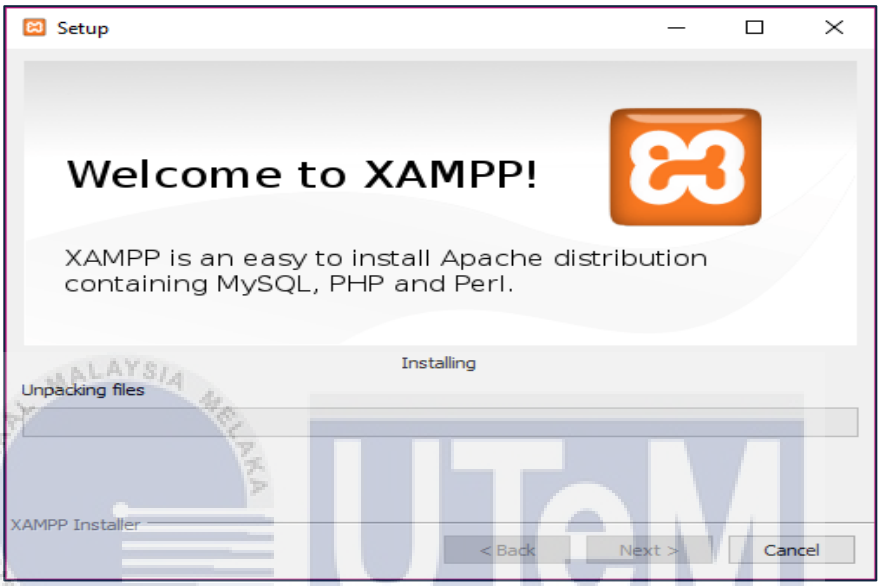


Figure 5.5: Installation Progress

Figure 5.5 shown XAMPP will be install into laptop. Wait until the process finish and click next.

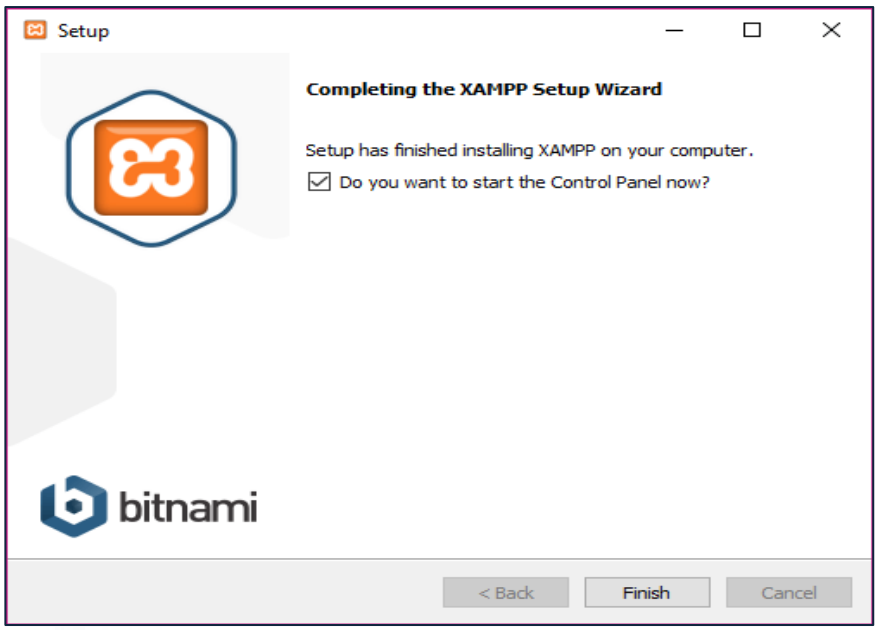


Figure 5.6: Finish XAMPP Installation

Figure 5.6 shown has stated that the XAMPP application has been installed successfully. It a prompts to start the XAMPP Control Panel, so that the XAMPP can start to run its services.

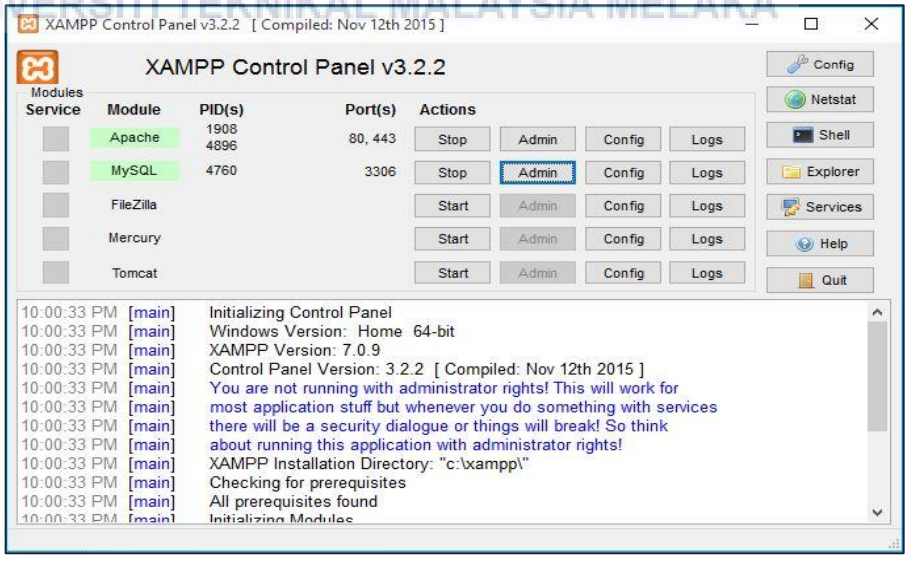


Figure 5.7: Control panel to start the services

From what has been shown in **Figure 5.7**, the control panel will either start or stop the services anytime the user wants to. It is displaying the port number used by each services and the services activity log from time to time.

5.2.2 Database Environment Setup

The DBMS chosen in this system is Oracle 11g. The installation steps are:

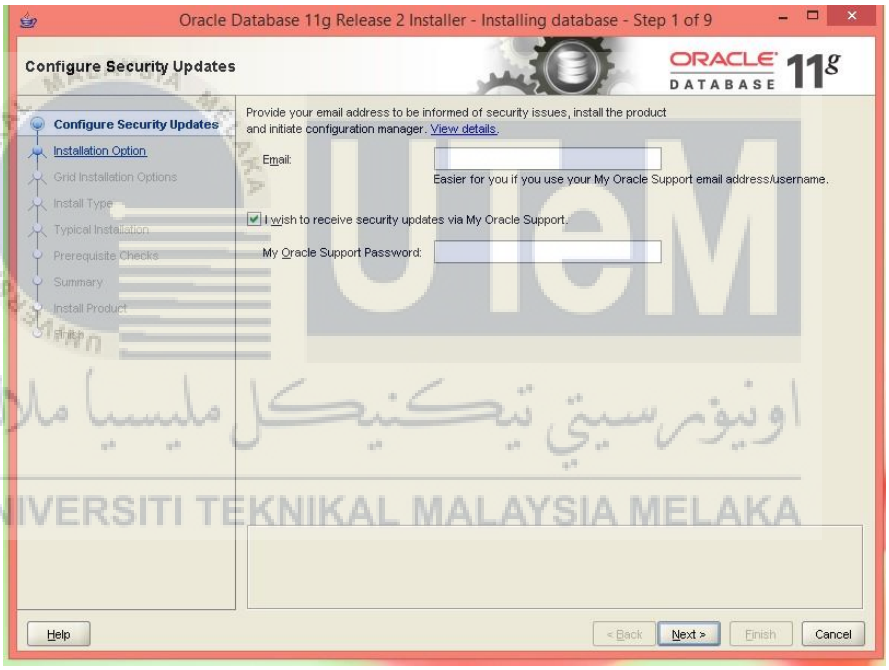


Figure 5.8: Configure Security Update

In the **Figure 5.8**, it can be seen right after the Oracle 11g installer has been executed. Users have to insert email and oracle support password.

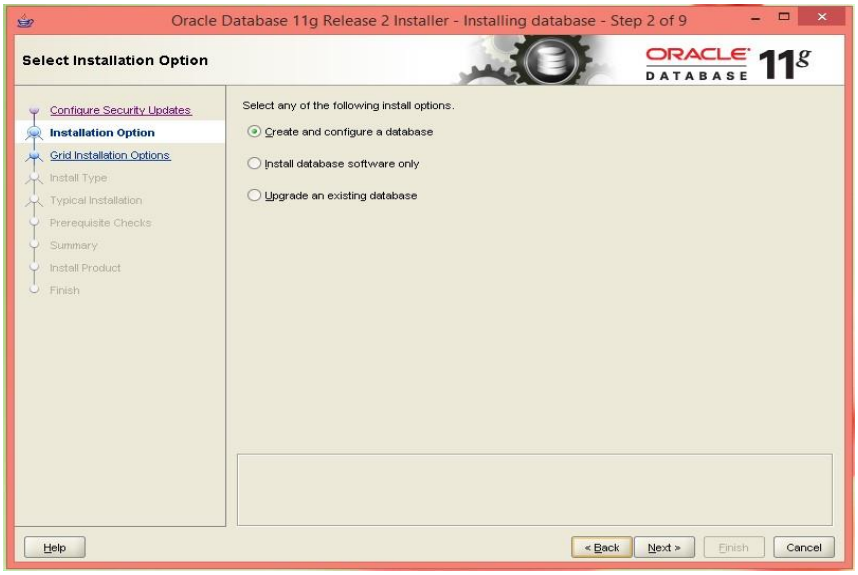


Figure 5.9: Installation Option

From **Figure 5.9**, it is prompting the selection of installation option. We have to choose create and configure database and then click next.

.

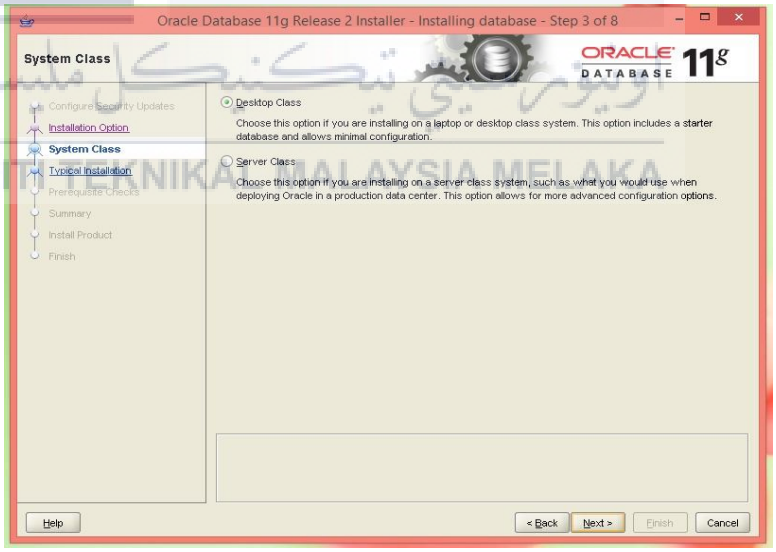


Figure 5.10: System Class

In this **Figure 5.10**, we have to select Desktop Class for system class that will be use and click next

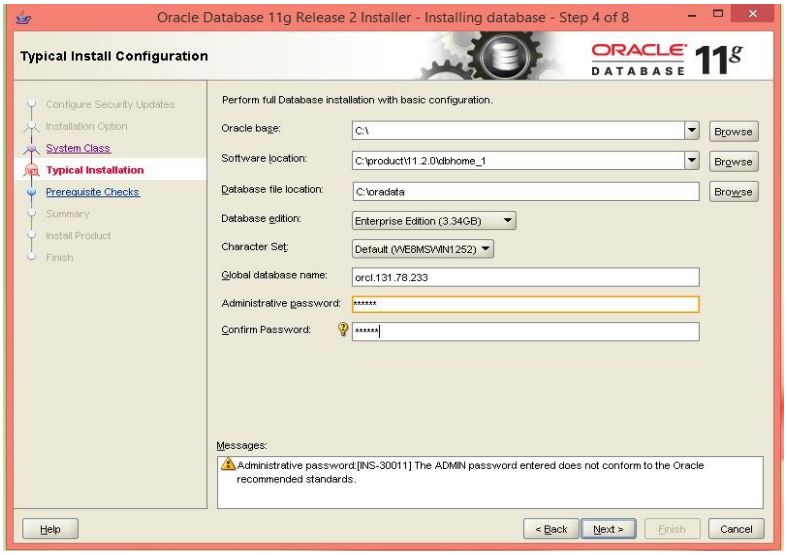


Figure 5.11: Typical Installation

From **Figure 5.11,** the installation is prompting to choose a folder for the Oracle 11g should be installed. As default, the installation will be placed on C:/oracle folder location. It also required administrative password.

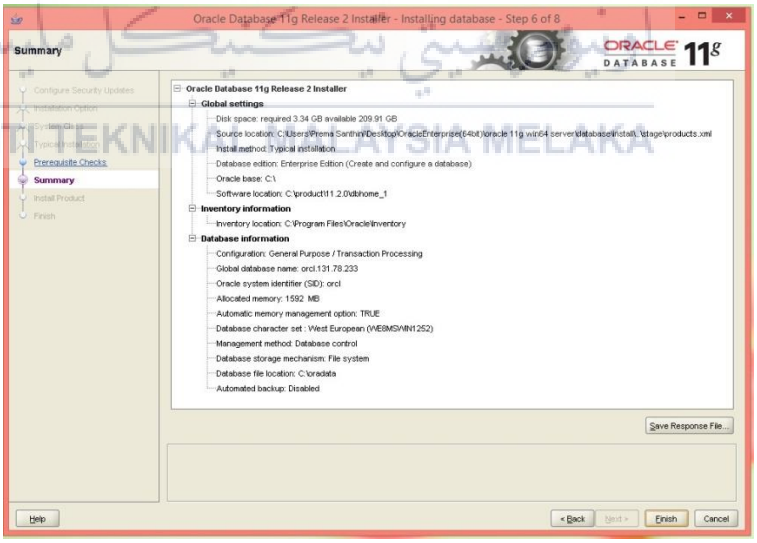


Figure 5.12: Summary

Figure 5.12 shows summary of the installation. If there is an error, click back, if not click finish.

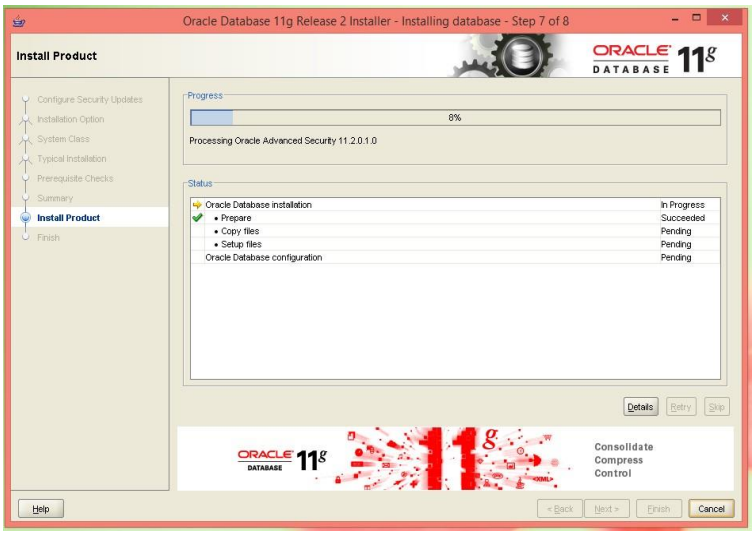


Figure 5.13: Install Product

Figure 5.13 is prompting the install product. Oracle product will be installed into laptop. Wait until the progress 100% and click finish.

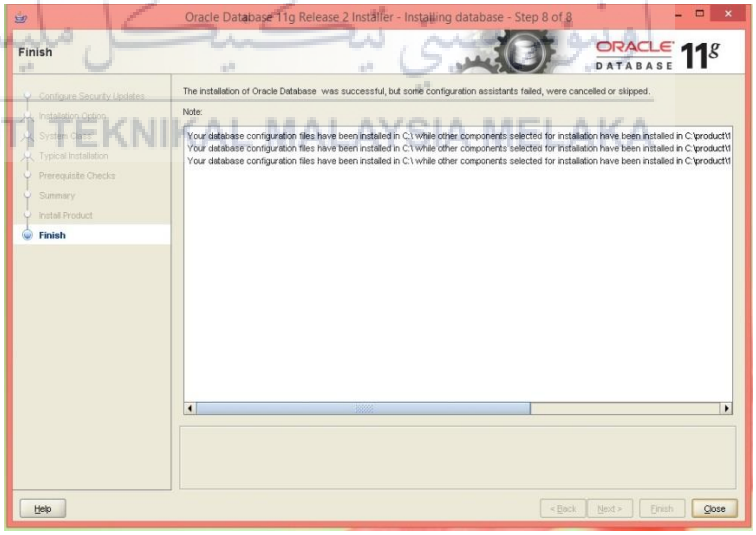


Figure 5.14: Complete Installation

Prompt such as **Figure 5.14** will be shown after click finish and click close.

5.2.3 Database Creation

A database is a collection of information that is organized so that it can be easily accessed, managed and updated. Data is organized into rows, columns and tables, and it is indexed to make it easier to find relevant information. Database also perform tasks such as data analysis, data storage and data manipulation.

The database that has been used in this system is Oracle. It used Oracle SQL Developer for develop database for this system. Oracle SQL Developer is an integrated development environment (IDE) for working with SQL in Oracle database. It is a reliable platform to do tasks like retrieving, creating, update and delete the database objects. It also able to execute SQL statements and scripts to manipulate database objects.

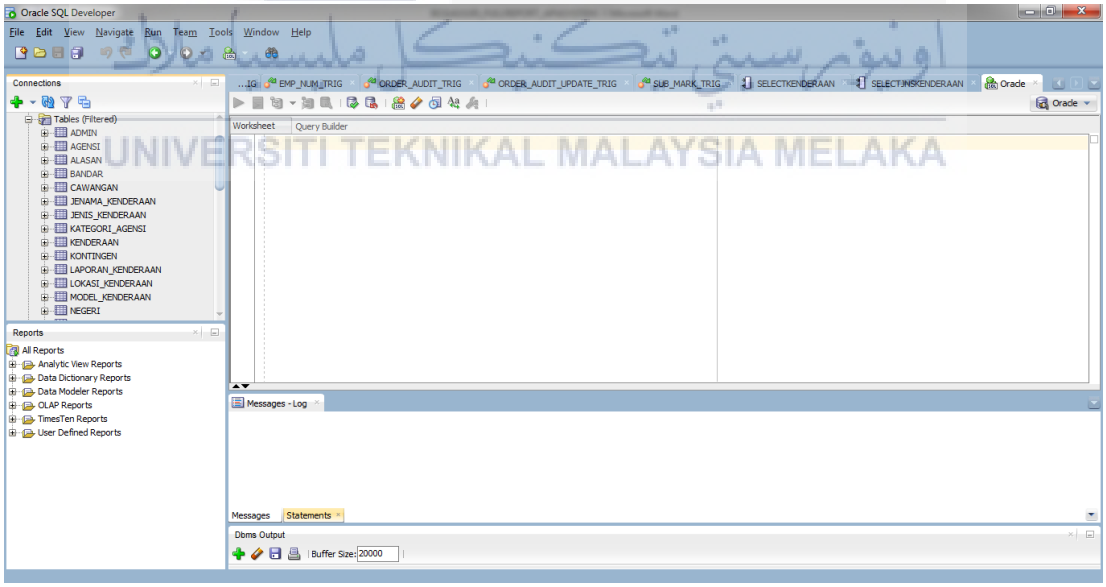


Figure 5.15: SQL Developer Interface

From the **Figure 5.15**, it is showing the interface of Oracle SQL Developer. It is an interface to set up database and manage database which includes creating tables, stored procedures and triggers. It can also configure on the database user roles and privileges.

5.3 Database Implementation

This section will describe about how Oracle query and command are being implemented to access the database during the development of the system. In this database implementation part will show the data definition language (DDL) for all database objects. It also shows the implementation of main processes using the selected programming language. There are many types of programming language such as C, C++, Java and PHP. For this system the PHP programming languages has been chosen. Below are some examples of database implementation.

i) CREATE TABLE clause

Tables are created based on the Entity Relationship Diagram (ERD) on **Figure 4.1**. The example of create table syntax are shown in the **Figure 5.16** and the rest are included in **Appendix B**.

Table 5.1: Create Table Jenis_Kenderaan

```
CREATE TABLE JENIS_KENDERAAN
(
     JenisKenderaanID          INTEGER,
     JenisKenderaan            VARCHAR2(50)
     NOT NULL       UNIQUE,
     PRIMARY KEY(JenisKenderaanID)
);
```

Table 5.2: Create Table Branch

```
CREATE TABLE CAWANGAN
(
     CawanganID     INTEGER,
     AgensiID       NOT NULL,
     IbuPejabat     CHAR(1),
     KontingenID    NOT NULL,
     Alamat    VARCHAR2(50)    NOT NULL,
     Poskod    NUMBER(5)       NOT NULL,
     BandarID       NOT NULL,
     NoTelefon VARCHAR2(15)    NOT NULL,
     NoFaks    VARCHAR2(15)    NOT NULL,
     Emel           VARCHAR2(40),
     PRIMARY KEY(CawanganID),
     FOREIGN KEY(AgensiID) REFERENCES
     AGENSI(AgensiID)    ,
     FOREIGN KEY(KontingenID) REFERENCES
     KONTINGEN(KontingenID),
     FOREIGN KEY(BandarID) REFERENCES
     BANDAR(BandarID)
);
```

ii) TRIGGER clause syntax

A trigger is like a stored procedure that Oracle Database invokes automatically whenever a specified event occurs in a table. **Table 5.3** to **Table 5.6** showing the example of trigger. The rest of trigger can refer in **Appendix B**.

1) Before Insert Trigger

Table 5.3: Before Insert Trigger of Table Transport Brand

Table: Transport brand
Function: This before insert trigger is used for auto increment the transportation type unique id.

CREATE OR REPLACE TRIGGER JKENDERAAN_ID
BEFORE INSERT ON JENAMA_KENDERAAN
FOR EACH ROW
 	
BEGIN
  SELECT JNMKENDERAAN_SEQ.NEXTVAL
  INTO :new.JENAMAID
  FROM dual;
END;

Table 5.4: Before Insert Trigger of Table Agency

Table: Agency
Function: This before insert trigger is used for auto increment the agency unique id by adding 'A' in front of the id. For example 'A1001'.

create or replace trigger trig_agensiid
before insert on agensi
for each row
begin
if :new.AGENSIID is null then
select 'A' || to_char(AGENSIID_SEQ.NEXTVAL,'FM0000')
into :new.AGENSIID from dual;
end if;
END trig_agensiid;

2) After Insert Trigger

Table 5.5: After Insert Trigger of Table Agency and Branch

Table: Branch and agency
Function: This after insert trigger is used for update the total transport for each agency after register the transport in each branch for the agency. 'A1001'.
create or replace trigger trig_cawangan after insert on CAWANGAN for each row begin update AGENSI set JUMLAHCAWANGAN = JUMLAHCAWANGAN + 1 WHERE AGENSIID = :NEW.AGENSIID; END;

iii) Stored Procedure syntax

Stored procedure can manage and maintain data easily. The benefits of using stored procedure are they allow modular programming, faster execution and can reduce network traffic. Stored procedure also can be used as a security mechanism. In **Table 5.6** to **Table 5.10** shows the example of stored procedure. The rest of stored procedure can refer to **Appendix B**.

1) Stored Procedure for Insert

Table 5.6: Stored Procedure Insert for Table Transport Type

Table: Transport Type
Function: The stored procedure used to insert the transport type into its table when admin register the transport type.

create or replace PROCEDURE INSERTJNSKENDERAAN (
p_jenis IN JENIS_KENDERAAN.JENISKENDERAAN%TYPE)
IS
BEGIN
INSERT INTO JENIS_KENDERAAN(JENISKENDERAAN)
VALUES (p_jenis);
COMMIT;
END;

Table 5.7: Stored Procedure Insert for Table Transport Model

Table: Transport Model
Function: The stored procedure used to insert the transport model during the registration of transport brand.

create or replace PROCEDURE INSERTJENAMA (
p_jenama IN JENAMA_KENDERAAN.NAMAJENAMA%TYPE)
IS
BEGIN
INSERT INTO JENAMA_KENDERAAN(NAMAJENAMA)
VALUES (p_jenama);
COMMIT;
END;

2) Stored Procedure for Select

Table 5.8: Stored Procedure Select for Table Transport Type

Table: Transport Type
Function: This stored procedure is to select transport type where the type is like the input that inserted.

Create or replace PROCEDURE
searchJnsKenderaan(JnsKenderaan varchar2, v_cur IN OUT
sys_refcursor)
AS
BEGIN
OPEN v_cur FOR
SELECT * FROM JENIS_KENDERAAN WHERE JENISKENDERAAN
LIKE '%'||JnsKenderaan||'%';
END;

Table 5.9: Stored Procedure Select for Table Branch

Table: Branch
Function: This stored procedure is to list all the branches from their agency.

create or replace PROCEDURE SELECTCAWANGAN (p_agensiid
IN varchar2,c_dbuser OUT SYS_REFCURSOR)
IS
BEGIN
OPEN c_dbuser FOR
select * from CAWANGAN WHERE AGENSIID = p_agensiid;
COMMIT;
END;

3) Stored Procedure for Update

Table 5.10: Stored Procedure Update for Table Transport Type

Table: Transport Type
Function: This stored procedure is to update the transport type for each transport.

create or replace PROCEDURE UPDATEJENISKENDDERAAN(
p_jenis IN JENIS_KENDERAAN.JENISKENDERAAN%TYPE,
p_id IN JENIS_KENDERAAN.JENISKENDERAANID%TYPE
)
IS
BEGIN
UPDATE JENIS_KENDERAAN set JENISKENDERAAN = p_jenis
where JENISKENDERAANID = p_id;
COMMIT;
END;

5.1 Conclusion

A database can be thought of as a set of logically related files organized to facilitate access by one or more applications programs and to minimize data redundancy. Databases are structured to facilitate the storage, retrieval, modification, and deletion of data in conjunction with various data-processing operations. A database management system (DBMS) extracts information from the database in response to queries.

In the implementation phase is where you install the DBMS on the required hardware, optimize the database to run best on that hardware and software platform, and create the database and load the data. It also consist about the database implementation. This chapter also described the implementation of trigger and stored procedure in this system. This phase required many times to developed the system.

After implementation phase, we have to do testing phase. Testing phase is the process of executing the system with the intent of finding the software bugs. It will explain more in the Chapter VI.

CHAPTER VI

TESTING

6.1 Introduction

This chapter will discuss about testing period and activity of Auxiliary Police Association Management System. Testing is a process of executing a program or application with the intent of finding the software bugs. It can also be stated as the process of validating and verifying that a software program or application or product. The software testing is important ass it can avoid mistakes that are visible to users and also to avoid organization involved has a bad reputation.

Software testing is a very challenging task. But, testing is not limited to the implementation of the system with the purpose of finding defects. It is also need to determine test plan, test strategy, test design, test results and analysis during the testing phase. This can gain confidence that the system will work smoothly and free from any errors. The method and strategy of software testing will be discussed more in this chapter.

6.2 Test Plan

Test plan is a document that contains of detailed procedures that determine the scope, approach, resources and schedule of all testing activities. Test plan should be carefully planned by study about the functionality and characteristic of the system. In the test plan there are test organization, test environment and test schedule.

6.2.1 Test Organization

Test organization will define user responsibility on the system. This testing group will responsible in managing, executing and checking. The tester developer will lead the testing process and the main predominant in testing organization. The detailed about the person who involved in this testing process is shown in **Table 6.1**.

Table 6.1: Test Organization

Name	Roles	Responsibilities
Ummi Izzati Binti Mustapha	System Developer	Responsible for any errors identified and bug that appeared on this system. They will make sure the system can functional well and meet the requirement.
Puan Nor Mas Aina Binti Md Bohari	Supervisor	Monitor the work flow of the system. Analyze and review the functionality of the

		requirement.

6.2.2 Test Environment

In this section will explain the details about the location of testing to be carried out. In test environment also defined and setup of software and hardware for the testing teams to execute test cases. The **Table 6.2** shows the details about test environment.

Table 6.2: Environment Setup Specification

Environment Specification	Description
Operating System	Windows 7
Processor	Intel(R)
Random Access Memory (RAM)	8GB and greater
Database	Oracle
Server	Apache Web Server
Server Scripting Language	PHP

6.2.3 Test Schedule

Test schedule is a record of testing time table made for the system. All the testing made is based on the module stated for the system. The test schedule consist of testing

type, start date, end date and duration take to complete the system. In the **Table 6.3** described the testing process in the details.

Table 6.3: Test Schedule Detail

Activities	Description	Start Date	End Date	Duration
Unit Testing	Used to test functions and code module	7 August 2017	9 August 2017	3 days
System Testing	Evaluate system compliance with its specific requirements.	8 August 2017	9 August 2017	2 days
Acceptance Testing	Test completed system to end user.	10 August 2017	12 August 2017	3 days

6.3 Test Strategy

A test strategy is an outline that describes the testing approach of the software development cycle. It is created to inform project managers, testers, and developers

about some key issues of the testing process. For the APA Management System, Dynamic Testing will be used to evaluate this system. Dynamic testing is a process of validating software applications as an end user under different environments to build the right software. It is classifies into two categories that which are known as white box testing and black box testing.

Black box testing is a method of testing in which the internal structure or code is not known to the tester. The main purpose of this testing is to verify the functionality of the system. This black box testing requires to execute the complete test and it is performed by the Testers and there is no need of any programming knowledge. Black box testing is focused solely on the output generated in respond to selected input and execution condition.

For the testing part of this system, black box testing has been used because it will test the functionality of this system. Besides, the user of this system is among the person who not have knowledge about the programming.

6.3.1 Classes Of Test

In this section will explain about the classes of test. The tests are classified in two types. They are functional testing and non-functional testing. Functional testing is testing process used within software development in which software is tested to ensure that it confirms with all requirements. Besides, non-functional testing is focus more on the behaviors of the system and the way a system operates.

There are many types of Functional Testing out of which the most important are as described below.

i) Functionality Testing

It is being done to make sure that there are only valid data are stored in the database. It is performed at individual unit of software and is performed by developers. This testing is to make sure APA Management System performs as per requirements and is generally performed when the complete system is ready. It also used to verify whether this system has met the business requirements and is ready to use by the end user.

ii) Security Testing

Security testing is performed to ensure that only the authorize users are accessing the system.

6.4 Test Design

Test design is creating a set of input for given software that will provide a set of expected output. In this test design phase, there are two parts needs to be implemented which are test description and test data. Test description will describe more on the activities and it will document. Test data will use synthetic data to test the system and user acceptance test.

6.4.1 Test Description

Test description for Auxiliary Police Association Management System is explained based on the system module as shown in table below. The rest will show in **Appendix C**.

i) Login details

Login module is important for authorized user. Users are required to enter ID and password in order to log in into the system. **Table 6.4** will display the test details for login module.

Table 6.4: Test Case of Login Module

Test ID	T001		
Module Name	Login Module (System)		
Description	To ensure the system only can be access by the authorize users		
Test Designed by	System developer	**Test Executed by**	Supervisor and system developer
Pre-Condition	User had to register as the agency of Auxiliary Police Association		
Test Case ID	**Test Case**	**Procedures**	**Excepted Result**
T001_1	Login button is clicked with all fields are empty	Press login button	Alert box about field required fields is display
T001_2	Login button is clicked with user is empty	Insert password fields	Alert box about invalid username or password
T001_3	Login button is clicked with password is empty	Insert username fields	Alert box about invalid username or password
T001_4	Login button is clicked with	Insert username and incorrect password	Alert box about invalid username or

	password fields incorrect		password
T001_5	Login button is clicked with incorrect username fields	Insert incorrect username and password	Alert box about invalid username or password
T001_6	Login button is clicked with invalid username and password	Insert invalid username and password	Alert box about invalid username or password
T001_7	Login button is clicked with valid username and password	All input inserted are valid	Alert box about Successfully login!

ii) Transportation Model Module

Admin have to register the transport model to this system. **Table 6.5** shows the test case of transportation model module.

Table 6.5: Test Case of Transportation Model Module

Test ID	T002		
Module Name	Registration of Transportation Model Module (System)		
Description	To register the agency of Transportation Model		
Test Designed by	System developer	**Test Executed by**	Supervisor and system developer
Pre-Condition	-		

Test Case ID	Test Case	Procedures	Excepted Result
T002_1	Add transport model button clicked with all the data required is empty	Press add transport button	Alert box about field required fields is display
T002_2	Add transport model button is clicked with one of the required fields is empty	If either one of required field is empty	Alert box about field required fields is display
T002_3	Register button is clicked with all of the required fields is inserted	All input inserted	Alert box about Data Has Been Saved.

iii) Transport Details

The transport must be register to this system. All the information of transport must be register. **Table 6.6** shows the test case of transport details.

Table 6.6: Test Case of Transport Registration

Test ID	T003		
Module Name	Transport Registration Module(System)		
Description	To register the information of transport that APA have.		
Test Designed by	System developer	**Test Executed by**	Supervisor and system developer
Pre-	-		

Condition			
Test Case ID	**Test Case**	**Procedures**	**Excepted Result**
T003_1	Register button is clicked with all fields are empty	Press register button	Alert box about field required fields is display.
T003_2	Register button is clicked with one of required field(plate number field) is empty	Insert engine power field, picture file, PDRM approval date field, transport model field.	Alert box about field required fields is display.
T003_3	Register button is clicked with one of required field(engine power field) is empty	Insert plate number field, picture file, PDRM approval date field, transport model field.	Alert box about field required fields is display.
T003_4	Register button is clicked with one of no-required field(picture field) is empty	Insert plate number field, engine power field, PDRM approval date field, transport model field.	Alert box about Data Has Been Saved.
T003_5	Register button is clicked with one of required field(PDRM approval date field) is empty	Insert plate number field, engine power field, picture field, transport model field.	Alert box about field required fields is display.
T003_6	Register button is clicked with all required field are inserted.	All input inserted are valid	Alert box about Data Has Been Saved.

6.4.2 Test Data

Test data is a part of actual data will be used to ensure the system for correctness and system effectiveness. The following are some of the test data for every each of the test case. For other test data, can refer to **Appendix C**.

Table 6.7: System Login Test Data

Test ID	T001
Module Name	Login Module (System)
Test Case ID	T001_1
Test Case	Login button is clicked with all fields are empty
Input Field	**Test Data**
User ID	None
Password	None
Test Case ID	T001_2
Test Case	Login button is clicked with user is empty
Input Field	**Test Data**
User ID	None
Password	12345
Test Case ID	T001_3
Test Case	Login button is clicked with password is empty
Input Field	**Test Data**
User ID	A001
Password	None
Test Case ID	T001_4
Test Case	Login button is clicked with password fields incorrect
Input Field	**Test Data**
User ID	A001
Password	Ud874

Test Case ID	T001_5
Test Case	Login button is clicked with incorrect username fields
Input Field	**Test Data**
User ID	S998
Password	12345
Test Case ID	T001_6
Test Case	Login button is clicked with invalid username and password
Input Field	**Test Data**
User ID	S998
Password	Ju89d
Test Case ID	T001_7
Test Case	Login button is clicked with valid username and password
Input Field	**Test Data**
User ID	A001
Password	12345

Table 6.8: Registration of Transportation Model Module Test Data

Test ID	T002
Module Name	Registration of Transportation Model Module (System)
Test Case ID	T002_1
Test Case	Add transport model button clicked with all the data required is empty
Input Field	**Test Data**
Model Name	None

Transport Type	None
Brand Name	None
Test Case ID	T002_2
Test Case	Add transport model button is clicked with one of the required fields is empty
Input Field	**Test Data**
Model Name	None
Transport Type	Kereta
Brand Name	Proton
Test Case ID	T002_3
Test Case	Add transport model button is clicked with all of the required fields is inserted.
Input Field	**Test Data**
Model Name	Suprima S
Transport Type	Kereta
Brand Name	Proton

Table 6.9: Transport Registration Module (System) Test Data

Test ID	T003
Module Name	Transport Registration Module(System)
Test Case ID	T003_1
Test Case	Register button is clicked with all fields are empty
Input Field	**Test Data**
Plate Number	None
Engine Power	None
Transport Image	None
PDRM Approval Date	None
Transport Model	None

Test Case ID	T003_2
Test Case	Register button is clicked with one of required field(plate number field) is empty
Input Field	**Test Data**
Plate Number	None
Engine Power	1600
Transport Image	Swift.jpg
PDRM Approval Date	22-MAY-2017
Transport Model	Swift
Test Case ID	T003_3
Test Case	Register button is clicked with one of required field(engine power field) is empty
Input Field	**Test Data**
Plate Number	WUP 1965
Engine Power	None
Transport Image	Swift.jpg
PDRM Approval Date	22-MAY-2017
Transport Model	Swift
Test Case ID	T003_4
Test Case	Register button is clicked with one of no-required field(picture field) is empty
Input Field	**Test Data**
Plate Number	WUP 1965
Engine Power	1600
Transport Image	None
PDRM Approval Date	22-MAY-2017
Transport Model	Swift
Test Case ID	T003_5
Test Case	Register button is clicked with one of required

	field(PDRM approval date field) is empty
Input Field	**Test Data**
Plate Number	WUP 1965
Engine Power	1600
Transport Image	Swift.jpg
PDRM Approval Date	None
Transport Model	Swift
Test Case ID	T003_6
Test Case	Register button is clicked with all required field are inserted.
Input Field	**Test Data**
Plate Number	WUP 1965
Engine Power	1600
Transport Image	Swift.jpg
PDRM Approval Date	22-MAY-2017
Transport Model	Swift

6.5 Test Result and Analysis

This section will described about the validation of the data over APA System. This test document is to see whether the result in the test design is the same with the test result and analysis. **Table 6.9** until **Table 6.11** shows the details of the test document. For other test result and analysis, can refer to the **Appendix C**.

Table 6.9: Test Result and Analysis for Login Module

Test ID	T001

Module Name	Login Module (System)	
Test Case ID	**Actual Result**	**Status (Pass/Fail)**
T001_1	Alert box about field required fields is display	Pass
T001_2	Alert box about invalid username or password	Pass
T001_3	Alert box about invalid username or password	Pass
T001_4	Alert box about invalid username or password	Pass
T001_5	Alert box about invalid username or password	Pass
T001_6	Alert box about invalid username or password	Pass
T001_7	Alert box about Successfully login!	Pass

Table 6.10: Test Result and Analysis for Transportation Model Module

Test ID	T002	
Module Name	Registration of Transportation Model Module (System)	
Test Case ID	**Actual Result**	**Status (Pass/Fail)**
T002_1	Alert box about field required fields is display	Pass

T002_2	Alert box about field required fields is display	Pass
T002_3	Alert box about Data Has Been Saved.	Pass

Table 6.11: Test Result and Analysis for Transportation Registration

Test ID	T003	
Module Name	Transport Registration Module(System)	
Test Case ID	**Actual Result**	**Status (Pass/Fail)**
T003_1	Alert box about field required fields is display.	Pass
T003_2	Alert box about field required fields is display.	Pass
T003_3	Alert box about field required fields is display.	Pass
T003_4	Alert box about Data Has Been Saved.	Failed
T003_5	Alert box about Data Has Been Saved.	Pass
T003_6	Alert box about Data Has Been Saved.	Pass

For the test analysis, the data for inserted image in T003_4 is failed. It is because the oracle can't read the binary image. Most of the test cases are succeed to insert data. For the improvement of this Auxiliary Police Association Management System, it has to validate the valid data inserted such as when inserted the phone number, user can't inserted symbol or letter.

6.1 Conclusion

In conclusion, testing phase is very important and crucial to check on any error and malfunction program that may occur in Auxiliary Police Association Management System. This chapter explains the method use to verify and validate the system to make sure the quality of the project have achieved through requirements. Testing must be planned thoroughly due to the cost of fixing one major defects can be very costly. Test plan consists of several test that examine varies aspects of the system.

In the next chapter will discuss about the project conclusion. It will conclude all the document that has been done. In conclusion chapter, we will discuss about the objective of this system.

CHAPTER VII

CONCLUSION

7.1 Introduction

Malaysia Auxiliary Police Association (APA) Management System (Transportation Module) is a web-based system that provides efficient management to manage transportation in this association. This system replaces the existing manual system that uses the form to register the transportation and distribute the transportation to the agencies. This system provides a systematic management for the transportation.

.

However, this system has many flaws but there are some advantages and it is to upgrade in a long-term duration. During the development of this system, there are a lot of knowledge that I have gained in terms of drafting the final report, designing the system and flow data. This system has the ability to be developed more sophisticated with better expertise than my lack of knowledge and expertise in this area.

In conclusion, I have developed a system that is 70% achieving the main objectives despite of many shortcoming and failure in work. This chapter will explain about the observation of weakness and strength for this system, propositions for improvement in the future, project contribution for the community.

7.2 Observation on Weaknesses and Strengths

During the development Auxiliary Police Association Management System, there are a few of weaknesses and strengths have been identified. This observation is being made up for find the flaws of system and make some improvement to the system in the future.

1) Strengths

Strengths of Auxiliary Police Association Management System are:

- Can manage the transportation via online (no writing in paper).
- Data is stored in an organized way due to ERD design.
- Can view transport that they have easily.
- Can print the report of transportation for each agency.
- Reduce the use of time when register the transport.
- User friendly. Have easy features with minimal GUI for the interface and navigation button.

2) Weaknesses

The weaknesses of Auxiliary Police Association Management System are:

- No backup and recovery. If the system down, all the data information will be lost.
- There is no notification system when users update the status of each transport.
- The system can't produce report about the transportation.

7.3 Proposition for Improvement

For the improvement of the system, it should be better if backup and recovery are implemented in the system. If there is data corruption or problems with the system, all the data can be retrieved without have to think about loss of the important data.

Other than that, the system should add notification system for users. So, if any update from the agency, the Auxiliary Police Association (Malaysia) will be able to know it. The system also should be able to produce report in PDF format file about the transportation for each agency.

7.4 Project Contribution

Auxiliary Police Association Management System (Transport Module) is developed for Malaysia Auxiliary Police Association management. This system is to manage the transport for each agency in Auxiliary Police Association. The agency can

manage their transport that given by PDRM. This system replaced the manual system that they used before. The manual of the system can refer to **Appendix D**.

7.5 Conclusion

The conclusion that can be concluded after completing the system, it been developed to ease for the users to manage daily management of a company by implementing various functions in the system. This system has been achieved some of the objective. It has been successful to change the manual system into web-based system. It also provided a systematic management system for the transportation. The system is developed to save time during registration of transport by using computerized system.

Although the system is not met all the objectives, but it is still can function smoothly. However, if there is some improvements add to the system such as a user friendly system or backup and recovery is implemented the system will be more reliable and well developed. The system should be able to produce report for transport in each agency. However, almost all the functions and requirements were successfully implemented to this Auxiliary Police Association Management System (Transport Module).

REFERENCES

[1] Carlos Coronel and Steven Morris “Database System: Design, Implementation, and Management” Eleventh Edition[Book]

[2] Universiti Teknikal Malaysia Melaka . “Panduan Penulisan Laporan PSM BITD

2015.” [Online] Available from: http:/www.ulearn.utem.edu.my/

BIBLIOGRAPHY

[1] Universiti Teknikal Malaysia Melaka . “Panduan Penulisan Laporan PSM BITD

2015.” [Online] Available from: http:/www.ulearn.utem.edu.my/

[2] Atiqah S(2005). “Visitor Managemnet System For Am Biofuels SDN BHD”

Perpustakaan Universiti Teknikal Malaysia Melaka.

APPENDIX A

Attachment 1.1 : Data Dictionary Agency Table

Attribute	Content	Type	Domain	Required	PK/FK	Referred Table
AgensiID	Agency unique ID	INT (10)	A1001	Y	PK	
TarikhPermohonanPB	Date of application to be APA agency	DATE		Y		
NamaAgensi	Agency name	VARCHAR (50)		Y		
Alamat	Agency address	VARCHAR(50)		Y		
Poskod	Postcode of agency	INT(10)		Y		

BandarID	City Unique ID	INT(10)		Y	FK	BANDAR
KategoriAgensiID	Agency Category Unique ID	INT(10)		Y	FK	KATEGORI_AGENSI
SektorPerniagaan	Agency Business Sector	VARCHAR(50)		Y		
NoSyarikat	Registered company number	VARCHAR(50)		Y		
Logo	Agency logo	BLOB				
Nama	Agency representative name	VARCHAR(50)		Y		
Jawatan	Agency representative position	VARCHAR(50)		Y		
NoKadPengenalan	Agency representative IC number	VARCHAR(14)		Y		
Emel	Agency representative email	VARCHAR(40)		Y		
KataLaluan	Agency representative username for login to the system	VARCHAR(12)		Y		
PerananID	Role of agency unique ID	INT(10)		Y		
NoTelefonBimbit	Agency representative hand phone number	VARCHAR(15)		Y		

NoTelefonPejabat	Agency representative office phone number	VARCHAR(15)		Y		
NoFaks	Faks number for agency	VARCHAR(15)		Y		
TarikhKelulusanPDRM	Approval date from PDRM	DATE				
NoRujukanKelulusanPDRM	PDRM Approval references number	VARCHAR(30)				
TarikhKelulusanAPA	Approval date from APA	DATE				
NoRujukanKelulusanAPA	APA Approval references number	VARCHAR(30)				

Attachment 1.2: Data Dictionary Reason Table

Attribute	Content	Type	Domain	Required	PK/FK	Referred Table
AlasanID	Reason unique ID	INT (10)		Y	PK	
NamaAlasan	Reason name	VARCHAR(100)		Y		
StatusID	Status unique ID	INT (10)		Y	FK	STATUS

Attachment 1.3: Data Dictionary for Status Table

Attribute	Content	Type	Domain	Required	PK/FK	Referred Table
StatusID	Status unique ID	INT (10)		Y	PK	
JenisStatus	Status type	VARCHAR(50)		Y		
Penerangan	Explanation about status	VARCHAR(150)				

Attachment 1.4: Data Dictionary Agency Category

Attribute	Content	Type	Domain	Required	PK/FK	Referred Table
KategoriAgensiID	Agency category unique ID	INT (10)		Y	PK	
NamaKategori	Category Name	VARCHAR(50)		Y		
Penerangan	Category description	VARCHAR(150)				

Attachment 1.5: Graphical User Interface Transport Model Page

Title	Transport Model	Designer	Ummi Izzati Binti Mustapha
Project	APA (Transport Module) System	Page	4
		Description	Model Kenderaan page

Description

Label 1 : "MODEL KENDERAAN"

Text field 1: Model Kenderaan field

Button 1 = "Cari" button to Search the "Model kenderaan".

Label 2: "Model Kenderaan Id"

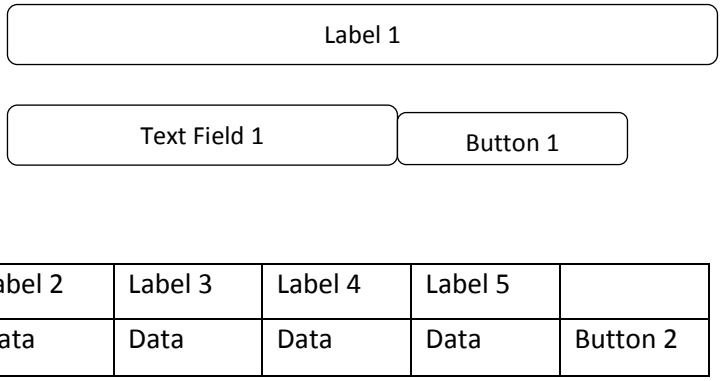
Label 3: "Model Kenderaan"

Label 4: "Jenama"

Label 5: "Jenis Kenderaan"

Button 2= "Kemaskini" script to update the "Model Kenderaan".

Data=the data of "Model Kenderaan"



Attachment 1.6: Graphical User Interface for Agency Page

Title	Agency Page	Designer	Ummi Izzati Binti Mustapha
Project	APA (Transport Module) System	Page	5
		Description	Agency page

Description

Label 1 : “SISTEM PENGURUSAN POLIS BANTUAN”.

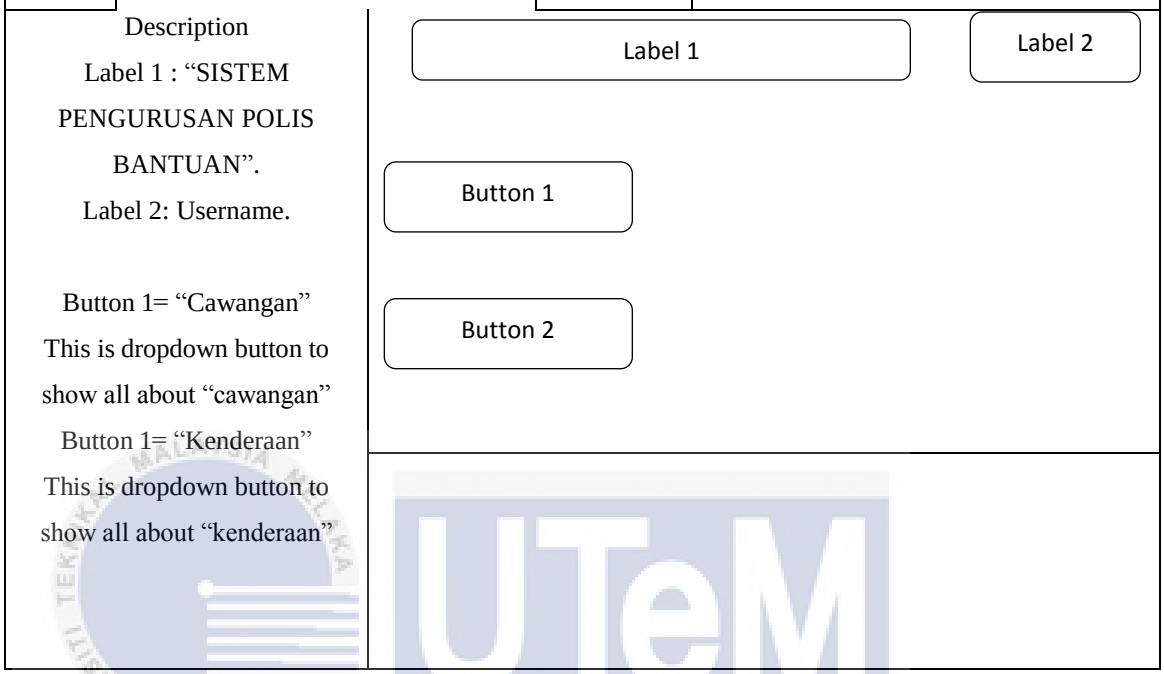
Label 2: Username.

Button 1= “Cawangan”

This is dropdown button to show all about “cawangan”

Button 1= “Kenderaan”

This is dropdown button to show all about “kenderaan”



APPENDIX B

Attachment 2.1: Create Table Transport Location

```
CREATE TABLE LOKASI_KENDERAAN
(
     NoPendaftaran,
     CawanganID,
     TarikhMulaGuna  DATE     NOT NULL,
     TarikhTamatGuna DATE,
     StatusID      NOT NULL,
     AlasanID,
     AlasanLain     VARCHAR2(150),
     PRIMARY KEY(NoPendaftaran, CawanganID),
     FOREIGN KEY(NoPendaftaran) REFERENCES
     KENDERAAN(NoPendaftaran),
     FOREIGN KEY(CawanganID)  REFERENCES
     CAWANGAN(CawanganID),
     FOREIGN KEY(StatusID)    REFERENCES
     STATUS(StatusID),
     FOREIGN KEY(AlasanID)    REFERENCES
     ALASAN(AlasanID)
);
```

Attachment 2.2: Create Table Transport Model

```
CREATE TABLE MODEL_KENDERAAN
(
     ModelID         INTEGER,
     NamaModel       VARCHAR2(50)    NOT NULL
     UNIQUE,
     JenamaID             NOT NULL,
     JenisKenderaanID     NOT NULL,
     PRIMARY KEY(ModelID),
     FOREIGN KEY(JenamaID) REFERENCES
     JENAMA_KENDERAAN(JenamaID),
     FOREIGN KEY(JenisKenderaanID) REFERENCES

     JENIS_KENDERAAN(JenisKenderaanID)
);
```

Attachment 2.3: Create Table Transport brand

```
CREATE TABLE JENAMA_KENDERAAN
(
     JenamaID        INTEGER,
     NamaJenama      VARCHAR2(100)  NOT NULL
     UNIQUE,
     PRIMARY KEY(JenamaID)
);
```


Attachment 2.4: Create Table Agency Category

```
CREATE TABLE KATEGORI_AGENSI
(
     KategoriAgensiID INTEGER,
     NamaKategori     VARCHAR2(50)  NOT NULL
     UNIQUE,
     Penerangan       VARCHAR2(150),
     PRIMARY KEY(KategoriAgensiID)
);
```

Attachment 2.5: Create Table Agency

```
CREATE TABLE AGENSI
(
     AgensiID       INTEGER,
     TarikhPermohonanPB  DATE NOT NULL,
     NamaAgensi     VARCHAR2(50)   NOT NULL,
     Alamat         VARCHAR2(50)   NOT NULL,
     Poskod         NUMBER(5)      NOT NULL,
     BandarID                      NOT NULL,
     KategoriAgensiID              NOT NULL,
     SektorPerniagaan VARCHAR2(50) NOT NULL,
     NoSyarikat     VARCHAR2(30),
     Logo           BLOB,
     Nama           VARCHAR2(50)   NOT NULL,
     Jawatan        VARCHAR2(50)   NOT NULL,
     NoKadPengenalan VARCHAR2(14)  NOT NULL,
     Emel           VARCHAR2(40)   NOT NULL,
     KataLaluan     VARCHAR2(12),
     PerananID,
     NoTelefonBimbit VARCHAR2(15)  NOT NULL,
     NoTelefonPejabat VARCHAR2(15) NOT NULL,
     NoFaks          VARCHAR2(15),
     TarikhKelulusanPDRM     DATE,
     NoRujukanKelulusanPDRM   VARCHAR2(30),
     TarikhKelulusanAPA      DATE,
     NoRujukanKelulusanAPA VARCHAR2(30),
     PRIMARY KEY(AgensiID),
     FOREIGN KEY(BandarID) REFERENCES
     BANDAR(BandarID),
     FOREIGN KEY(KategoriAgensiID) REFERENCES
     KATEGORI_AGENSI(KategoriAgensiID),
     FOREIGN KEY(PerananID)        REFERENCES
     PERANAN(PerananID)
);
```

Attachment 2.6: Create Table City

```
CREATE TABLE BANDAR
(
     BandarID       INTEGER,
     NamaBandar     VARCHAR2(60)    NOT NULL,
     NegeriID       NOT NULL,
     PRIMARY KEY(BandarID),
     FOREIGN KEY(NegeriID) REFERENCES
     NEGERI(NegeriID)
);
```

Attachment 2.7: Create Table State

```
CREATE TABLE NEGERI
(
     NegeriID  INTEGER,
     NamaNegeri VARCHAR2(40)  NOT NULL UNIQUE,
     PRIMARY KEY(NegeriID)
);
```


Attachment 2.8: Trigger Contingency ID

Table: Contingency

Function: This before insert trigger is used for auto increment the Contingency unique id.

```
create or replace TRIGGER KONTINGENID
BEFORE INSERT ON KONTINGEN
FOR EACH ROW

BEGIN
  SELECT KONTINGENID_SEQ.NEXTVAL
  INTO   :new.KONTINGENID
  FROM   dual;
END;
```

Attachment 2.9: Trigger Branch ID

Table: Branch
Function: This before insert trigger is used for auto increment the Branch unique id.

create or replace trigger trig_cawanganid
before insert on cawangan
for each row
begin
if :new.CAWANGANID is null then
select '' || to_char(CAWANGAN_SEQ.NEXTVAL,'FM0000')
into :new.CAWANGANID from dual;
end if;
END trig_cawanganid;

APPENDIX C

Attachment 3.1: Test Case for Transportation Type Module

Test ID	T004		
Module Name	Registration of Transportation Type Module (System)		
Description	To register the Transportation Type		
Test Designed by	System developer	**Test Executed by**	Supervisor and system developer
Pre-Condition	-		
Test Case ID	**Test Case**	**Procedures**	**Excepted Result**
T004_1	Add transport button clicked with all the data required is empty	Press add transport button	Alert box about field required fields is display
T004_2	Add transport model button is clicked with one of the required fields is empty.	If either one of required field is empty	Alert box about field required fields is display
T004_3	Add state button is clicked with all of the required fields is inserted	All input inserted	Alert box about Data Has Been Saved.

Attachment 3.2: Test Case for Register State Module

Test ID	T005		
Module Name	Registration of State Module (System)		
Description	To register the state		
Test Designed by	System developer	**Test Executed by**	Supervisor and system developer
Pre-Condition	-		
Test Case ID	**Test Case**	**Procedures**	**Excepted Result**
T005_1	Add state button clicked with all the data required is empty.	Press add transport button	Alert box about field required fields is display
T005_2	Add state model button is clicked with one of the required fields is empty.	If either one of required field is empty	Alert box about field required fields is display
T005_3	Add state button is clicked with all of the required fields is inserted.	All input inserted	Alert box about Data Has Been Saved.

Attachment 3.3: Test Data for Registration of Transportation Type Module (System)

Test ID	T004
Module Name	Registration of Transportation Type Module (System)
Test Case ID	T004_1
Test Case	Add transport button clicked with all the data required is empty
Input Field	**Test Data**

Transport Type	None
Test Case ID	T004_2
Test Case	Add transport model button is clicked with one of the required fields is empty.
Input Field	**Test Data**
Transport Type	None
Test Case ID	T004_3
Test Case	Add state button is clicked with all of the required fields is inserted.
Input Field	**Test Data**
Transport Type	Motosikal

Attachment 3.4: Test Data for Registration of State Module (System)

Test ID	T005
Module Name	Registration of State Module (System)
Test Case ID	T005_1
Test Case	Add state button clicked with all the data required is empty
Input Field	**Test Data**
State name	None
Test Case ID	T005_2
Test Case	Add state model button is clicked with one of the required fields is empty.
Input Field	**Test Data**
State Name	None
Test Case ID	T005_3
Test Case	Add state button is clicked with all of the required fields is inserted.
Input Field	**Test Data**
State name	Melaka

Attachment 3.5: Test Result for Registration of Transportation Type Module (System)

Test ID	T004	
Module Name	Registration of Transportation Type Module (System)	
Test Case ID	**Actual Result**	**Status (Pass/Fail)**
T004_1	Alert box about field required fields is display	Pass
T004_2	Alert box about field required fields is display	Pass
T004_3	Alert box about Data Has Been Saved.	Pass

Attachment 3.6: Test Result for Registration of State Module (System)

Test ID	T005	
Module Name	Registration of State Module (System)	
Test Case ID	**Actual Result**	**Status (Pass/Fail)**
T005_1	Alert box about field required fields is display	Pass
T005_2	Alert box about field required fields is display	Pass
T005_3	Alert box about Data Has Been Saved.	Pass

APPENDIX D

UTeM
اونيۏرسيتي تيکنيکل مليسيا ملاک
UNIVERSITI TEKNIKAL MALAYSIA MELAKA