E-REPORT MANAGEMENT SYSTEM FOR DAMAGE OF FACILITY FOR UTeM RESIDENTIAL COLLEGE

MOHAMAD AZIM BIN SUHAIMI

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

BORANG PENGESAHAN STATUS TESIS

JUDUL: E-Report Management System for Damage of Facility for UTeM Residential College

SESI PENGAJIAN: 2015/2016

Saya MOHAMAD AZIM BIN SUHAIMI

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

_____ (Mr. Mohd Fadzil bin Zulkifli)

Alamat tetap: Blok A A-2-3 Flat Uda,
Bandar Baru Ampangan,
70400 Seremban, Negeri Sembilan.

Tarikh: Tarikh:

CATATAN: * Tesis dimaksudkan sebagai Laporan Projek Sarjana Muda (PSM)
**Jika tesis ini SULIT atau TERHAD, sila lampirkan surat daripada pihak berkuasa.

E-REPORT MANAGEMENT SYSTEM FOR DAMAGE OF FACILITY FOR UTeM RESIDENTIAL COLLEGE

MOHAMAD AZIM BIN SUHAIMI

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

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY
UNIVERSITI TEKNIKAL MALAYSIA MELAKA
2016

DECLARATION

I hereby declare that this project report entitled

E-REPORT MANAGEMENT SYSTEM FOR DAMAGE OF FACILITY FOR UTeM RESIDENTIAL COLLEGE

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

STUDENT : _____ Date: 25/8/2016

(MOHAMAD AZIM BIN SUHAIMI)

I hereby 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: 25/8/2016

(MR. MOHD FADZIL BIN ZULKIFLI)

DEDICATION

To my parents, Mr. Suhaimi bin Jaafar and Mrs. Saleha binti Ahmad.

To my supervisor, Mr. Mohd Fadzil bin Zulkifli.

To my lecturers, friends and family.

.

ACKNOWLEDGEMENT

Firstly, I would like to express my grateful to Allah S.W.T for good health and time to complete this thesis of E-Report Management System for Damage of Facility for UTeM Residential College. It would be impossible for me without the will of Him. Salutations and peace to His beloved messenger; Prophet Muhammad S.A.W.

Besides, I would like to express my sincere gratitude to my supervisor Mr. Mohd Fadzil bin Zulkifli for the continuous support for my Bachelor's Final Year Project, for his patience, motivation and immense knowledge.

I also thank my fellow friends for the stimulating discussion from the beginning until this system completed. Last but not least, I would like to thank my family for supporting me throughout my final year project and writing this thesis. Without them, I will not stand where I am now.

I also place on record, my sense of gratitude to one and all, who directly or indirectly, have lent their hand in this E-Report Management System for Damage of Facility for UTeM Residential College project.

THANK YOU

ABSTRACT

E-Report Management System is a reporting system that report on damage of facility for UTeM residential college that is Lestari. This system needs to be develop because it can facilitate student to do reports on damage of facility in their room. This system involved on manipulation status and calculation of damaged charges, it can be update automatically if student have any damage charges. The objective of this system to reduce the paper form, to simplifying process of reporting form submission, and to automate the student charges status for report of damage of facility. For this project, the project methodology used was agile. Waterfall method cannot be used because waterfall methods not to effective to use for developing the project like this. At the end of the project, there are some expected findings that may be discovered. This system will help the admin of this system that is Lestari Head of Department to checking student status more ease, and student not necessary to do reporting in the office anymore.

ABSTRAK

E-Report Management System ialah satu sistem laporan laporan itu di kerosakan kemudahan untuk kolej kediaman UTeM yang Lestari. Keperluan sistem ini membangunkan kerana ia boleh memudahkan pelajar lakukan laporan di kerosakan kemudahan dalam bilik mereka. Sistem ini melibatkan di status manipulasi dan pengiraan caj-caj rosak, ia boleh mengemaskinikan secara automatik jika pelajar mempunyai mana-mana caj-caj kerosakan. Objektif sistem ini mengurangkan bentuk kertas, untuk memudahkan proses penyerahan borang laporan, dan mengautomatikkan pelajar mendakwa status untuk laporan kerosakan kemudahan. Bagi projek ini, kaedah projek digunakan tangkas. Kaedah air terjun tidak boleh diguna kerana kaedah-kaedah air terjun untuk tidak berkesan untuk menggunakan kerana membangunkan projek itu begini. Pada akhir projek, terdapat beberapa penemuan dijangka yang boleh dijumpai. Sistem ini akan membantu pentadbiran sistem ini yang Lestari Head of Department pada pemeriksaan status pelajar lebih meringankan, dan pelajar tidaklah perlu lakukan melaporkan di dalam pejabat lagi.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

PSM	-	Project Sarjana Muda
CPU	-	Central Processing Unit
DBMS	-	Database Management System
DFD	-	Data Flow Diagram
ERD	-	Entity Relationship Diagram
GUI	-	Graphic User Interface
RAM	-	Random Access Memory
SDLC	-	System Development Life Cycle
UTeM	-	Universiti Teknikal Malaysia Melaka

LIST OF ATTACHMENT

CHAPTER 1

INTRODUCTION

1.1 Project Background

The system that will be develop for Project Sarjana Muda (PSM) is called E-Report Management System. This system is able to manage and monitoring the Lestari residential college facility more effective. Current damage reporting method for facility do not consider the distinctiveness of buildings in analysis. The current system have many disadvantage such as; abundant of paper forms and time limited of reporting. Therefore, the current system mostly have to make the student come to the office to make the report of the damage according to office hours. Staff also have to handle many forms filled by the student and the possibility of lost the form is high.

E-Report Management System have several advantage such as providing online report form to fill-in and submit more easily and prolifically. This system will help the student make a report regarding the damage of their room facility. Means that student does not have to go office to make a report if there have any damage in their room except to make a payment, get or return room key and any other equipment's. The student report will processed automatically by the system and the status of the report can be checked by

online. The report even can be do in weekend and will help the staff to arrange and monitoring the room for keep track the report to settle the problem. This system also provide an online application to leave in or not in the residential college. This system also able to monitoring number of job of staff and arrange which staff have to manage the damage of facility report that been done by student.

1.2 Problem Statement

E-Report Management System is developed based on some problem arising in Lestari residential college. For example:

i. Abundant of paper forms

The current process is that student have to make fill in the paper form for making the report. The problem of paper form is when the paper lost before being recorded and overlooked the form by the staff. Then staff has to get through or find the form submitted by student to solve the reported problem.

ii. Time of report

Student have to go to the Lestari residential college office to make a report according to an office hours of the staff. Student also cannot do a report if their have any damage in weekend.

1.3 Objective

The objective of developing E-Report Management System are identified based on the review of the problem statement. The purposes initiated with following objective:

i. To reduce paper forms usage

The system can reduce the paper form by doing online application forms for report of damage of facility. This system also can save residential college admin for

searching and get the information of the report. The data will be inserted in local database of Lestari.

ii. To ease up the application form submission process

E-Report management system is web-based system. Student can fill in the report form by online without going to the residential college office and this is on 24 hours a day. The student can make a report without have to wait for office hour.

iii. To automate the student charges status process

This system allow admin to check student damage charges status before make any action to student.

1.4 Scope

The target users of this system are admin of the residential college that is the Lestari Head of Department, and student that rental in Lestari residential college.

i. Admin of Lestari Residential College

With this system, Admin can view the list of student that rental in the residential college and view the activity and report of the damage of facility. Admin also, able to insert, update, delete and view a facility of each house and room within the residential college. Before admin can register a student that want to rental the room, they can easily search available room. They also can view all the damage report, update damage report status and calculate any charge of the damage.

ii. Student that rental in Lestari Residential College

With this system, student able to make a report of damage facility online without have to go report in the residential college office. They also can view their facility and charges of damage of facility.

1.5 Project Significance

The importance of E-Report Management System to be developed are managing data efficiently and allow users to perform tasks with ease. Currently, to make a damage report student have to come to Lestari residential college within office hour and fill in the form. The staff are storing the data in a paper and sometimes have a hard time to find it.

E-Report Management System can stores, organizes and manages a large amount of information data within a single software application. By using this system, it can increase efficiency of their daily operation. The system ensure all the data are stored safely and ready to be displayed accordingly. Furthermore, the system eases and fastens the assessment process and be able to react quickly when there are report from the damage of facility.

1.6 Expected Output

The anticipated outcomes from E-Report Management System are listed as below:

i. Article theories/New findings/Knowledge

At the end of the project, there are some expected findings that may be discovered. This system will help the admin of this system that is Lestari Head of Department to checking student status more ease, and student not necessary to do reporting in the office anymore.

ii. Project Publications

At the end of the project, hopefully this system able to be publication because it is very helpful to admin and student of Lestari residential college. In the future, this system can be improve by adding another function for the reporting on damage of facility.

1.7 Conclusion

The result of this project will allow student to ease up the damage report form submission process to student rather than fill in the form in residential college office. Henceforth, it can be facilitated staff to manage the process more efficiently and reduce paper form usage. This system also can ease; can search the required report. As a conclusion, this project will provides overview of the report operation of the Lestari residential college.

Introduction to the project is a good starting point for need to coordinate the work effectively. Hence, the next chapter will be Chapter II – Project Methodology and Planning, which will discuss how the project is plan and what methodolgy will be used to attain the aims as stated in this chapter.

CHAPTER II

PROJECT METHODOLOGY AND PLANNING

2.1 Introduction

This chapter is about the project methodology and of the project. The system of the project is E-Report Management System. This system has two scopes, staff and student or resident at Lestari Residential College. When there is a project was being developed, the method used was very important for the success of the project. Method was used to estimate the time of the system to be delivered on the stage. Therefore for this E-Report Management System project, method that will be used is agile method because this agile method is more flexible compared to waterfall method. Agile method is an alternative to traditional project management.

2.2 Project Methodology

The current system that are used now does not efficient and effective during the operation. So that E-Report Management System is the system that will be used to replaces the current system. There are two project methodology; software development methodology and database development methodology.

2.2.1 Software Development Methodology

The current system is not so efficient and effective during the operation. Hence, e-Report Management System proposed the system that can replace the current system. This section defines the methodology or the procedures that will be followed in order to meet the main goal and also to accomplish all the objectives. The methodology to develop the system is agile method. Even though this method is one of the traditional approaches for project management, this method easy to respond any unpredictability, through incremental, iterative work cadences, known as sprints. By focusing on the repetition of abbreviated work cycles as well as the functional product yield, agile methodology is described as "iterative" and "incremental.

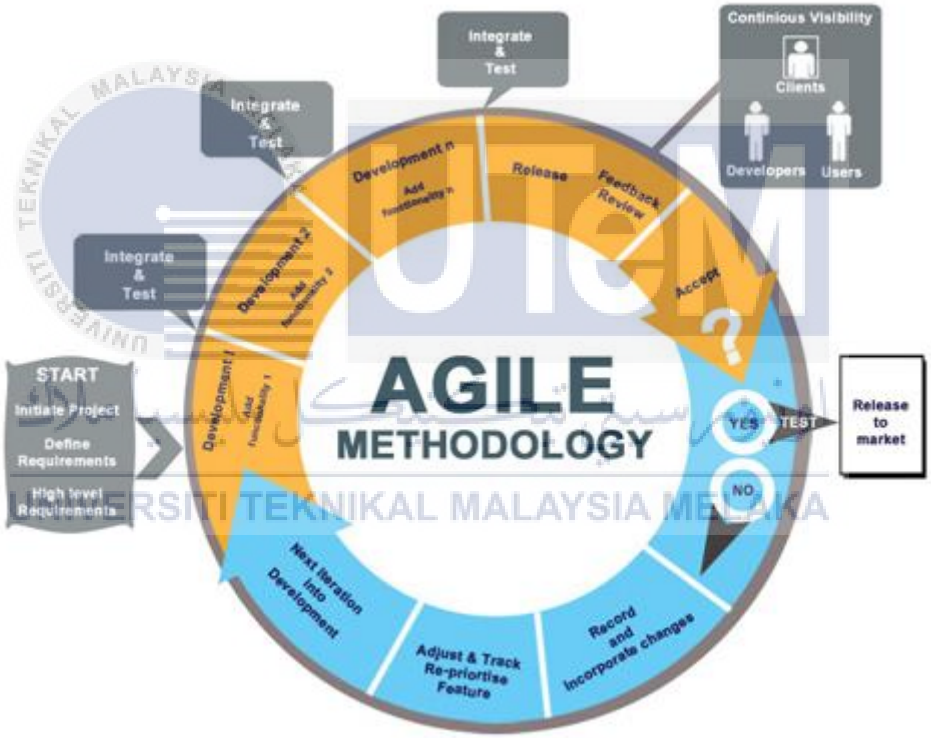


Figure 2.1: SDLC Methodology Diagram of Agile Development

There are going to be continuously planning, testing and integrating will be conducted within the process of develop this project for detect the defects of system. This is to ensure that our system meets all the requirements and to avoid wasting of time and cost in a long term.

2.2.2 Database Development Methodology

As for e-Report Management System project database methodology on Database Development Life Cycle (DBLC), it is going to be developed as a top down approach and implemented separately with SDLC.

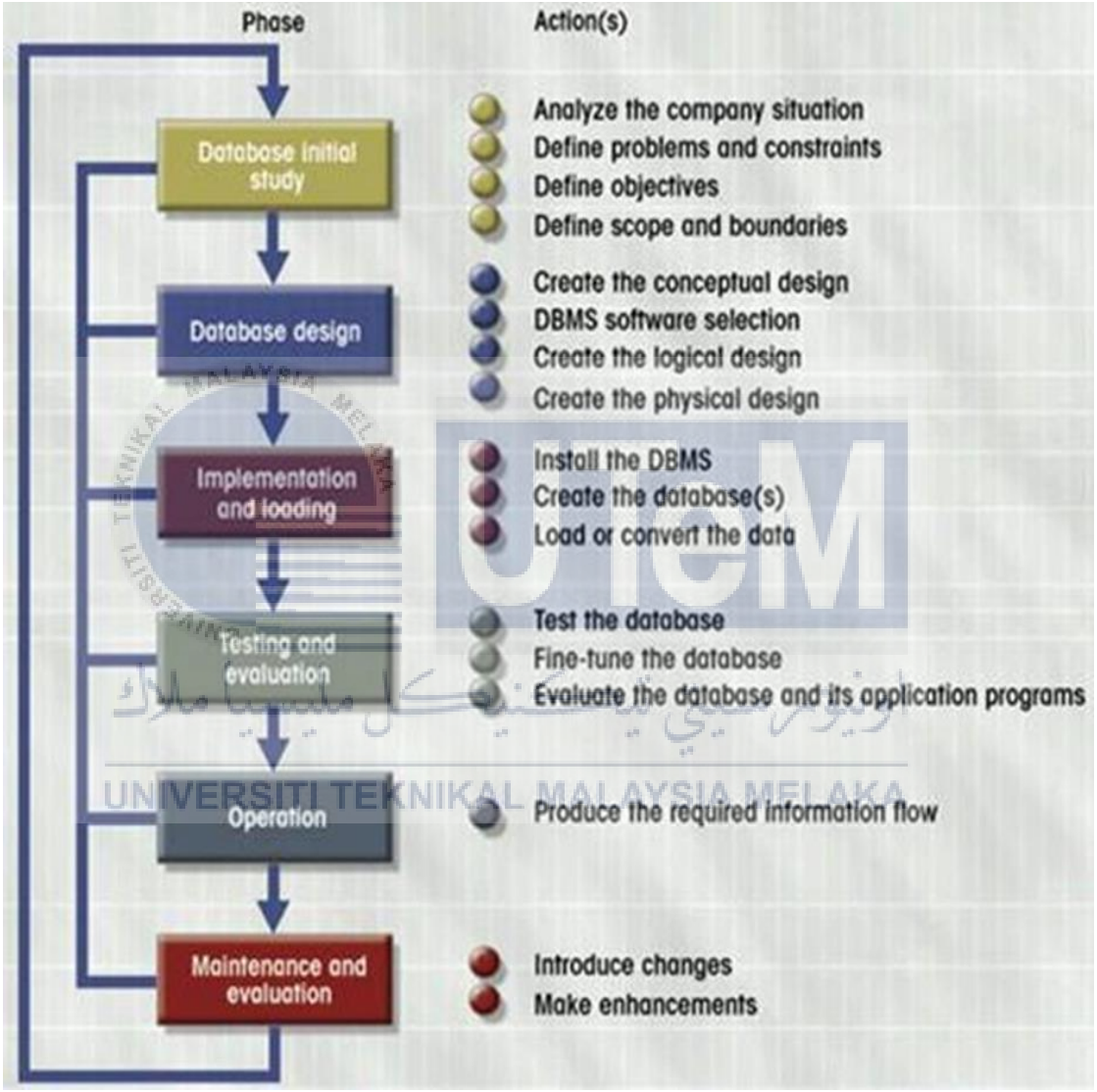


Figure 2.2: DBLC Methodology Diagram

2.2.2.1 Database Initial Study

The project starts in this phase and Database Life Cycle (DBLC) is used as project methodology. Firstly, gathering all the information about the current system and

the system requirement for this project. After throughout the analysis and observation of the current system, the problem statement of the system have been identified. Then, the objective of the project can be identify and project scope can be extracted to develop the system.

2.2.2.2 Database Design

During this phase, conceptual design for e-Report Management Report is been created. E-Report Management System are choosing Oracle 11g for the DBMS software. The minimum requirement for the installation need to be confirmed first in order for the DBMS in the server to run smoothly. Moreover, the Entity Relationship Diagram (ERD), and data dictionary is create where it will explain the main basic workflow of the system. All relationship between the tables, define the storage structures and the access paths will be known. While business rules that extract from a detailed description will help to create actions within the organizations environment. The business rules defined will properly describe the entities, attributes, relationship and connectivity and constraints.

2.2.2.3 Implementation and Loading

Next phase Implementation, this is done by installing the DBMS selected that is Oracle and creates the related data such as tables, triggers and stored procedure in the database. Then, the details gathered will be load and convert into data for example insert the data into table or entity correctly.

2.2.2.4 Testing and Evaluation

In testing and evaluation phase, the database will be tested and fine-tune the database. It is when the coding is being test through connection with database and then test the performance, integrity and security constraints. Next, the database and application programs will be evaluate whether it parallel with the system. If the database implementation fails to meet the user's requirement, several options will be in order enhancing the system.

2.2.2.5 Operation

Operation phase is to produce the required information flow into the system. At this point the database, management and users will compose a complete information system.

2.2.2.6 Maintenance and Evaluation

This last phase is about introducing new changes that will be added or updated into the system and make enhancements for a better system. The system developer will perform routine maintenance to the e-Report Management System which periodic maintenance require doing on the system backup, recovery, enhancing or normal maintenance.

2.3 Project Management

	Task Name	Duration	Start	Finish
1	**E-Report Management System for Damage of Facility for UTeM Residential College**	**80 days**	**Mon 22/2/16**	**Fri 10/6/16**
2	Proposal PSM: Submission & Presentation (Proposal Assessment and Verification)	5 days	Mon 22/2/16	Fri 26/2/16
3	Proposal Correction/Improvement (List of Supervisor/Title)	5 days	Mon 29/2/16	Fri 4/3/16
4	Chapter 1(System Development Begins)	5 days	Mon 7/3/16	Fri 11/3/16
5	Chapter 1 & Chapter 2	5 days	Mon 14/3/16	Fri 18/3/16
6	Chapter 2	5 days	Mon 21/3/16	Fri 25/3/16
7	Chapter 2 & Chapter 3 (Student Status)	5 days	Mon 28/3/16	Fri 1/4/16
8	Project Demo & Chapter 3, Chapter 4	5 days	Mon 4/4/16	Fri 8/4/16
9	MID TERM SEMESTER BREAK	5 days	Mon 11/4/16	Fri 15/4/16
10	Project Demo & Chapter 4	5 days	Mon 18/4/16	Fri 22/4/16
11	Project Demo & Chapter 4 (Student Status)	5 days	Mon 25/4/16	Fri 29/4/16
12	Project Demo (Determination of Student Status (Continue/Withdraw))	5 days	Mon 2/5/16	Fri 6/5/16
13	Project Demo & PSM Report	5 days	Mon 9/5/16	Fri 13/5/16
14	Project Demo & PSM Report (Presentation Schedule)	5 days	Mon 16/5/16	Fri 20/5/16
15	Project Demo & PSM Report	5 days	Mon 23/5/16	Fri 27/5/16
16	FINAL PRESENTAION (PA)	5 days	Mon 30/5/16	Fri 3/6/16
17	Revision Week (Correction Draft Report Based on Supervisor's and Evaluator's Comments During the Final Presentation Session)	5 days	Mon 6/6/16	Fri 10/6/16
18	FINAL EXAMINATION SEMESTER			

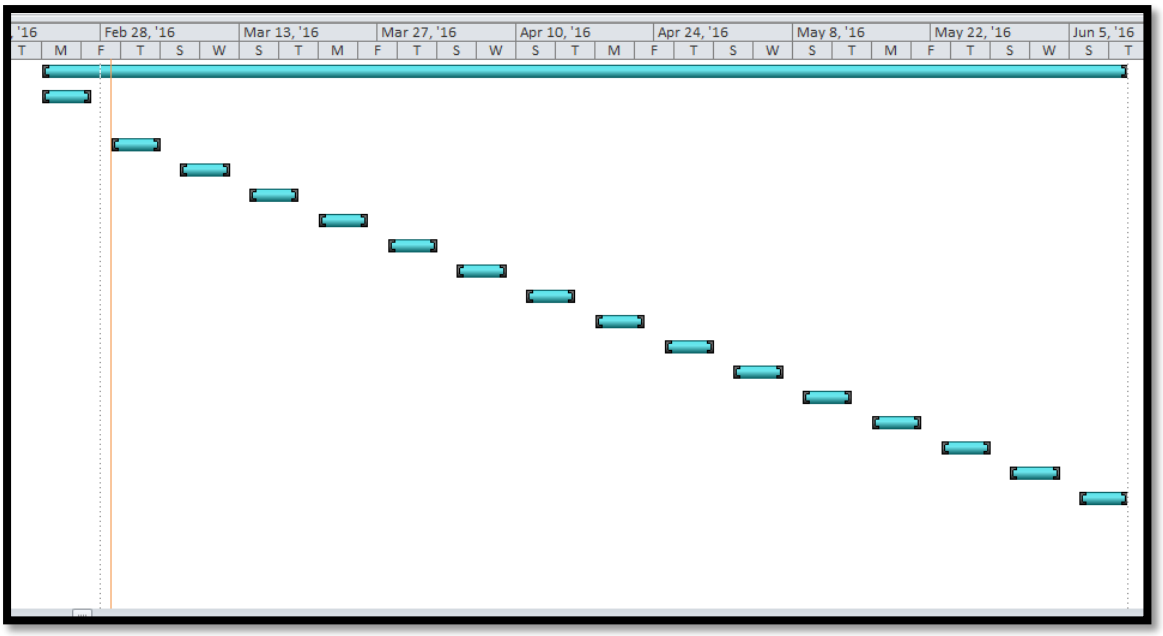


Figure 2.3: Gantt chart

2.4 Conclusion

As for conclusion, there are lots of benefits using agile method. For example agile methods grew out of the real-life project experiences of leading software professionals who had experienced the challenges and limitations of traditional waterfall development on project after project. The approach promoted by agile development is in direct response to the issue associated with traditional software development both in terms of overall philosophy as well as specific processes.

CHAPTER III

ANALYSIS

3.1 Introduction

The development of techniques of data analysis have 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 computer and be accessed by many users. The data can be gathered by conventional methods such as interviewing people in the organization and studying documents.

The facts can be represented as objects of interests. There are number of documentation tools available for data analysis, such as entity relationship diagrams (ERD). These are useful aids to communication, help to ensure that the work is carried out in a thorough manner, and ease the mapping processes that follow data analysis. Some of the documents can be used as source documents for the data dictionary.

The aims of this analysis is find the problem exists in the current system and helping in describing the problem in details, provide a sound understanding, both theoretical and practical, of basic systems theory, approaches to analysis, common methodologies, and the tasks of analysis and design in the context of developing computer-based information systems; the skills to apply such theory and methodologies, and to undertake such tasks a clear knowledge of the various requirements for a successful methodology in the field and confidence and skills to complete extended tasks using them. Besides, in order to collect

the requirement for a new system, analysis part are the main part for gathering all the requirement.

3.2 Problem Analysis

To develop and implement a system successfully, it is important to follow the procedures. The current system, they have no system that provide an auto-record data or information the report for damage of facility. The student need manually to write the form for the report. Data record for form of paper is high risk to missing and staff hard to find the information if they needed it. Another problem for the current system is time of report. The student need go the residential college office to make the report. The possible problem for student faced is they do not have much time to do the report. Student also have to go to office when office hour time and cannot do report in weekend.

3.3 The Proposed Improvements/Solutions

In order to analyse and understand the problem that exists in current system, E-Report Management System use flowchart as shown in figure below. After analyse the current system, it shows that all the management are work and communicate separately. In order to interact to each other, they need to communicate manually either by make a call or going to that management by themselves. Illustrate current system using Flowchart.

Figure 3.1 shows the current system of Registration Management. The system is initiated by the staff who is assigned in that management. Student have to fill in registration form in residential college office for apply admission to residential college. Staff require student's detail information including their email and phone number in order to complete the registration. Staff will notify the student whether student enrolment approved or not. If the enrolment approved, staff will arrange the room and student will receive the key.

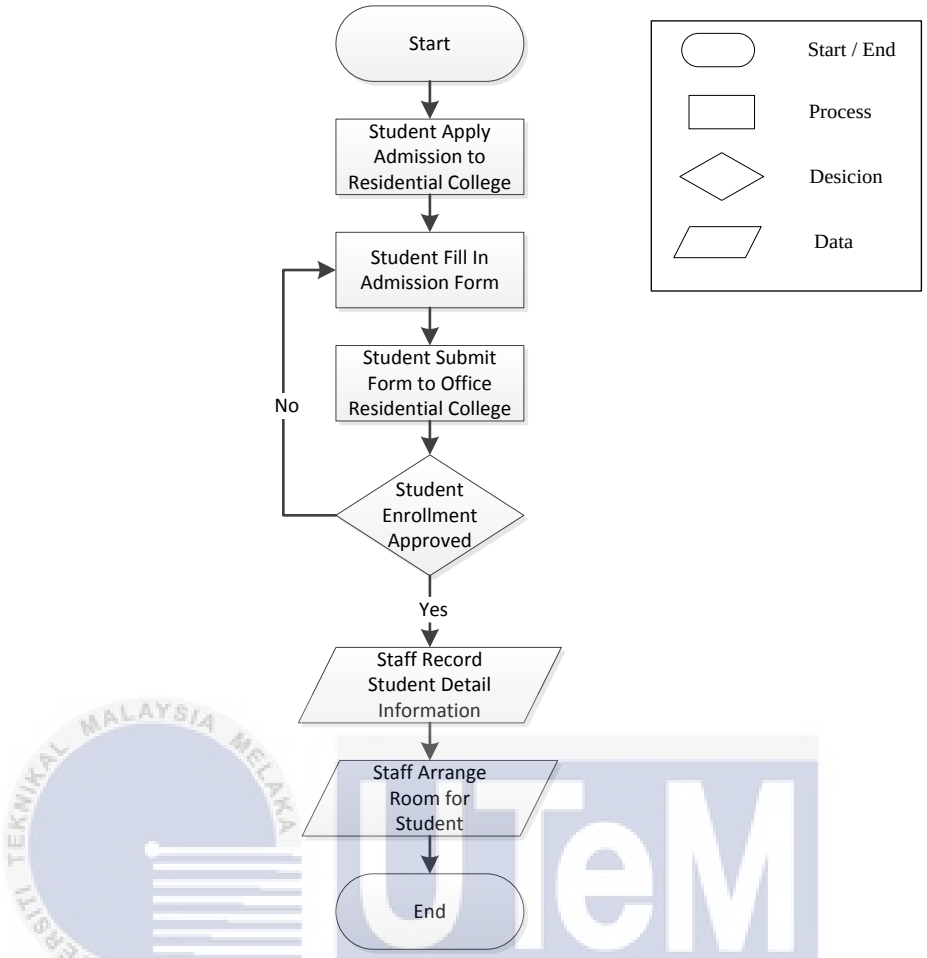


Figure 3.1: Flowchart Current System of Registration Management

Figure 3.2 shows the current system of Damage of Facility Report Management. The student have to go to Residential College Office in order to make a report for damage of facility in their house or room. Student have to fill in damage report form and describe the damage of that facility. After that, staff will identify the facility and take an action whether to repair or replace of that facility. Next, staff will identify and generate the charges of the damage and notify the student.

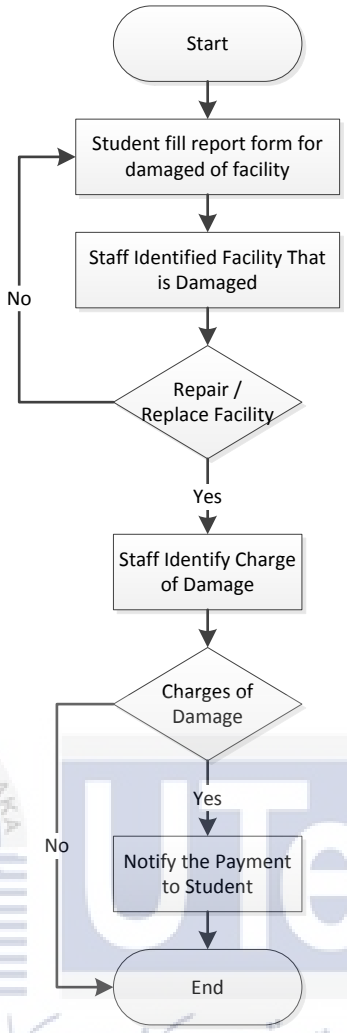


Figure 3.2: Flowchart Current System of Damage Report Management

Lastly figure 3.3 shows a current system of Payment Management. The staff will generate the total payment generating the total charges. The total charges will be generate if there are damage report of facility that been done by the student. After generating total payment, staff will notify the student the payment they have to make. Next after student making a payment, staff will update and print the payment information to makes as prove of the payments.

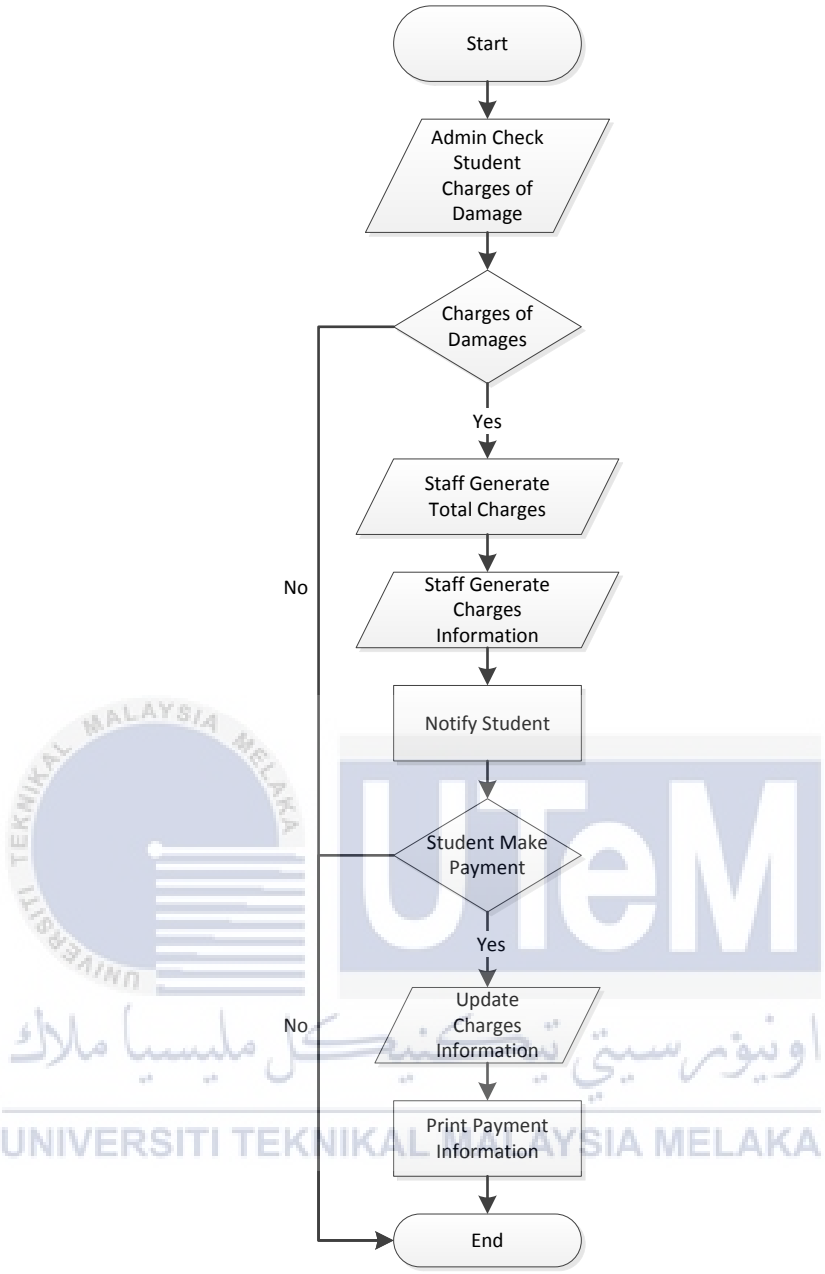


Figure 3.3: Flowchart Current System of Charges Payment Management

3.4 Requirement Analysis of the To-Be System

3.4.1 Functional Requirement (Process Model)

To analyse the database system that we are going to develop for this project, we use context diagram and the data flow diagram (DFD). This diagram is used to illustrate the system in details about what we are developing.

Figure 3.4 and Figure 3.5 shows the flow chart proposed to be develop for the E-Report Management System for Damage of Facility for Lestari Residential College. Figure 3.4 explain the flow for the damage of facility report and Figure 3.5 explain the generating and flow of payment for the charges of damage.

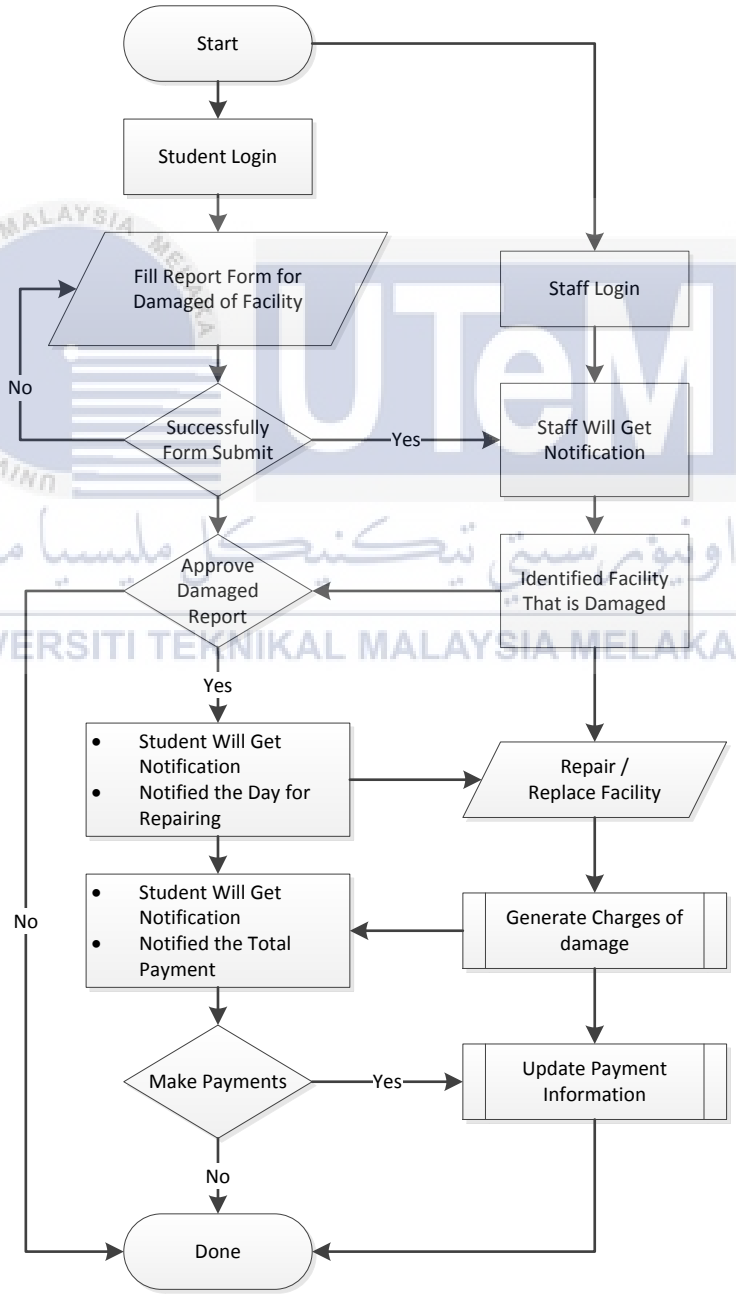


Figure 3.4: Flowchart To-Be System of Damage Report Management

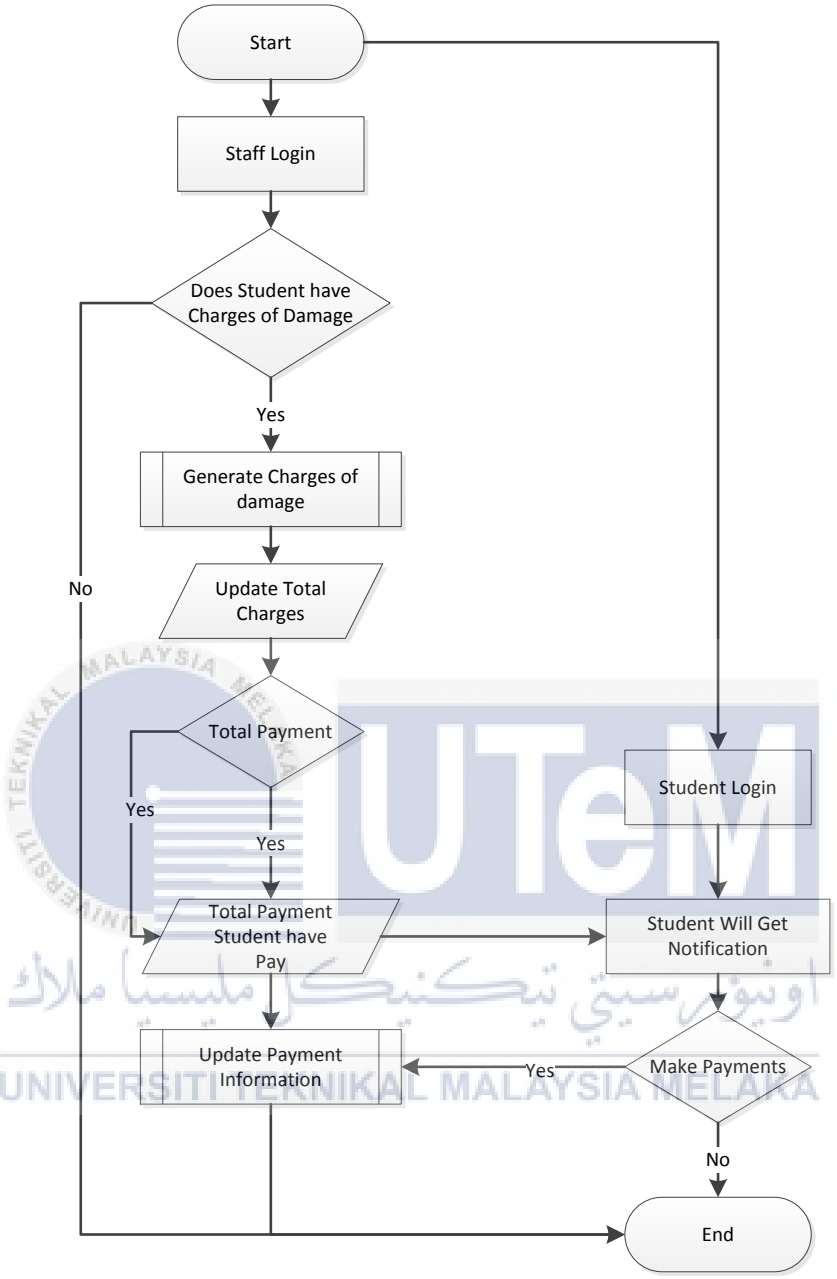


Figure 3.5: Flowchart To-Be System of Payment Management

Figure 3.6 shows context diagram for the system that is e-Report Management System for Damage of Facility for Lestari Residential College. This context diagram is to ensure that flow of the new system to be developed will be understood.

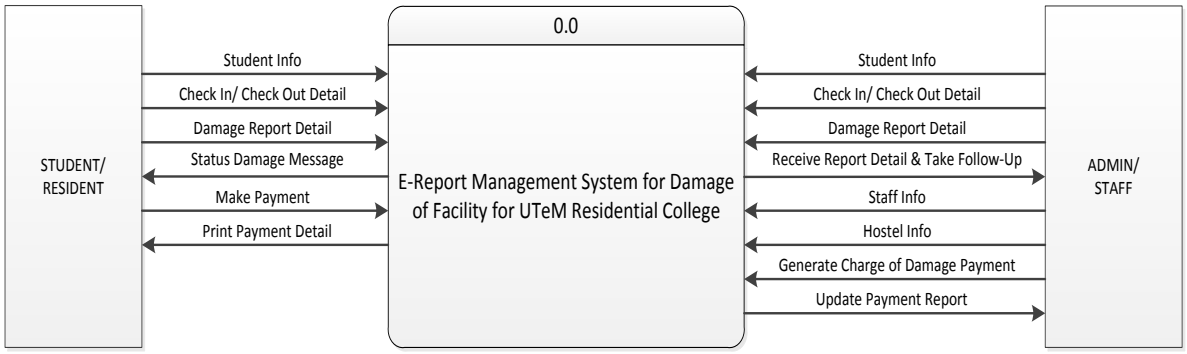


Figure 3.6: Context Diagram of e-Report Management System for Damage of Facility

Figure 3.7 shows a data flow diagram for Student. The data flow diagram shows the flow of student in the system. The student need to login before can use the system including the damage report management, charges management, printing management.

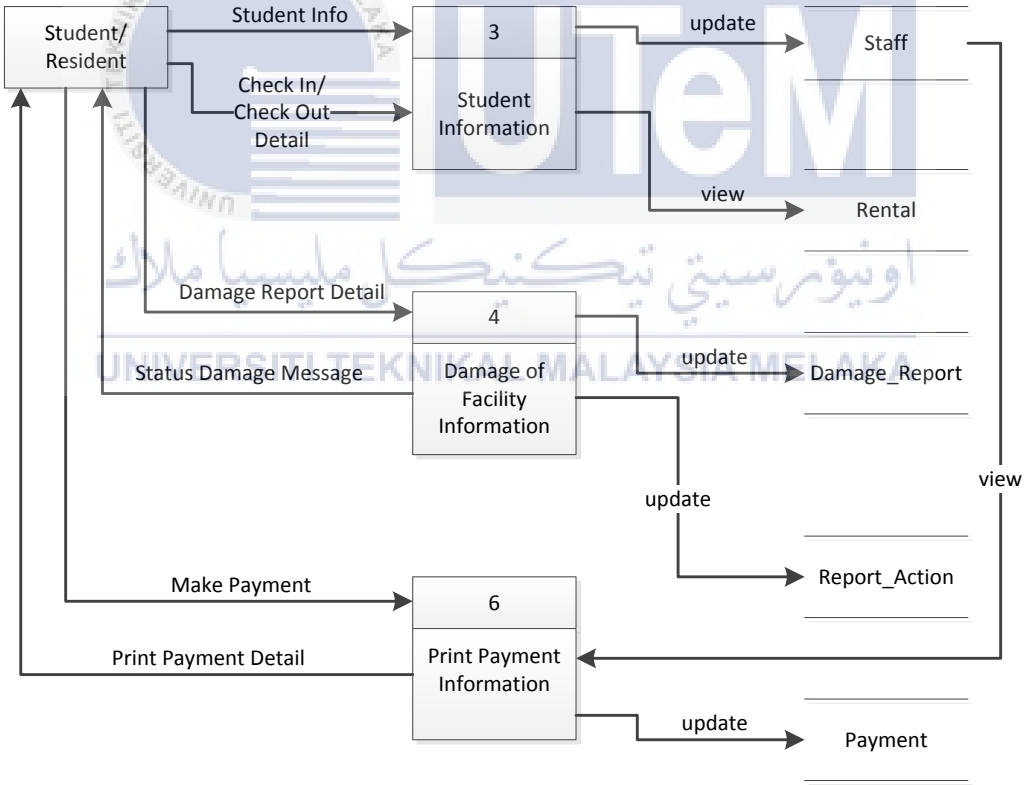


Figure 3.7: Data Flow Diagram for Student

Figure 3.8 shows a data flow diagram for Admin. The data flow diagram shows the flow of admin or staff in the system. The admin need to login before can use the system

including the hostel management, student management, damage report management payment management and printing management.

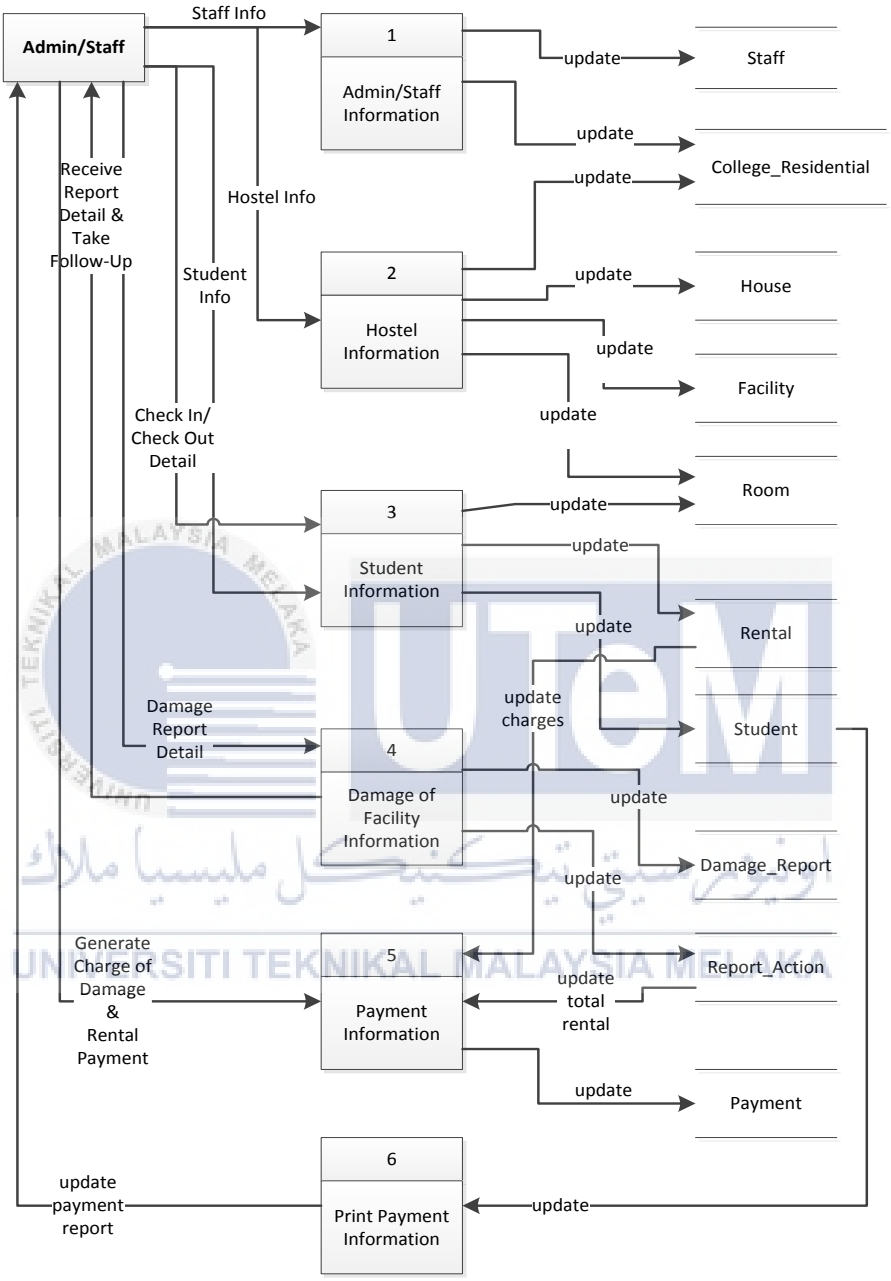


Figure 3.8: Data Flow Diagram for Admin

3.4.2 Non-functional Requirement

The non-functional requirement for this system that can be consider to make sure the system will functional well. The system maintainability is important because

one system take large amount of time to spend to make any changes. Secondly is reliability of the system, where the system has capability of the software to maintain its performance over time unreliable software fails.

Next is the security for the system. The integrity requirement determine properties of the security system or restrict access to its data to specific the users and protect the privacy of the data entered into the system.

3.4.3 Other Requirement

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

3.4.3.1 Software Requirement

There are requirement and specification of software components, which are been used in E-Report Management System. The software as described below:

Table 3.1: Software Requirement and Function

SOFTWARE	DESCRIPTION
Windows 8.1	Windows 8.1 is the operating system that suitable for client server application, and also used for the development computer and testing terminal.
Oracle Database	Oracle database (Oracle DB) is a relational database management system (RDBMS) from the Oracle Corporation. Originally developed in 1977 by Lawrence Ellison and other developers, Oracle Database is one of

	the most trusted and widely-used relational database engines.
PHP	PHP is a server-side scripting language designed for web development but also used as a general-purpose programming language.
Microsoft Office Word 2010	Microsoft Office Word 2010 is used to do the documentation for the proposal, project report and final report for PSM.
Microsoft Visio 2010	Microsoft Office Visio 2007 used to design Entity Relationship Diagram (ERD) and UML Diagram for develop the e-Report Management System.
Microsoft Office Project 2010	Microsoft Office Project 2010 is used to create Gantt Chart of the system such as plan, assigning to task, tracking progress, managing and analyzing for e-Report Management System.
SQLDeveloper	SQLDeveloper is used to create the entity, SQL query and connection to 9i.

3.4.3.2 Hardware Requirement

There have listed the requirement and specification of hardware components, which have been used in E-Report Management System. Such as:

i. Personal Computer Specification

- Intel Core 2 Duo Processor and above
- 2GB RAM and above

ii. Other Accessories

- Printer – printing documentation
- USB Drive – temporary storage and transferring data

External Hard Disk – backup all data project and source code for the system

3.5 Conclusion

In this chapter, the current system has been analysed from the stated requirements to be presented in the flow chart. In addition, the system that been developed is also represented in the form of context diagram. Context diagram explained the whole system to be developed in general. There is several site for each units where staff must connected to get the data from others.

CHAPTER IV

DESIGN

4.1 Introduction

This chapter will discuss about system design for E-Report Management System after all requirements of the system have been identified. The main objective of the design phase is to develop a design based on application requirements. The system design is explaining the system based on the flow of the system that are system architecture, user interface design, navigation design, input design and output design.

The results obtained from the analysis of the design will be used to produce computer specifications to solve this problem. All results are representative of the system design, including software specification for each function in E-Report Management System and database design to be used.

4.2 High-Level Design

High level design is to display an overview of each model set out in the requirements analysis phase. It explains the overall view of how the system should work and will study the user's functional and non-functional requirements and design an overall solution architecture of the application which can handle those needs.

4.2.1 System Architecture

One of the important activities in system architecture is designing the system. System architecture was identified during the first phase on the project development phase and identifies the needs analysis.

The system architecture of E-Report Management System is using two-tier architecture which consists of two layers that are client computers and database servers. The two-tier architecture is shown in Figure 4.1.

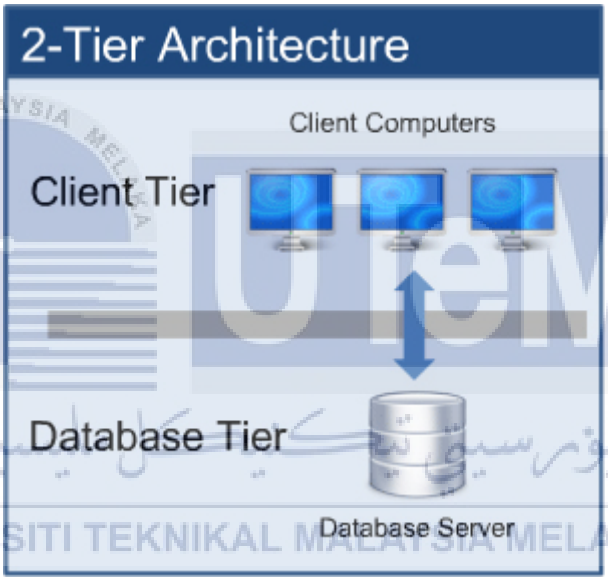


Figure 4.1: Two-Tier Architecture

4.2.2 User Interface Design

User interface is the medium that allows the users to communicate with the system. It is one of the most important parts because the good interfaces are depending on how many users can understand the way to use the system. User interface must be designed appropriately in order to make the interaction between the user and the system effective.

There are three main parts of the interface design that are navigation mechanism, input mechanism and output mechanism. The interfaces of this system are divided into 2 views of users that are the Admin or Staff and the Student that resident in Lestari Residential College. The description each the user interface will explain in Appendix.

4.2.2.1 Navigation Design

Navigation design is for describing how the user gives instructions to the system by using the buttons from each menu to enable the user to use every function in the system. The navigation design represents the user interfaces in the system. Figure 4.2 shows the navigation design of this system.

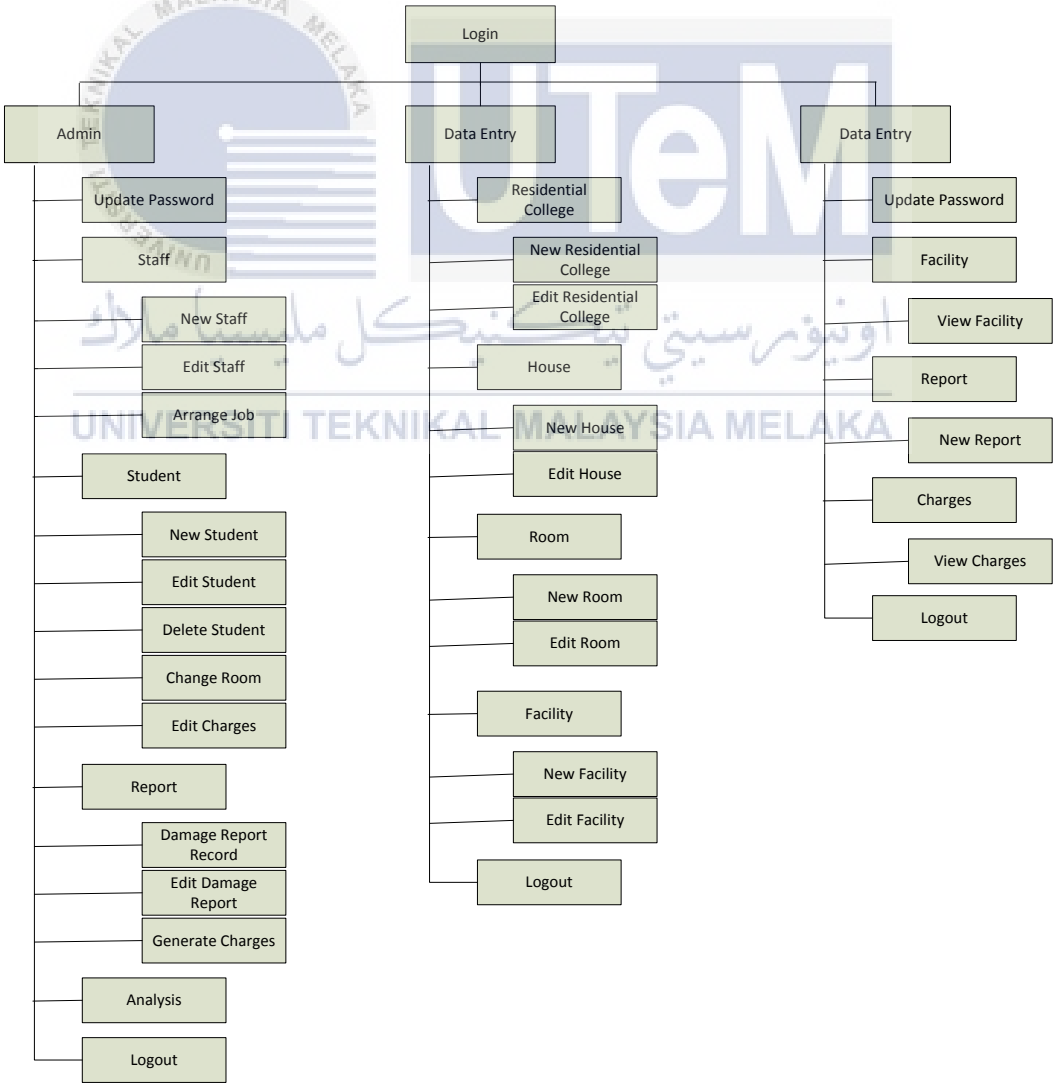


Figure 4.2: Navigation Design

4.2.2.2 Input Design

Input design is a design of types of input that will be at user interface. Example of input types is text, numbers, alphabet, symbol and many more. Some input is important and validation is needed to make sure the data will be saving to the database. Table 4.1 until Table 4.3 shows the input design in each form in the system.

Table 4.1: Input Design of Login Module

Form	Field Name	Input Type	Validation Rules
Login	User ID (Admin ID/Student ID)	Text Box	• Required field • Correct values will allow user to access Admin main page or Student main page. • Error message will pop up if null or incorrect values insert. Then, go back to Login Page.
	Password	Text Box	

Table 4.2: Input Design of Student Registration Module

Form	Field Name	Input Type	Validation Rules
Student/Resident Registration	Student ID	Text Box	• Required field
	Student Name	Text Box	• Required field
	Student Gender	Text Box	• Required field
	Student Phone No	Text Box	• Required field
	Student Email	Text Box	• Required field
	Student Faculty	Text Box	• Required field
	Student Year Study	Text Box	• Required field

	Student Semester	Text Box	• Required field
	Student Password	Text Box	• Required field

Table 4.3: Input Design of Damage Report Module

Form	Field Name	Input Type	Validation Rules
Damage of Facility Report	Damage Item Name	Text Box	• Read only
	Damage Description	Text Box	• Required field

4.2.2.3 Output Design

The output design is the design of the reports will produce by the system. The main purpose is to deliver the information in a clear way and easy to understand. The output design result will be shown in the following Table 4.4.

Table 4.4: Output Design Module

Form Name	**Output Name**	**Description**
Login	Login Error Message	The login error message will appear when the admin/student had entered the wrong user ID and Password
Student Registration	Validation Message	Farmer must enter all information required or the ID already exist

4.2.3 Conceptual and Logical Database Design

There are two parts in this section, namely Conceptual Database Design and Logical Database Design. Conceptual database design is the process of producing a detailed data model of a database. This logical data model contains all the needed

logical and physical design choices and physical storage parameters needed to generate a design in a Data Definition Language, which can then be used to create a database. A fully attributed data model contains detailed attributes for each entity.

4.2.3.1 Conceptual Database Design

In this phase, the Business Rules and Entity Relationship Diagram (ERD) of this system to show the flow of the data conceptually.

1. **Business Rule**
   i. One Residential College have many house in that particular Residential College
   ii. One Residential College can employed many staff and staff can be employed one or not in that Residential College
   iii. One house have many room while one room can be in one house
   iv. One house can have many facility while facility can be in one house
   v. One room can have many facility while facility can be in one room
   vi. One room can be rental many time while one rental can only rent one room
   vii. One student can rent one room in one particular semester or not rental room at all
   viii. One facility can be done reporting the damage many time while one damage report can have one facility
   ix. One student can do many report on many damage of facility while one report can be done by one student
   x. Admin can make staff (Technician) do many job on conducting damage of facility report while one damage report can be conducting by one technician
   xi. One report can do many damage report action while one report action only have one damage report
   xii. Many damage report can be paid in one payment while one payment can have one or many damage report

2. **Entity Relationship Diagram (ERD)**

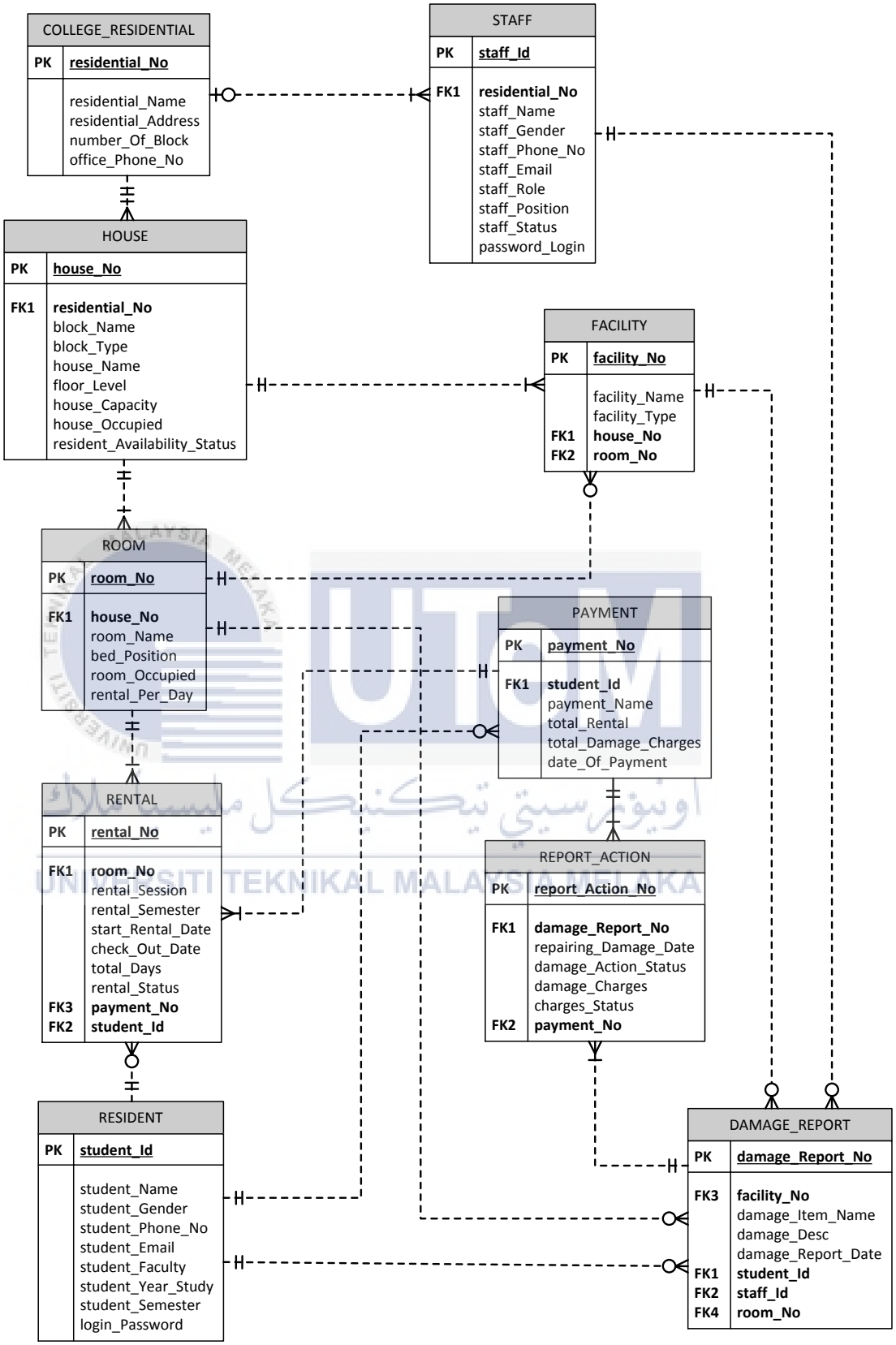


Figure 4.3: Entity Relationship Diagram (ERD)

4.2.3.2 Logical Database Design

Logical Database Design is used to translate the conceptual representation to the logical structure of the database. It represents the data dictionary. The logical entity relationship model contains more detail than the conceptual entity relationship model. In addition to master data entities, operational and transactional data entities are now defined. The details of each data entity are developed and the entity relationships between these data entities are established.

Table 4.5: Data Dictionary of Residential College Table

Attribute Name	Data Type	Length	Key	Description
residential_No	varchar2	15	PK	Residential No for Lestari Residential College
residential_Name	varchar2	75		Name of Residential
residential_Address	varchar2	250		Residential Address
number_Of_Block	number			Number of Block
office_Phone_No	varchar2	25		Residential Office Number

Table 4.6: Data Dictionary of House Table

Attribute Name	Data Type	Length	Key	Description
house_No	varchar2	15	PK	House No
residential_No	varchar2	15	FK	References from table Residential College
block_Name	varchar2	20		Block Name
block_Type	varchar2	20		Residential College Block Based on Resident Gender
house_Name	varchar2	20		House Name
floor_Level	varchar2	10		House Floor Level

house_Capacity	number			House Capacity for Rental
house_Occupied	number			Number of Occupied
resident_Availability_ Status	varchar2	30		House Status Availability

Table 4.7: Data Dictionary of Room Table

Attribute Name	Data Type	Length	Key	Description
room_No	varchar2	15	PK	Room No
house_No	varchar2	15	FK	References from table House
room_Name	varchar2	20		Room Name
bed_Position	varchar2	20		Bed Position
room_Occupied	varchar2	30		Room Status Availability
rental_Per_Day	number			Room Rental Per Day

Table 4.8: Data Dictionary of Facility Table

Attribute Name	Data Type	Length	Key	Description
facility_No	varchar2	15	PK	Facility No
facility_Name	varchar2	75		Name of Facility
facility_Type	varchar2	30		Type of Facility
house_No	varchar2	15	FK	References from table House
room_No	varchar2	15	FK	References from table Room

Table 4.9: Data Dictionary of Resident Table

Attribute Name	Data Type	Length	Key	Description
student_Id	varchar2	15	PK	Student ID
student_Name	varchar2	75		Student Name

student_Gender	varchar2	20		Student Gender
student_Phone_No	varchar2	25		Student Contact Number
student_Email	varchar2	150		Student Email
student_Faculty	varchar2	30		Student Faculty
student_Year_Study	varchar2	10		Year of Study
student_Semester	varchar2	30		Semester
login_Password	varchar2	30		Password Login

Table 4.10: Data Dictionary of Staff Table

Attribute Name	Data Type	Length	Key	Description
staff_Id	varchar2	15	PK	Staff ID/Admin ID
residential_No	varchar2	15	FK	References from table Residential College
staff_Name	varchar2	75		Staff/Admin Name
staff_Gender	varchar2	20		Staff/Admin Gender
staff_Phone_No	varchar2	25		Staff/Admin Contact Number
staff_Email	varchar2	150		Staff/Admin Email
staff_Role	varchar2	30		Role of Staff/Admin in Residential College
staff_Position	varchar2	30		Position in Residential College
staff_Status	varchar2	30		Staff Status
login_Password	varchar2	30		Password Login

Table 4.11: Data Dictionary of Payment Table

Attribute Name	Data Type	Length	Key	Description
payment_No	varchar2	15	PK	Payment No
student_Id	varchar2	15	FK	References from table Resident
payment_Name	varchar2	150		Payment Description
total_Rental	number	5,2		Total Rental
total_Damage_Charges	number	5,2		Total Damage Charges
date_Of_Payment	date			Date of Payment

Table 4.12: Data Dictionary of Rental Table

Attribute Name	Data Type	Length	Key	Description
rental_No	varchar2	15	PK	Rental No
room_No	varchar2	15	FK	References from table Room
rental_Session	varchar2	20		Rental Session
rental_Semester	varchar2	20		Rental in Semester
start_Rental_Date	date			Check In Residential College
check_Out_Date	date			Check Out
total_Days	number			Rental Total Days
rental_status	varchar2	50		Rental Status
payment_No	varchar2	15	FK	References from table Payment
student_Id	varchar2	15	FK	References from table Resident

Table 4.13: Data Dictionary of Damage Report Table

Attribute Name	Data Type	Length	Key	Description
damage_Report_No	varchar2	15	PK	Damage Report No
facility_No	varchar2	15	FK	References from table Facility
damage_Item_Name	varchar2	75		Name of Damage Item
damage_Desc	varchar2	150		Damage Description
damage_Report_Date	date			Date of Report
student_Id	varchar2	15	FK	References from table Resident
staff_Id	varchar2	15	FK	References from table Staff
room_No	varchar2	15	FK	References from table Room

Table 4.14: Data Dictionary of Report Action Table

Attribute Name	Data Type	Length	Key	Description
report_Action_No	varchar2	15	PK	Report Action No
damage_Report_No	varchar2	15	FK	References from table Damage Report
repairing_Damage_Date	date			Report Action Date
damage_Action_Status	varchar2	30		Damage Report Status
damage_Charges	number	3,2		Charges of Damage
charges_Status	varchar2	30		Charges Status
payment_No	varchar2	15	FK	References from table Payment

4.3 Data Definition Language (DDL)

DDL is stands for "Data Definition Language". DDL is a language used to define data structures within a database. DDL is used to create, alter or drop database and database objects. The database objects consist of schemas, tables, views, sequences and indexes. This command is used by database administrators during the setup and removal phases of databases objects. DDL statement will be created and compiled to show the output. DDL is also known as a computer language for defining the data structures.

i. Table Residential College

```
create table RESIDENTIAL_COLLEGE (
residential_No varchar2(15) PRIMARY KEY,
residential_Name varchar2(75),
residential_Address varchar2(250),
number_Of_Block number,
office_Phone_No varchar2(25)
);
```

ii. Table House

```
create table HOUSE (
house_No varchar2(15) PRIMARY KEY,
residential_No REFERENCES RESIDENTIAL_COLLEGE(residential_No),
block_Name varchar2(20),
block_Type varchar2(20),
house_Name varchar2(20),
floor_Level varchar2(10),
house_Capacity number,
house_Occupied number,
resident_Availability_Status varchar2(30)
);
```

iii. Table Room

```
create table ROOM (
room_No varchar2(15) PRIMARY KEY,
house_No REFERENCES HOUSE(house_No),
room_Name varchar2(20),
bed_Position varchar2(20),
room_Occupied varchar2(30),
rental_Per_Day number
);
```

iv. Table Facility

```
create table FACILITY (
facility_No varchar2(15) PRIMARY KEY,
facility_Name varchar2(75),
facility_Type varchar2(30),
house_No REFERENCES HOUSE(house_No),
room_No REFERENCES ROOM(room_No)
);
```

v. Table Resident

```
create table RESIDENT (
student_Id varchar2(15) PRIMARY KEY,
student_Name varchar2(75),
student_Gender varchar2(20),
student_Phone_No varchar2(25),
student_Email varchar2(150),
student_Faculty varchar2(30),
student_Year_Study varchar2(10),
student_Semester varchar2(30),
login_Password varchar2(30)
);
```

vi. Table Staff

```
create table STAFF (
```

```
staff_Id varchar2(15) PRIMARY KEY,
residential_No REFERENCES RESIDENTIAL_COLLEGE(residential_No),
staff_Name varchar2(75),
staff_Gender varchar2(20),
staff_Phone_No varchar2(25),
staff_Email varchar2(150),
staff_Role varchar2(30),
staff_Position varchar2(30),
staff_Status varchar2(30),
login_Password varchar2(30)
);
```

vii. Table Payment

```
create table PAYMENT (
payment_No varchar2(15) PRIMARY KEY,
student_Id REFERENCES RESIDENT(student_Id),
payment_Name varchar2(150),
total_Rental number(5,2),
total_Damage_Charges number(5,2),
date_Of_Payment date
);
```


viii. Table Rental

```
create table RENTAL (
rental_No varchar2(15) PRIMARY KEY,
room_No REFERENCES ROOM(room_No),
rental_Session varchar2(20),
rental_Semester varchar2(20),
start_Rental_Date date,
check_Out_Date date,
total_Days number,
rental_status varchar2(50),
payment_No REFERENCES PAYMENT(payment_No),
```

```
student_Id REFERENCES RESIDENT(student_Id)
);
```

ix. Table Damage Report

```
create table DAMAGE_REPORT (
damage_Report_No varchar2(15) PRIMARY KEY,
facility_No REFERENCES FACILITY(facility_No),
damage_Item_Name varchar2(75),
damage_Desc varchar2(150),
damage_Report_Date date,
student_Id REFERENCES RESIDENT(student_Id),
staff_Id REFERENCES STAFF(staff_Id),
room_No REFERENCES ROOM(room_No)
);
```

x. Table Report Action

```
create table REPORT_ACTION (
report_Action_No varchar2(15) PRIMARY KEY,
damage_Report_No references DAMAGE_REPORT(damage_Report_No),
repairing_Damage_Date date,
damage_Action_Status varchar2(30),
damage_Charges number(3,2),
charges_Status varchar2(30),
payment_No REFERENCES PAYMENT(payment_No)
);
```


4.4 Graphical User Interface (GUI)

The E-Report Management System is divided into two user scope which is admin and student. The admin and student have difference in the system scope and interfaces except by Login Interface. The interface for each role will be display below.

1. Login Interface

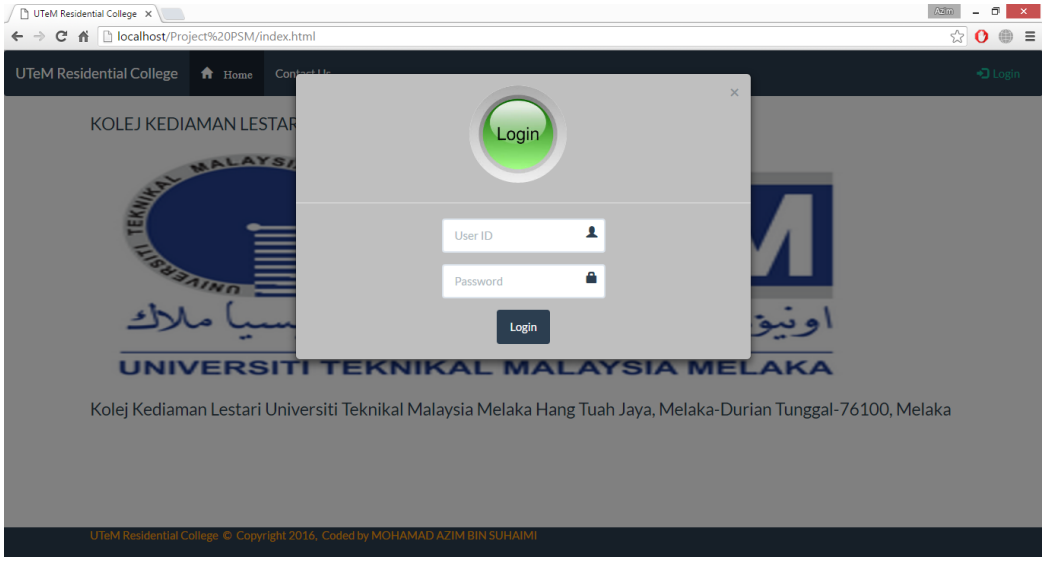


Figure 4.4: GUI for Login Interface

2. Student

Figure 4.5: GUI for List Room Facility

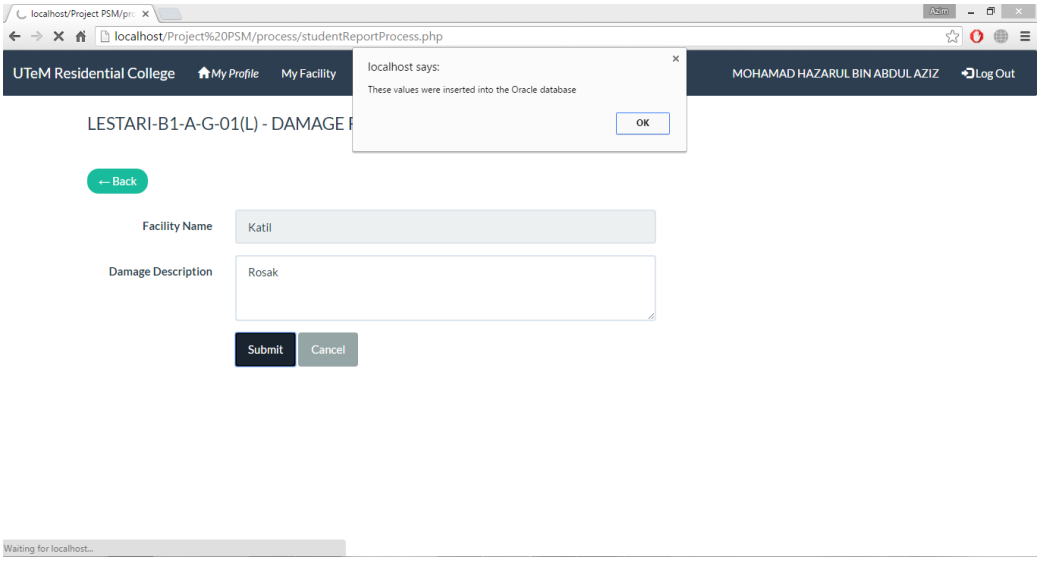


Figure 4.6: GUI for Damage of Facility Report

3. Admin

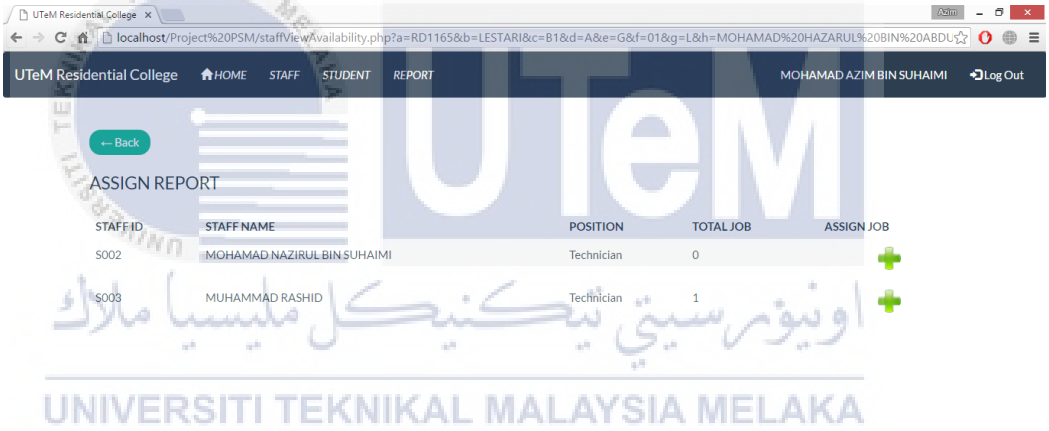


Figure 4.7: GUI for Assign Report to Staff (Technician)

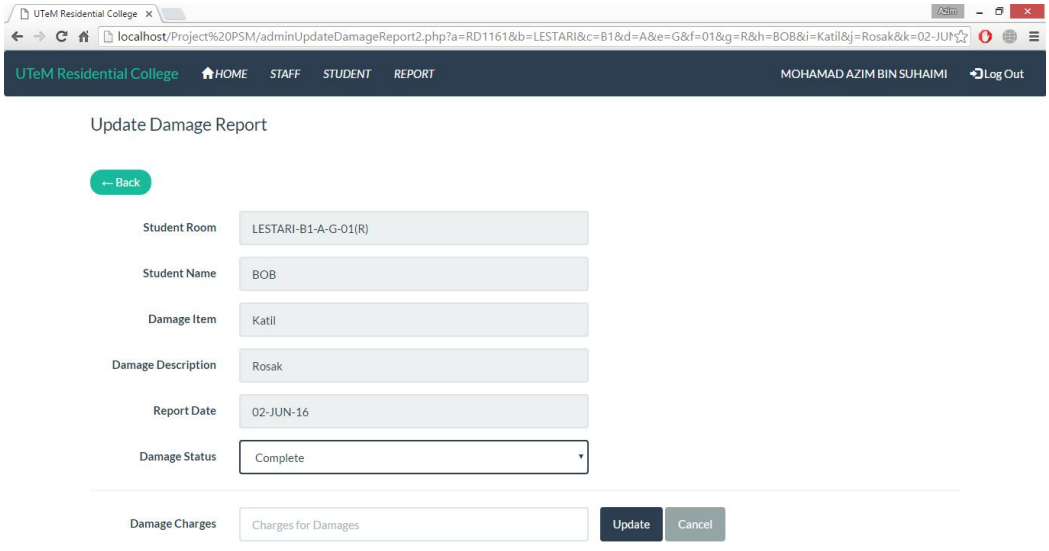


Figure 4.8: GUI for Update in Process Damage Report

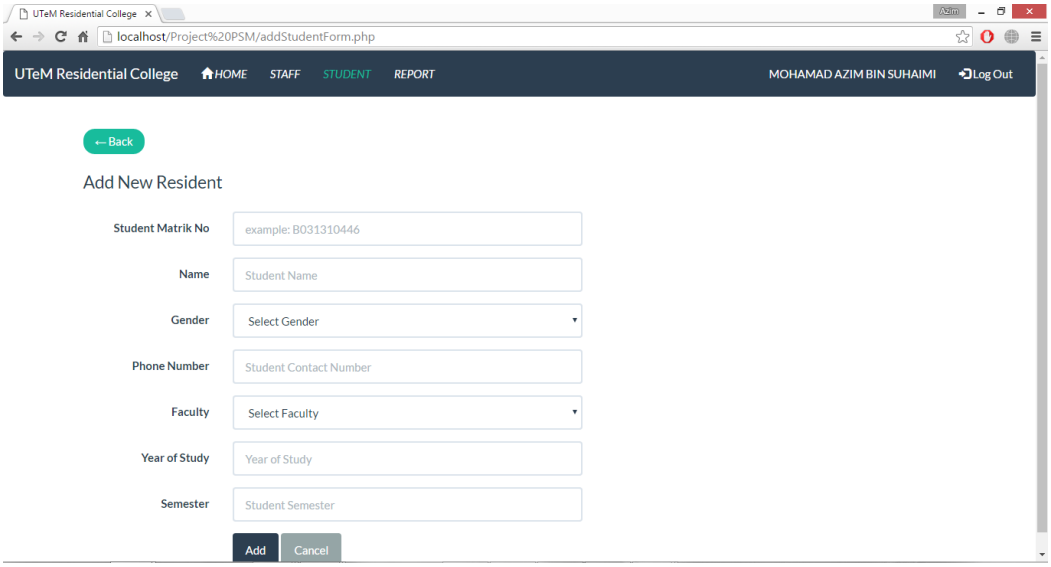


Figure 4.9: GUI for Student Registration

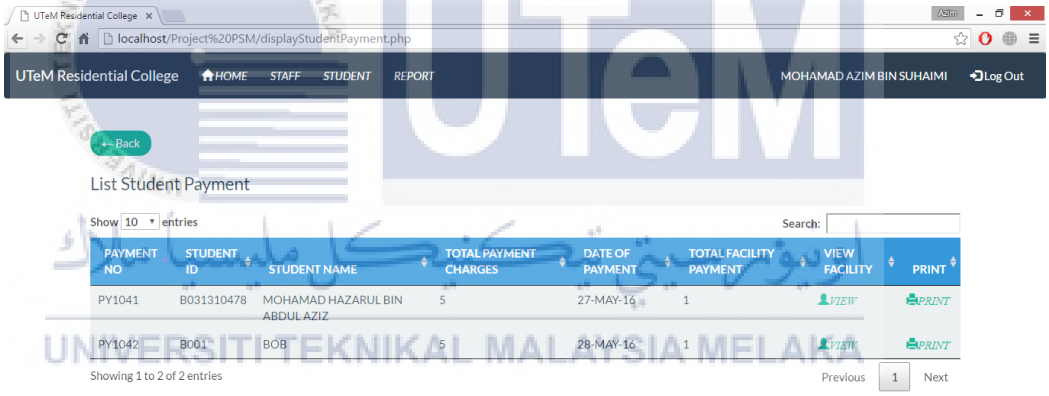


Figure 4.10: GUI for Payment of Damage Charges

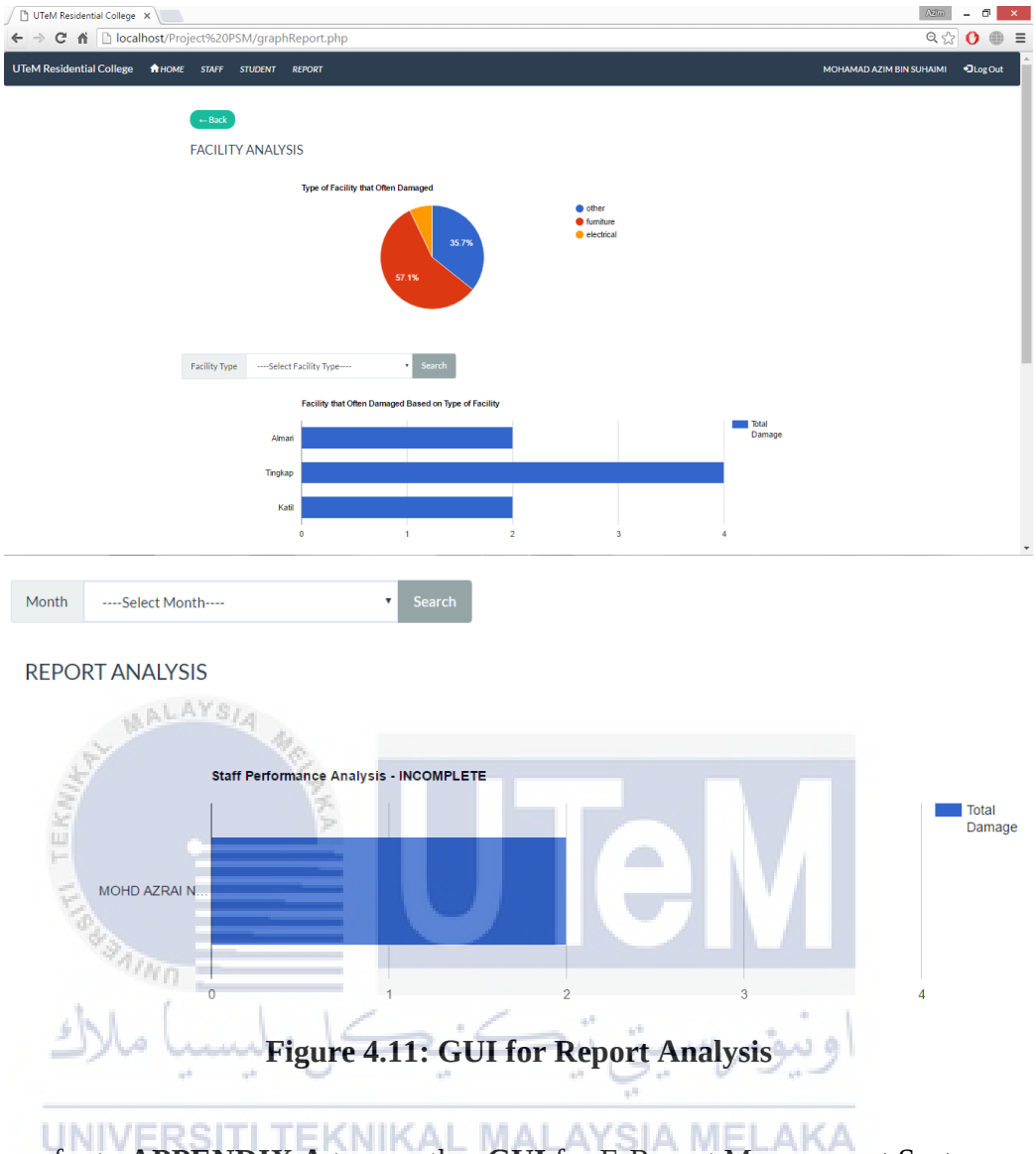


Figure 4.11: GUI for Report Analysis

Please refer to **APPENDIX A** to see other **GUI** for E-Report Management System.

4.5 Conclusion

Based on the on logical, physical and data dictionary that have been provide in this document, the overall of the database structure have been created with the specific module that will be develop in database Oracle. Overall view of the conceptual design, it shows the relationship for each entity that depending with each other**.** This database design makes the planning of the system become easier with a specific guideline. The next activity that needs to complete is implementation.

CHAPTER V

IMPLEMENTATION

5.1 Introduction

During the implementation phase, the application development environment needs to be managed and the environment needs to be configured. This phase will also need to display the status of implementation of the outcome to be specified in detail. During this phase, the focus will be devoted to the task of programming and system testing. In programming, the task will be carried out will involve writing code in PHP programming powerful program.

This chapter will discuss in details about the implementation of the E-Report Management System that divided into two which is system development and database development. The implementation including the software development environment setup which is covers the architecture of client and server software and database that will be used for the system development. Then, software configuration management covers about the configuration of the software to fulfil system requirements.

Therefore, this work will identify areas which have been completed or are still in development. For those still in development, hard work needs to be done to complete on time set. E-Report Management System function properly and achieve the objectives of which system administrators and client through the process of using the system properly.

5.2 Software Development Environment System

A development environment contains everything required by a team to build and deploy software-intensive systems (where software is an essential and indispensable element). This section should be considered as the initial setup for the software development environment setup for E-Report Management System project. It will describe the components necessary for the development environment. The system has two levels which is level one interaction between user and system through GUI. Meanwhile for level two, it covers about the communication between system and database which is the database let the user to retrieve and manipulate the data based on their authorization setting.

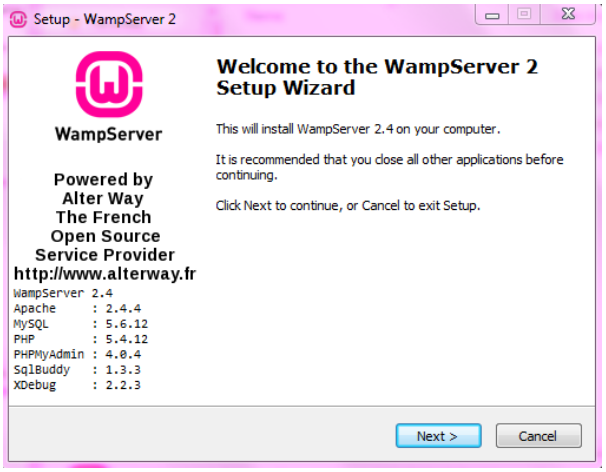
The software development environment setup of E-Report Management System is using three-tier architecture which consists of client tier, middle tier, and database tier. For client tier, user needs to use web browser to access the system. Then middle tier which is application server let the user to interact with database server to retrieve or manipulate the data with it. Each of the tiers should work together in good condition to ensure the system can run smoothly.

5.2.1 Software Environment System

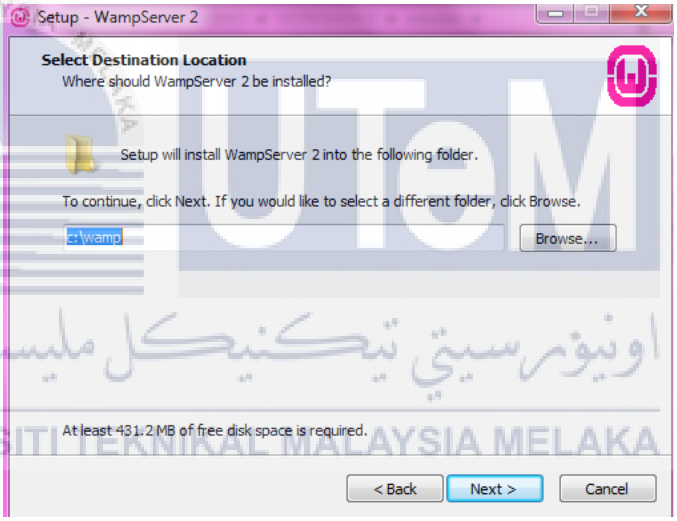
To develop the E-Report Management System, developers must have a computer that has installed Apache Server and PHP 5.4 which acts as a system design. Thus, WAMP Server 2.4 is installed because it is easy to configure and is compatible with the Windows operating system and come with Apache Server and PHP 5.4 in package.

i. WAMP Server Installation

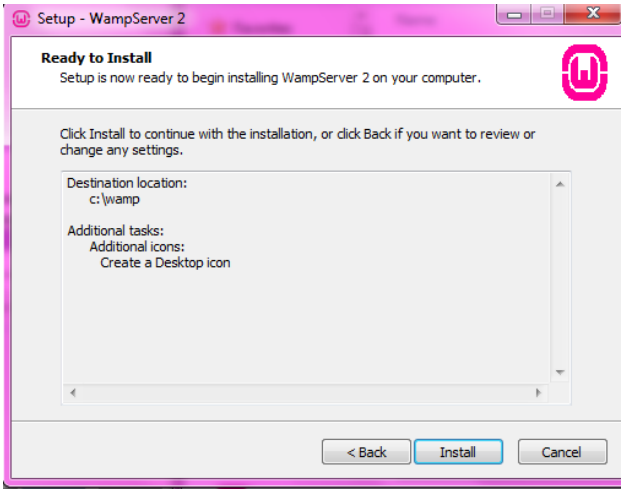
STEP 1: Run that exe file of the WAMP server. It will get the bellow window. Then, it needs to proceed with the following steps.



STEP 2: It should to agree the license of WAMP Server before selecting the installation destination at windows machine. Then, install WAMP Server in C: driver. Click next when finished.



STEP 3: Click on the Install button to start installation of WAMP Server.



STEP 4: Allow the access to Apache server at SMTP server configuration dialog. Then, specify the SMTP server and the address mail to be used by PHP when using the function mail (). It is will recommend the following values, in local host.

STEP 5: Completing installed WAMP Server 2.4 dialog with Apache Server and PHP 5.4 at computer. Click button "Finish" to start WAMP Server along with other services.

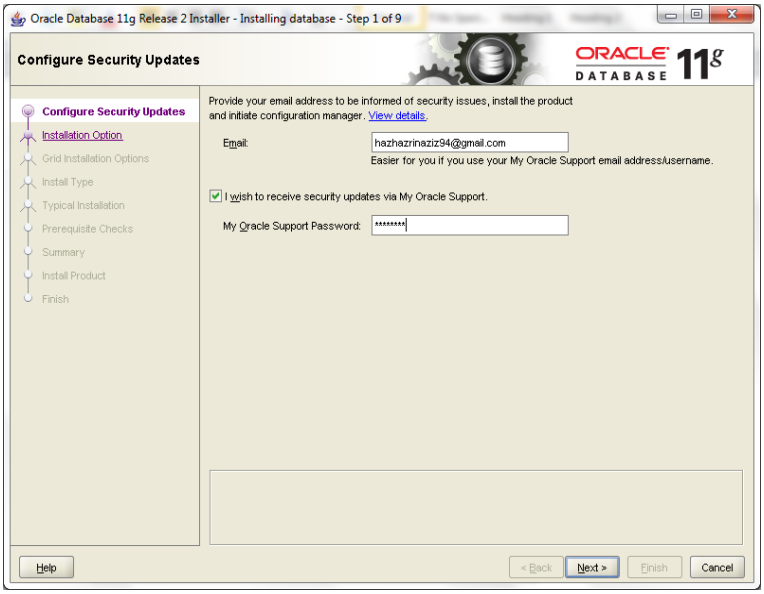
For this project, a folder name *"Project PSM"* was created with directory "*C:\wamp\www\Project PSM*" and when "*http://localhost"* is typing at the web browser it will show the main page of WAMP Server and the folder "*Project PSM*" will appear under "Your Projects". Clicking on it will access user to the main index page of E-Report Management System.

5.2.2 Database Environment System

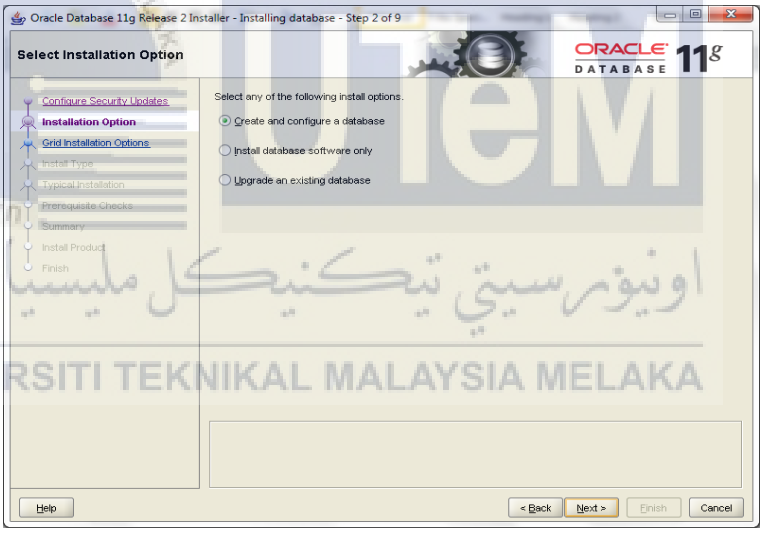
During the database environment setup, developer must configure the setting for database connection and grant some administration permission to access the database. E-Report Management System used Oracle 11g Enterprise Edition as a database and SQL Developer. For this project, a database connection name is "azim", username "azim" and password "1234" with hostname "localhost" and port "1521".

i. Oracle 11g Installation

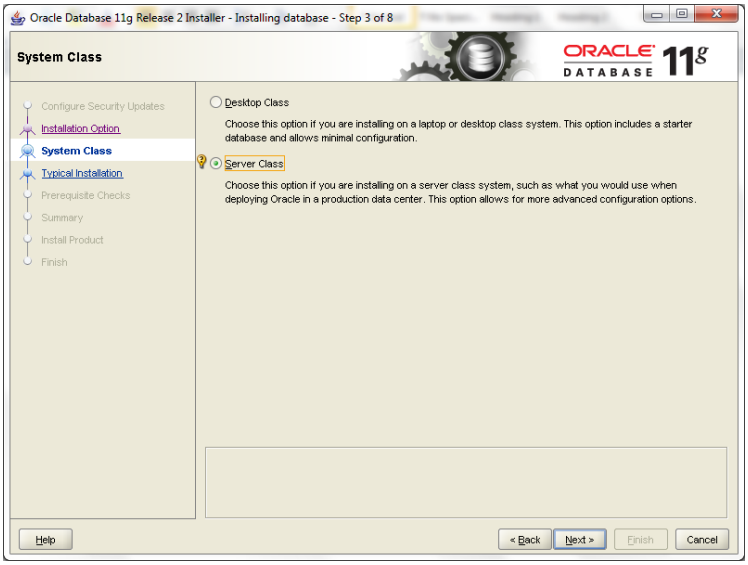
STEP 1: Run the Oracle 11g Enterprise Edition installer and fill the configure security updates and click the next button.



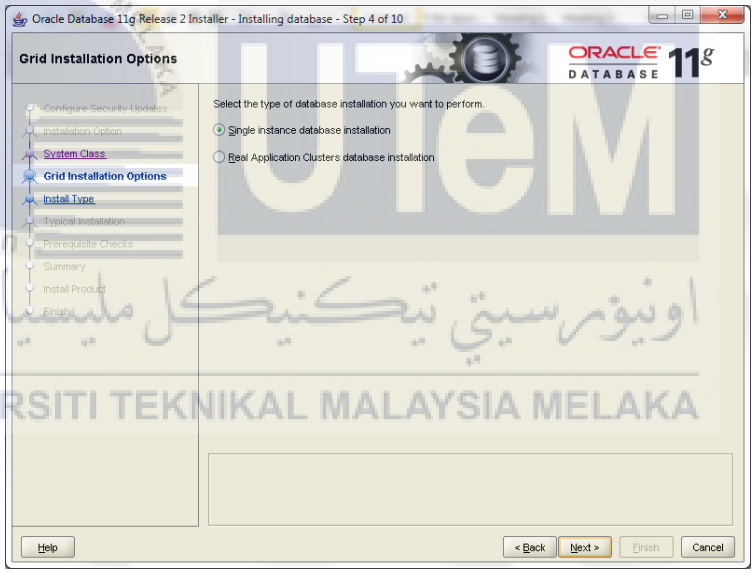
STEP 2: Select the installation option by clicking the options on create and configure a database.



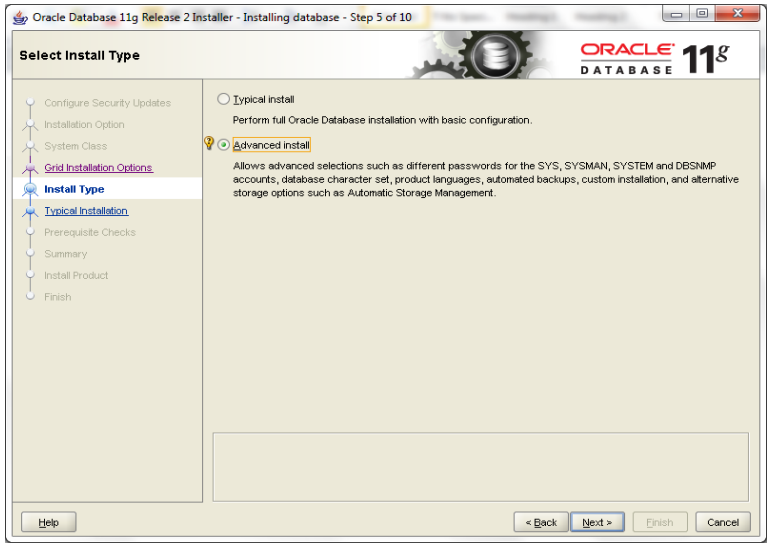
STEP 3: Select the system class as server class.



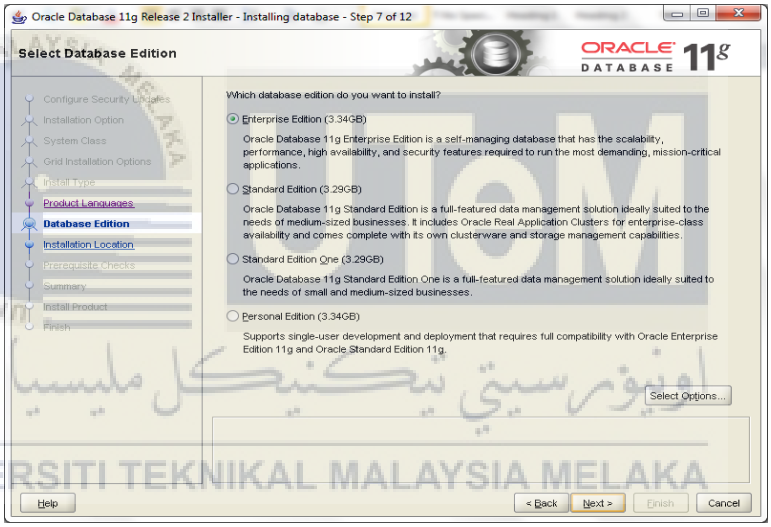
STEP 4: Select the single instance database installation on grid installation options.



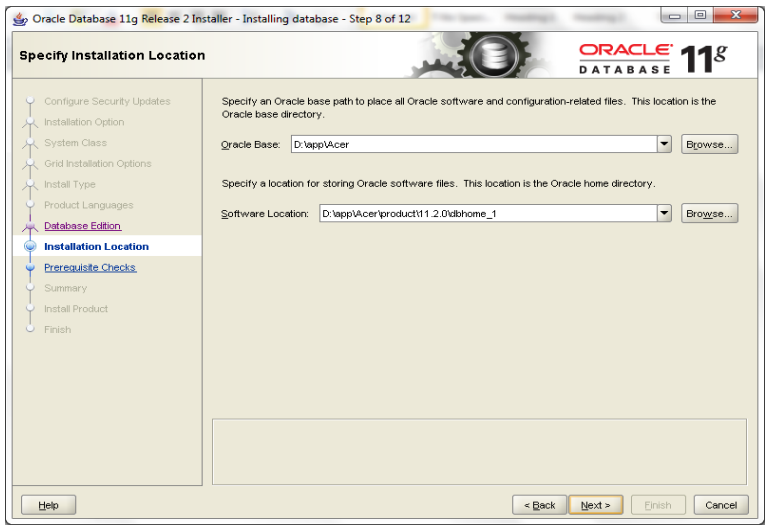
STEP 5: Select the install type as advanced install.



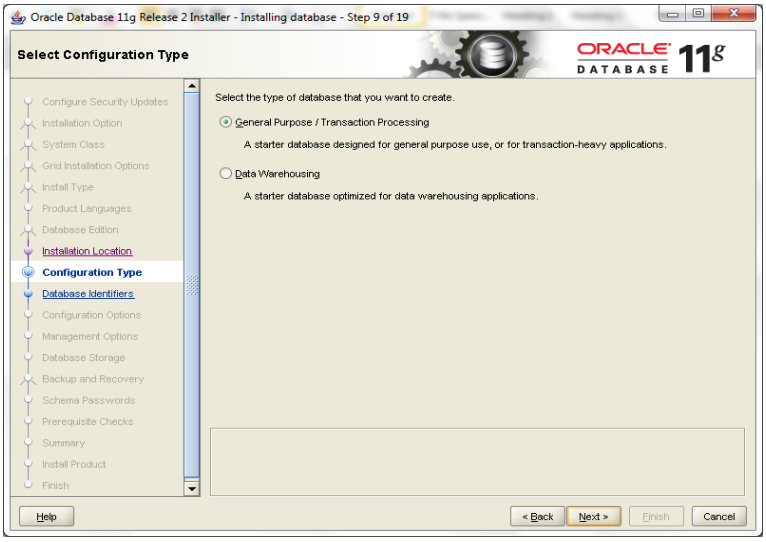
STEP 6: Select the database edition as Enterprise Edition.



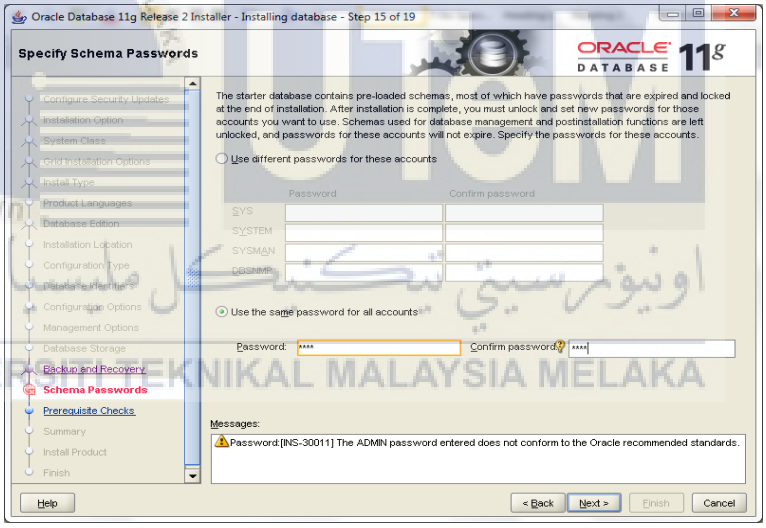
STEP 7: Specify the oracle base path in the directory.



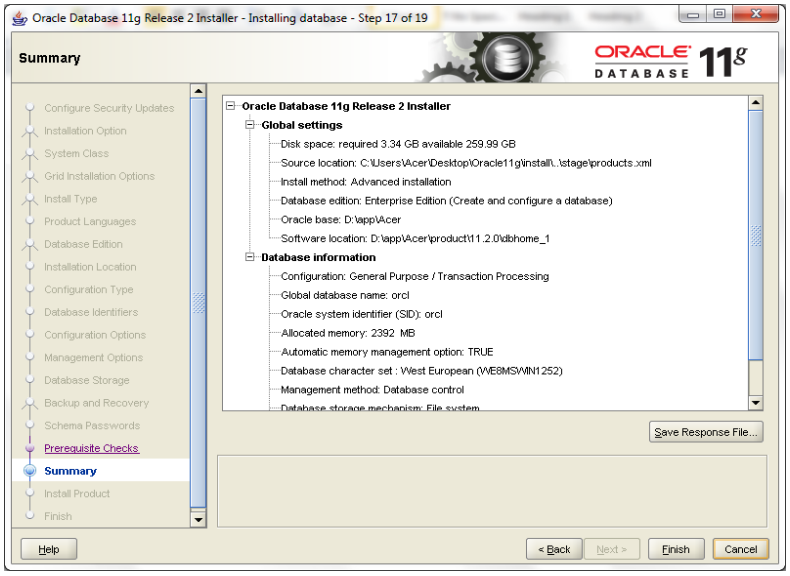
STEP 8: Select the configuration type as general purpose.



STEP 9: Specify the schema password.



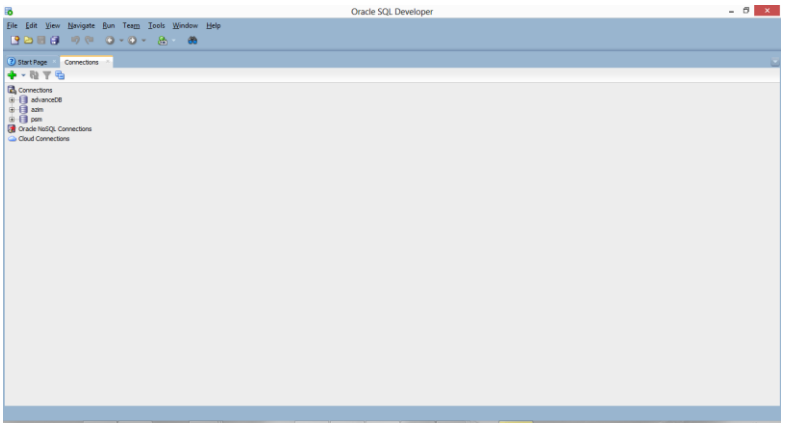
STEP 10: Click finish button and wait the installation finish.



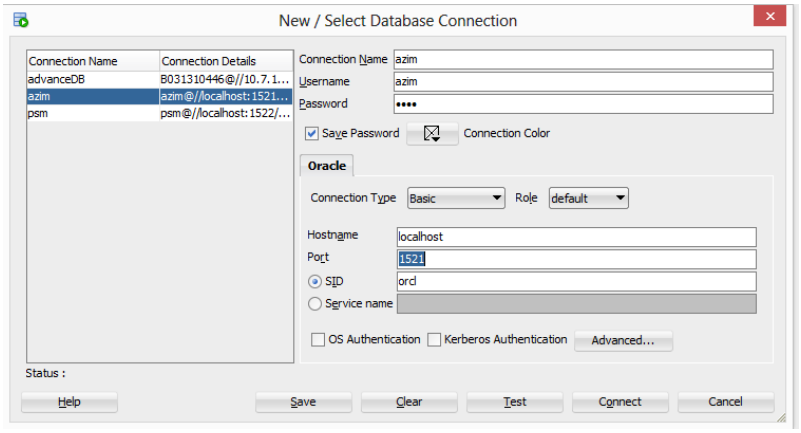
ii. SQL Developer Installation

STEP 1: Run that file of the SQL Developer. It will get the bellow window. Then, it needs to proceed with the following steps.

STEP 2: Click the new connection and fill all the requirement for connect to the Oracle 11g.



STEP 3: Then click test and the status will display success. Click save and click save button to save the connection.



5.3 Database Implementation

In this phase, the Data Definition Language (DDL) Statement and Main Processes of this system will be stated.

5.3.1 DDL/DCL Statements

i. Create Database

```
sqlplus /nolog
conn /as sysdba
create user azim identified by 1234;
```

ii. Data Definition Language (DDL)

Table 5.1: Data Definition Language

TABLES	RESIDENTIAL_COLLEGE
	create table RESIDENTIAL_COLLEGE (residential_No varchar2(15) PRIMARY KEY, residential_Name varchar2(75),

	residential_Address varchar2(250), number_Of_Block number, office_Phone_No varchar2(25));
	HOUSE
	create table HOUSE (house_No varchar2(15) PRIMARY KEY, residential_No REFERENCES RESIDENTIAL_COLLEGE(residential_No), block_Name varchar2(20), block_Type varchar2(20), house_Name varchar2(20), floor_Level varchar2(10), house_Capacity number, house_Occupied number, resident_Availability_Status varchar2(30)
	ROOM
	create table ROOM (room_No varchar2(15) PRIMARY KEY, house_No REFERENCES HOUSE(house_No), room_Name varchar2(20), bed_Position varchar2(20), room_Occupied varchar2(30), rental_Per_Day number);
	FACILITY
	create table FACILITY (facility_No varchar2(15) PRIMARY KEY, facility_Name varchar2(75), facility_Type varchar2(30), house_No REFERENCES HOUSE(house_No), room_No REFERENCES ROOM(room_No)

	);
	RESIDENT
	create table RESIDENT (student_Id varchar2(15) PRIMARY KEY, student_Name varchar2(75), student_Gender varchar2(20), student_Phone_No varchar2(25), student_Email varchar2(150), student_Faculty varchar2(30), student_Year_Study varchar2(10), student_Semester varchar2(30), login_Password varchar2(30));
	STAFF
	create table STAFF (staff_Id varchar2(15) PRIMARY KEY, residential_No REFERENCES RESIDENTIAL_COLLEGE(residential_No), staff_Name varchar2(75), staff_Gender varchar2(20), staff_Phone_No varchar2(25), staff_Email varchar2(150), staff_Role varchar2(30), staff_Position varchar2(30), staff_Status varchar2(30), login_Password varchar2(30));
	PAYMENT
	create table PAYMENT (payment_No varchar2(15) PRIMARY KEY, student_Id REFERENCES RESIDENT(student_Id), payment_Name varchar2(150), total_Rental number(5,2),

	total_Damage_Charges number(5,2), date_Of_Payment date);
	RENTAL
	create table RENTAL (rental_No varchar2(15) PRIMARY KEY, room_No REFERENCES ROOM(room_No), rental_Session varchar2(20), rental_Semester varchar2(20), start_Rental_Date date, check_Out_Date date, total_Days number, rental_status varchar2(50), payment_No REFERENCES PAYMENT(payment_No), student_Id REFERENCES RESIDENT(student_Id));
	DAMAGE_REPORT
	create table DAMAGE_REPORT (damage_Report_No varchar2(15) PRIMARY KEY, facility_No REFERENCES FACILITY(facility_No), damage_Item_Name varchar2(75), damage_Desc varchar2(150), damage_Report_Date date, student_Id REFERENCES RESIDENT(student_Id), staff_Id REFERENCES STAFF(staff_Id), room_No REFERENCES ROOM(room_No));
	REPORT_ACTION
	create table REPORT_ACTION (report_Action_No varchar2(15) PRIMARY KEY, damage_Report_No REFERENCES DAMAGE_REPORT(damage_Report_No),

	repairing_Damage_Date date, damage_Action_Status varchar2(30), damage_Charges number(3,2), charges_Status varchar2(30), payment_No REFERENCES PAYMENT(payment_No));

5.3.2 Main Processes

i. **Stored Procedure**

Table 5.2: Stored Procedure

STORE PROCEDURE	**LOGIN_STAFFVALIDATION**
	CREATE OR REPLACE PROCEDURE login_staffValidation(userID IN varchar2, userPassword IN varchar2, residentialNO OUT STAFF.residential_No%TYPE, staffName OUT STAFF.staff_Name%TYPE, staffGender OUT STAFF.staff_Gender%TYPE, staffPhone OUT STAFF.staff_Phone_No%TYPE, staffEmail OUT STAFF.staff_Email%TYPE, staffRole OUT STAFF.staff_Role%TYPE, staffPosition OUT STAFF.staff_Position%TYPE, staffStatus OUT STAFF.staff_Status%TYPE, v_count OUT number) IS BEGIN

	SELECT residential_No, staff_Name, staff_Gender, staff_Phone_No, staff_Email, staff_Role, staff_Position, staff_Status, COUNT(*) INTO residentialNO, staffName, staffGender, staffPhone, staffEmail, staffRole, staffPosition, staffStatus, v_count FROM STAFF WHERE staff_Id = userID AND login_Password = userPassword GROUP BY residential_No, staff_Name, staff_Gender, staff_Phone_No, staff_Email, staff_Role, staff_Position, staff_Status; COMMIT; END;
	LOGIN_STUDENTVALIDATION
	CREATE OR REPLACE PROCEDURE login_studentValidation(userID IN varchar2, userPassword IN varchar2, studentName OUT RESIDENT.student_Name%TYPE, studentGender OUT RESIDENT.student_Gender%TYPE, studentPhone OUT RESIDENT.student_Phone_No%TYPE, studentEmail OUT RESIDENT.student_Email%TYPE, studentFaculty OUT RESIDENT.student_Faculty%TYPE, studentYearStudy OUT RESIDENT.student_Year_Study%TYPE,

	studentSemester OUT
	RESIDENT.student_Semester%TYPE,
	v_count OUT number
	)
	IS
	BEGIN
	
	SELECT student_Name, student_Gender, student_Phone_No, student_Email, student_Faculty, student_Year_Study, student_Semester, COUNT(*)
	INTO studentName, studentGender, studentPhone, studentEmail, studentFaculty, studentYearStudy, studentSemester, v_count
	FROM RESIDENT
	WHERE student_Id = userID AND login_Password = userPassword
	GROUP BY student_Name, student_Gender, student_Phone_No, student_Email, student_Faculty, student_Year_Study, student_Semester;
	
	COMMIT;
	END;

	STUDENTINSERTREPORT
	CREATE OR REPLACE PROCEDURE studentInsertReport (studentID IN DAMAGE_REPORT.student_Id%TYPE, facilityNO IN FACILITY.facility_No%TYPE, damageItem IN DAMAGE_REPORT.damage_Item_Name%TYPE, damageDesc IN DAMAGE_REPORT.damage_Desc%TYPE, roomNO IN DAMAGE_REPORT.room_No%TYPE

	)
	AS
	BEGIN
	
	INSERT INTO DAMAGE_REPORT (facility_No, damage_Item_Name, damage_Desc, damage_Report_Date, student_Id, staff_Id, room_No)
	VALUES (facilityNO, damageItem, damageDesc, SYSDATE, studentID, '', roomNO);
	
	COMMIT;
	END;

	UPDATEDAMAGEREPORTSTUDENT
	CREATE OR REPLACE PROCEDURE updateDamageReportStudent (damageNO IN DAMAGE_REPORT.damage_Report_No%TYPE, reportNO IN REPORT_ACTION.report_Action_No%TYPE, damageStatus IN REPORT_ACTION.damage_Action_Status%TYPE, charges IN REPORT_ACTION.damage_Charges%TYPE, chargesStatus IN REPORT_ACTION.charges_Status%TYPE) AS BEGIN UPDATE REPORT_ACTION SET repairing_Damage_Date = SYSDATE, damage_Action_Status = damageStatus,

	damage_Charges = charges, charges_Status = chargesStatus WHERE report_Action_No = reportNO; COMMIT; END;
	VIEWSTAFFAVAILABILITY

	CREATE OR REPLACE PROCEDURE viewStaffAvailability(
	availabilityRefcur OUT SYS_REFCURSOR
	)
	AS
	BEGIN
	OPEN availabilityRefcur FOR
	
	SELECT s.staff_Id, staff_Name, staff_Position, COUNT(r.damage_Report_No) JOB
	FROM STAFF s
	LEFT JOIN DAMAGE_REPORT d
	ON s.staff_Id = d.staff_Id
	LEFT JOIN REPORT_ACTION r
	ON d.damage_Report_No = r.damage_Report_No
	AND r.damage_Action_Status = 'IN PROCESS'
	WHERE staff_Position = 'Technician'
	AND staff_Status = 'Active'
	GROUP BY s.staff_Id, staff_Name, staff_Position
	ORDER BY JOB ASC;
	
	COMMIT;
	END;

	VIEWSTAFFTOTALJOB
	CREATE OR REPLACE PROCEDURE viewStaffTotalJob(

```
viewStaffTotalJobRef OUT SYS_REFCURSOR
)
AS
BEGIN
OPEN viewStaffTotalJobRef FOR

SELECT d.staff_Id, staff_Name, staff_Position,
COUNT(r.damage_Report_No) JOB
FROM STAFF s
LEFT JOIN DAMAGE_REPORT d
ON s.staff_Id = d.staff_Id
LEFT JOIN REPORT_ACTION r
ON d.damage_Report_No = r.damage_Report_No
AND r.damage_Action_Status = 'IN PROCESS'
WHERE staff_Position = 'Technician'
AND staff_Status = 'Active'
GROUP BY d.staff_Id, staff_Name, staff_Position
ORDER BY JOB ASC;

COMMIT;
END;
```

CALCUALLCHARGESSTUDENT

```
CREATE OR REPLACE PROCEDURE
calcuAllChargesStudent (
studentID IN PAYMENT.student_Id%TYPE,
calcuCharges OUT
PAYMENT.total_Damage_Charges%TYPE
)
AS
BEGIN

SELECT SUM(damage_Charges)
INTO calcuCharges
```

	FROM DAMAGE_REPORT d, REPORT_ACTION r WHERE r.damage_Report_No = d.damage_Report_No AND charges_Status = 'STILL NOT PAY' AND student_Id = studentID; COMMIT; END;
	DISPLAYSTUDENTPAYMENT
	CREATE OR REPLACE PROCEDURE displayStudentPayment(viewStudentPayment OUT SYS_REFCURSOR) AS BEGIN OPEN viewStudentPayment FOR SELECT p.student_Id, r.payment_no, student_Name, payment_Name, total_Damage_Charges, date_Of_Payment, COUNT(r.payment_no) totalFacility FROM PAYMENT p, REPORT_ACTION r, DAMAGE_REPORT d, RESIDENT s WHERE p.payment_no = r.payment_no AND d.damage_Report_No = r.damage_Report_No AND s.student_Id = d.student_Id AND s.student_Id = p.student_Id GROUP BY p.student_Id, r.payment_no, student_Name, payment_Name, total_Damage_Charges, date_Of_Payment HAVING COUNT (r.payment_no) >= 1; COMMIT;

	END;

ii. **Trigger**

Table 5.3: Trigger

TRIGGER	BEFORE INSERT

	create or replace trigger pk_facility_trig
	before insert on facility
	for each row
	begin
	if (:new.facility_type = 'furniture') then
	select 'ff' || facility_seq_1.nextval into
	:new.facility_no from dual;
	elsif (:new.facility_type = 'electrical') then
	select 'ef' || facility_seq_2.nextval into
	:new.facility_no from dual;
	elsif (:new.facility_type = 'other') then
	select 'of' || facility_seq_3.nextval into
	:new.facility_no from dual;
	else
	raise_application_error (-20404, 'sorry not in data or already exist');
	end if;
	end;

	create or replace trigger pk_payment_trig
	before insert on payment
	for each row
	begin
	select 'py' || payment_seq.nextval into
	:new.payment_no from dual;
	end;

	create or replace trigger pk_rental_trig_pk before insert on rental for each row

	begin
	select 'sr' || rental_seq.nextval into :new.rental_no
	from dual;
	end;

	create or replace trigger pk_report_trig_pk
	before insert on damage_report
	for each row
	begin
	select 'rd' || damage_report_seq.nextval into
	:new.damage_report_no from dual;
	end;

	create or replace trigger pk_action_trig_pk
	before insert on report_action
	for each row
	begin
	select 'ra' || report_action_seq.nextval into
	:new.report_action_no from dual;
	end;

	BEFORE UPDATE

	create or replace trigger
	trigbefore_studentchangeroom
	before update on rental
	for each row
	begin
	if updating('room_no') then
	update room
	set room_occupied = 'in process repairing'
	where room_no = :old.room_no;
	end if;
	end;

	BEFORE DELETE
	create or replace trigger trigbefore_deletestudent before delete on resident

	for each row
	declare
	d_rentalstatus varchar2(50);
	d_chargesstatus varchar2(50);
	begin
	update damage_report
	set student_id = null
	where student_id = :old.student_id;
	
	update payment
	set student_id = null
	where student_id = :old.student_id;
	
	delete from rental
	where student_id = :old.student_id;
	end;

	AFTER INSERT

	create or replace trigger trig_studentinsertreport
	after insert on damage_report
	for each row
	begin
	insert into report_action(damage_report_no,
	repairing_damage_date, damage_action_status,
	damage_charges, charges_status, payment_no)
	values (:new.damage_report_no, '', 'validation', '', '', '');
	end;

	AFTER UPDATE
	create or replace trigger house_occupied_trig after update on room for each row declare v_capacity number; v_occupied number;

```
d_capacity number;
d_occupied number;
begin
if updating then
  if (:new.room_occupied = 'not available') then

    select house_capacity, house_occupied
    into v_capacity, v_occupied
    from house
    where house_no = :new.house_no;

    v_occupied := v_occupied + 1;
    update house
    set house_occupied = v_occupied
    where house_no = :old.house_no;

    if (v_occupied = v_capacity) then
      update house
      set resident_availability_status = 'not available'
      where house_no = :new.house_no;
    end if;

  elsif (:new.room_occupied = 'available') then

    select house_capacity, house_occupied
    into d_capacity, d_occupied
    from house
    where house_no = :new.house_no;

    d_occupied := d_occupied - 1;
    update house
    set house_occupied = d_occupied
    where house_no = :old.house_no;
```

```
    if (d_occupied < d_capacity) then
      update house
      set resident_availability_status = 'available'
      where house_no = :new.house_no;
    end if;
  end if;
end if;
end;
```

```
create or replace trigger damage_actionstatus_trig
after update on report_action
for each row
declare
d_roomstatus varchar2(30);
v_roomno varchar2(15);
begin
if updating('damage_action_status') then
 if :new.damage_action_status = 'complete' then

   select d.room_no, room_occupied
   into v_roomno, d_roomstatus
   from room m, damage_report d
   where m.room_no = d.room_no
   and damage_report_no = :new.damage_report_no;

   if (d_roomstatus = 'in process repairing') then
     update room
     set room_occupied = 'available'
     where room_no = v_roomno;
   end if;
  end if;
end if;
end;
```

	AFTER INSERT OR UPDATE OR DELETE

	create or replace trigger room_occupied_trig
	after insert or update or delete on rental
	for each row
	begin
	if inserting then
	update room
	set room_occupied = 'not available'
	where room_no = :new.room_no;
	
	elsif updating('room_no') then
	update room
	set room_occupied = 'not available'
	where room_no = :new.room_no;
	
	elsif deleting then
	update room
	set room_occupied = 'available'
	where room_no = :old.room_no;
	end if;
	end;

5.4 Conclusion

This chapter explains about the implementation of the system and it covers about the software development environment setup, software configuration management, database implementation and status implementation. All these can give developer clear vision about their system in order to fulfil system requirement to meets user satisfaction while using it.

CHAPTER VI

TESTING

6.1 Introduction

In this chapter, E-Report Management System will be elaborate the testing phase and the result have been carried out on this system. The purpose of this system testing is to determine the capability of the system either it meet all the requirements and how effective this system to be use by user. It also produces different types, test techniques and test result. Testing phase performed to test the functionality of the system. This is to ensure that the system has been fully developed and functioning. Furthermore, the testing process also ensures that all components and interfaces in the system working without any error that occurs.

Other than that, this chapter also discussing about the test plan including organization, test environment and test schedule. The test strategy will cover on classes of testing. The testing phase also involved in test description and test data. The system is tested to ensure all the components including functional and non-functional requirements work properly. The testing process performed in stages in order to identify error and defect and to make sure each components meet required system specification. At the end, based on the test result and analysis, it will determine either the system are failed or success and either it is meet the user satisfaction when using this system.

6.2 Test Plan

Test plan is used in the beginning of testing phase which involves organization testing and environment testing such as alpha testing and beta testing. Test content is the set of test inputs, the environment during implementation, and the expected results for specific objectives consists of test organization, test environment and test schedule.

6.2.1 Test Organization

Test organization defines who is responsible to test the system through testing process. This tester are come from different background that must have a lot of knowledge. The system is develop using the same medium and components to test the whole system and facilitate designers to produce procedures and functions for E-Report Management System. The users involved are:

i. **System Developer**

Responsibilities: System developer involved in the testing system, identify errors and documentation of the results of test content. System Developer ensures that the system will run smoothly based on requirement before delivered to the end user.

ii. **Administrator**

Responsibilities: Admin is the main user. Admin responsible to update the report of damage. Monitoring overall system performance. Test the system module and give their feedback. Their feedback can be as guide to enhance the system.

iii. **Student**

Responsibilities: User need to insert the required information in the system and view some information. Test the system module and give their feedback. Their feedback can be as guide to enhance the system.

6.2.2 Test Environment

A testing environment is a setup of software and hardware on which testing team is going to perform the testing on E-Report Management System. It is comprised of all the conditions, circumstances and influences surrounding and affecting the testing of software. The environment includes the organization's policies, test processes, test tools, method for developing and improving test processes as well as any test labs developed for the purpose of testing software and multiple operating environments

6.2.2.1 Environment Setup

Environment setup is to configure and manage the platform for the E-Report Management System to ensure the system can run successfully. Table 6.1 shows the application workspace for E-Report Management System.

Table 6.1: Environment Setup Specification

Environment Specification	Description
Operating System	Windows 8.1
Processor	Intel Core i5
Random Access Memory (RAM)	4 GB and above
Database	Oracle 11g Enterprise Edition
Server	WAMP Server 2.4
Server-Scripting	PHP

6.2.2.2 Software Application

Software application describe about all the contents or application inside the E-Report Management System. Description below shows all the application which is applicable in this system.

i. System Login
ii. Add, update, delete, and search in each form for administrator, and student.
iii. Report and analysis
iv. Print the report

6.2.2.3 System Software

System software consists of tools that have been implemented in the E-Report Management System. Below shows all the software that involved in this system development.

i. Windows 8.1 Premium
ii. WAMP Server 2.4
iii. Oracle 11g Enterprise Edition
iv. SQL Developer
v. Google Chrome (web browser)
vi. Sublime & Notepad++

6.2.2.4 System Hardware

System hardware is the hardware that involves in this system development. Below shows the related hardware for development of E-Report Management System.

i. Personal computer with hard disk, RAM, processor, monitor, keyboard installed on it
ii. Printer

6.2.3 Test Schedule

The test schedule purpose is to make sure that all the testing activities by whom has been already performed. The schedule will give a guide to the developer to do the testing on certain time accurately along the duration of project development. Table 6.2

shows the test schedule for staff as system developer of the system, Table 6.3 shows the test schedule for admin and Table 6.4 shows the test schedule for student.

Table 6.2: Test Schedule for System Developer

Module	Activity	Duration	Start Date	End Date
Creating and Searching the Data in the Database	Test unit integration, testing and user acceptance	2 days	26/7/2016	27/7/2016
Design Interface	Test unit integration, testing and user acceptance	3 days	26/7/2016	28/7/2016
System Forms	Test unit integration, testing and user acceptance	3 days	26/7/2016	28/7/2016
System Login	Test unit integration, testing and user acceptance	1 days	26/7/2016	26/7/2016
System View Process	Test unit integration, testing and user acceptance	3 days	26/7/2016	28/7/2016
Generate Report and graph	Test unit integration, testing and user acceptance	3 days	26/7/2016	28/7/2016
Error Handling	Test unit integration, testing and user acceptance	2 days	26/7/2016	27/7/2016

Table 6.3: Test Schedule for Admin

Module	Activity	Duration	Start Date	End Date
System Login	Test unit integration, testing and user acceptance	1 day	26/7/2016	26/7/2016

Management	Test unit integration, testing and user acceptance	2 days	26/7/2016	27/7/2016
Report	Test unit integration, testing and user acceptance	2 days	26/7/2016	27/7/2016

Table 6.4: Test Schedule for Student

Module	Activity	Duration	Start Date	End Date
System Login	Test unit integration, testing and user acceptance	1 day	8/8/2016	8/8/2016
Management	Test unit integration, testing and user acceptance	2 days	8/8/2016	9/8/2016
Report	Test unit integration, testing and user acceptance	2 days	8/8/2016	9/8/2016

6.3 Test Strategy

Test strategy is a set of guidelines which describe the test design. The purpose of this testing strategy is to achieve the objectives of the parties involved. The process involved manage the expectations from start until the developer clear to continue the next steps through the system development. There are two types of testing methods:

i. Alpha Testing

This methodology will be covered the subsystem of an application which will be done by the developers themselves. The aim is to improve software product and find bugs by see the situation inside the application. It is suitable which need more

focuses on the process of application inside the system development. The testing is done by using the false data to see the system running smoothly.

ii. Beta Testing

Beta testing comes after alpha testing and can be considered a form of external user acceptance testing. Versions of the software, known as beta versions, are released to a limited audience outside of the programming team. The software is released to groups of people so that further testing can ensure the product has few faults or bugs. Sometimes, beta versions are made available to the open public to increase the feedback field to a maximal number of future users.

6.3.1 Classes of tests

Classes of test are divided into security testing, error handling, output correctness test and user acceptance test.

i. Security Test

In this modern world, security problems greatly feared happened. Therefore, this security test committed. The purpose is to make sure the security of the system becomes stronger and to ensure that the data contained in the system is safe. The weak protection may lead to the security violations and become vulnerability thread to the system. For example, the security of data in the system will be controlled with the password.

ii. Error Handling Test

The purpose of error handling test is to ensure that E-Report Management System only accept right input from the user. The error message will appear for any wrong input that user entered to ensure the user know what input they have entered wrong or unfilled required value.

iii. User Acceptance Test

The purpose is to ensure the system is user friendly to the user of the E-Report Management System. The graphical user interface (GUI) must be comfortable to the user since there will be variety of IT knowledge level among them.

6.4 Test Design

There are two things of test design which are test description and test data. Test description covers the activities that required and it is will be documented for identifying the best data process. It will describe the test cases and expected result, meanwhile for test data it will covers about user acceptance.

6.4.1 Test Description

Test description is the result of a documented output to identify the results that are expected. A test case is a documented set of data input and operating condition required to run a test item for the expected result for each system modules.

Table 6.5: Test Description for Login Module

Description	Testing Type	Expected Result	Respondent Comment
Invalid User ID/ Invalid Password	Unit Testing/Integration	'SORRY! Your User ID or Password is Wrong' message will appear.	Message 'SORRY! Your User ID or Password is Wrong' will appear when the user enters a user name or password is incorrect.
User ID blank/ Password blank	Unit Testing/Integration	'Please fill out this field' message will appear.	Message 'Please fill out this field' will appear when the user do not field either field form.
Valid User ID and Password	Unit Testing/Integration	User can logon successfully	Success to login. Admin will access to admin mainpage, while student will access to student mainpage

Table 6.6: Test Description for Student Registration Module

Description	Testing Type	Expected Result	Respondent Comment
Invalid Student ID	Unit Testing/Integration	'Student ID Already Exist' message will appear.	Message 'Student ID Already Exist' will appear when there are blank fields.
Field Form blank	Unit Testing/Integration	'Please fill out this field' message will appear.	Message 'Please fill out this field' will appear when there are blank fields.
Valid input for each fields	Unit Testing/Integration	Registration data information successfully stored in the database.	Message 'OK' will appear and Registration process succeed.

Table 6.7: Test Description for Staff Registration Module

Description	Testing Type	Expected Result	Respondent Comment
Invalid Staff ID	Unit Testing/Integration	'Staff ID Already Exist' message will appear.	Message 'Staff ID Already Exist' will appear when there are blank fields.
Field Form blank	Unit Testing/Integration	'Please fill out this field' message will appear.	Message 'Please fill out this field' will appear when there are blank fields.
Valid input for each fields	Unit Testing/Integration	Registration data information successfully stored in the database.	Message 'OK' will appear and Registration process succeed.

Table 6.8: Test Description for Delete Student Module

Description	Testing Type	Expected Result	Respondent Comment
Invalid input	Unit Testing/Integration	Student cannot be because still renting or still have damage charges.	Message 'These student Still Renting' or 'These student Still not Pay Damage Charges' will appear when there are blank fields.

Valid input	Unit Testing/Integration	Deleting student successfully deleted in the database.	Deleting process succeed.

6.4.2 Test Data

Test data is where the data should be included to test most of the features. The data included functional to get the expected results. Please refer to **APPENDIX B** to see **Figure** of test data for E-Report Management System.

Table 6.9: Test Data for Login Module

Test	1	2	3
Username	S001	S001	S001
Password	1234		123
Result Test Data	Login failed. User Password not match with database.	Login failed. User have to require to fill in password field.	User login successfully.

Table 6.10: Test Data for Student Registration Module

Test	1	2	3
Student Matrik No	B031310478		B031310999
Name	Mohamad Faizal	Mohamad Faizal	Mohamad Faizal
Gender	Men	Men	Men
Phone Number	014-5163813	014-5163813	014-5163813
Faculty	FTMK	FTMK	FTMK
Year of Study	3	3	3
Semester	2	2	2
Result Test Data	Registration failed. User ID already exist.	Registration failed. User have to require to fill in Student	Registration successfully stored in database.

		Matrik No field blank.	

Table 6.11: Test Data for Staff Registration Module

Test	1	2	3
Staff ID	S001		S010
Name	Aiman Hamka	Aiman Hamka	Aiman Hamka
Gender	Men	Men	Men
Phone Number	012-2367487	012-2367487	012-2367487
Role	Staff	Staff	Staff
Position	Technician	Technician	Technician
Residential College	Lestari	Lestari	Lestari
Result Test Data	Registration failed. User ID already exist.	Registration failed. User have to require to fill in Staff ID field blank.	Registration successfully stored in database.

Table 6.12: Test Data for Delete Student Module

Test	1	2	3
Student Matrik No	B031310478	B001	B002
Name	Hazarul Hazrin	Bob	Syem
Rental Status	Still Renting	Check Out	Check Out
Charges Status		Still not Pay	Already Paid
Result Test Data	Deleting failed. Student still renting.	Deleting failed. Student still not paid charges of damage	Deleting successfully deleted in database.

6.5 Test Results and Analysis

All the test result is documented describe in the table as show below. Test case is the input to test the system. After you run tests, you can review your test results to see which

tests successes and which test are failure. The success or failure when using the actual data for testing process can be the factor to measure the system either it can worked efficiently or need to be fixed. Please refer table below to see the test result and analysis.

Table 6.13: Test Result for Analysis of Login Module

Module Component: Login		**Result**		
Test Case ID	**Test Data ID**	**Description**	**PASS**	**FAIL**
TC_01-1	TD_01-1	User ID and password didn't exist	✓	
TC_01-2	TD_01-2	User ID or password field blank	✓	
TC_01-3	TD_01-3	Valid user ID and password	✓	

Table 6.14: Test Result for Analysis of Registration User Module

Module Component: Staff		**Result**		
Test Case ID	**Test Data ID**	**Description**	**PASS**	**FAIL**
TC_02-1	TD_02-1	All field blank.	✓	
TC_02-2	TD_02-2	Invalid data or format in staff fields.	✓	
TC_02-3	TD_02-3	Valid data for each field.	✓	

Table 6.15: Test Result for Analysis of Deleting User Module

Module Component: Staff		**Result**		
Test Case ID	**Test Data ID**	**Description**	**PASS**	**FAIL**
TC_03-1	TD_03-1	Invalid data or format in staff fields.	✓	
TC_03-2	TD_03-2	Valid data for each field.	✓	

6.6 Conclusion

As the conclusion, these chapters that conduct the testing process are the most crucial part to be completed and develop for this system. This is because, developer need to test every single part of the system to know how the system will perform from several aspects. Many aspect need to be consider such as reliability, security of the system and the user efficiency to make sure that the system meets the entire requirement. From the testing process that have been done, developer can fixed any fault and problems that come up. Next chapter will cover on the conclusion of the overall system. This chapter will explain about the strength and weaknesses of the system, proposition for improve the system and what is the contribution for the future.

CHAPTER VII

CONCLUSION

7.1 Introduction

This chapter will discuss about the strengths and weaknesses in this project based on the observation and some test that had been made. This weaknesses can be refer if there any person who want to upgrade the system to be better in the future. Other than that is proposition to improve this system to be high level system in order can be used for the long period and effective. The outsider suggestion also take into consideration. This is because, all that suggestion are taken as the user view through the system and what user want when they use the system.

For the recommendation, efforts must be applied through all phases of development accordingly. The survey was conducted in connection with the system development in order to avoid confusion over the management and users that use the system. Finally is the contribution of this project to the university or individual that will be used this system either it bring a lot of goodness or badness.

7.2 Observation on Weaknesses and Strengths

The observation of the system's strengths and weaknesses can be identified deficiencies and advantages of the E-Report Management System to make any reforms to the existing system at the present time.

Table 7.1: Weaknesses and Strengths E-Report Management System

Weaknesses	Strengths
a. The status of the report for damage of facility are too complicated unclear the used of it.	a) Admin can assign job and analyse the performance of the staff (Technician).
b. Admin cannot updated the status rental for the student.	b) Student can pay one per charges or pay all of it at once.

7.3 Propositions for Improvement

Upon completing the testing phase and reviewing the system, several things should be considered for improving which can be strengthened the system desirability and functionality.

Firstly, done some improvement on the interface of the system make more user friendly and user can understand the flow without many guidance. Secondly, implement the import, and export to manage the record of the data more effectively. Next, add the rental functionality such as can be update the status of rental and calculation for renting the room when student check out of the room. Lastly, implement backup and recovery for the record data because to ease the searching all old record data and able to recovery the data if there any unexpected matter.

7.4 Project Contribution

As a contribution to the management system, E-Report Management System can be additional system which has an ability to manage and identify the damage of the facility in Lestari residential college. The system hopefully can ease record the report for damage

of facility and decrease the time for student to make the report. This system also capable to calculate the charges damage that student have to pay and generate the report.

7.5 Conclusion

Overall, the developed system has achieved its main objectives and scope proposed by the developers of the system. The weaknesses occur due some constraints and defect during development phase. The improvement will take more extra effort in order to ensure the functionality of system. However, the system already fulfilled the requirement and get the good feedback from users.

REFERENCES

Nadiah Ahmad (2014). “Merit Promotion System” Univeristi Teknikal Malaysia Melaka; Fakulti Teknologi Maklumat dan Komunikasi.

Installing Oracle Database 11g on Windows
http://www.oracle.com/webfolder/technetwork/tutorials/obe/db/11g/r1/prod/install/dbinst/windbinst2.htm

How to Create Report Generation using PHP Codes
http://www.webdeveloper.com/forum/showthread.php?237984-How-to-create-report-generation-using-php-codes

How to do Refcursor for Oracle Database
https://oracle-base.com/articles/misc/using-ref-cursors-to-return-recordsets

Call a Trigger in Procedure: Create Procedure << Function Procedure Packages << Oracle PL/SQL Tutorial
http://www.java2s.com/Tutorial/Oracle/0540__Function-Procedure-Packages/Callatriggerinprocedure.htm

How to do Dynamic Option List
http://www.plus2net.com/php_tutorial/php_drop_down_list.php

Ajax Tutorial for PHP Codes
http://www.w3schools.com/ajax/

Handling Checkbox in a PHP Form Processor
http://www.html-form-guide.com/php-form/php-form-checkbox.html

APPENDICES

APPENDIX A

GRAPHICAL USER INTERFACE (GUI)

Figure 4.12: GUI for Index of E-Report Management System

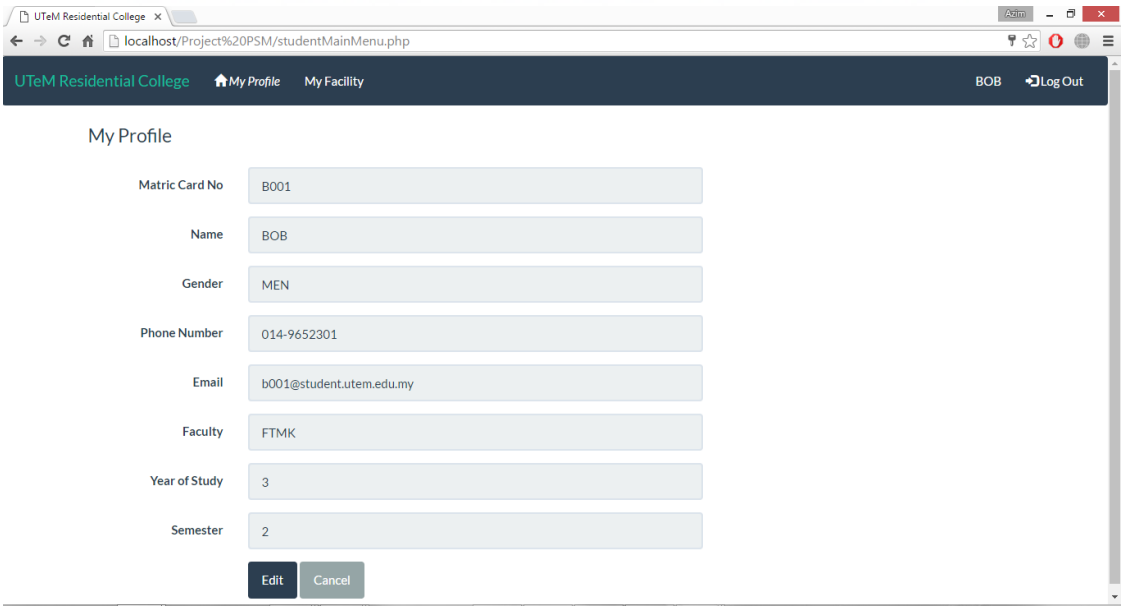


Figure 4.13: GUI for Student Main Index

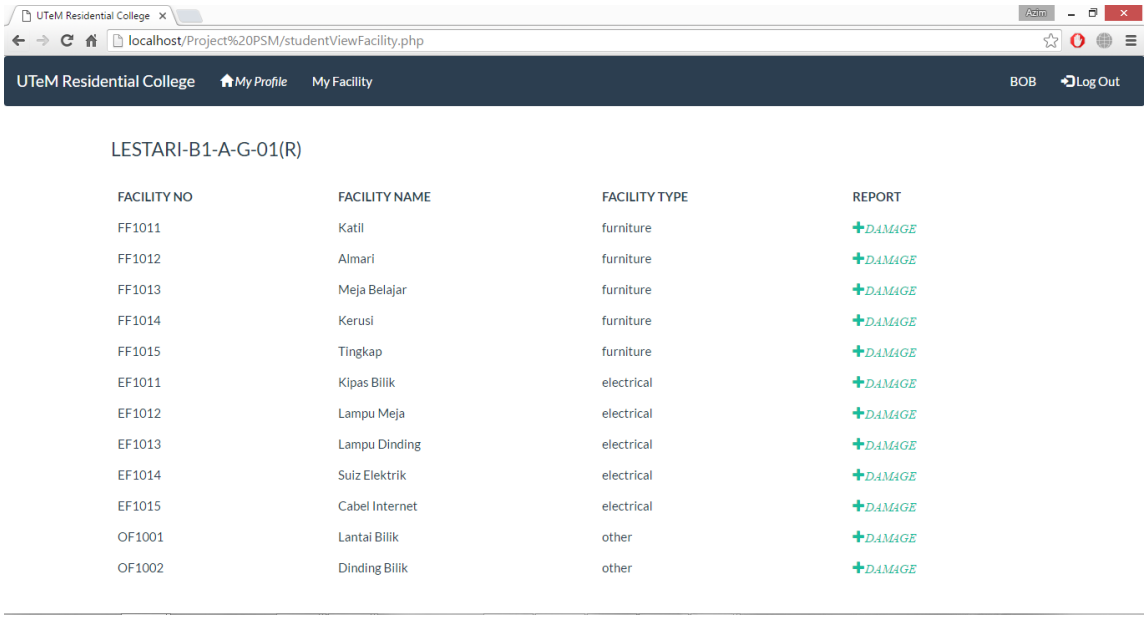


Figure 4.14: GUI for View of Student Room Facility

Figure 4.15: GUI for Admin Main Index

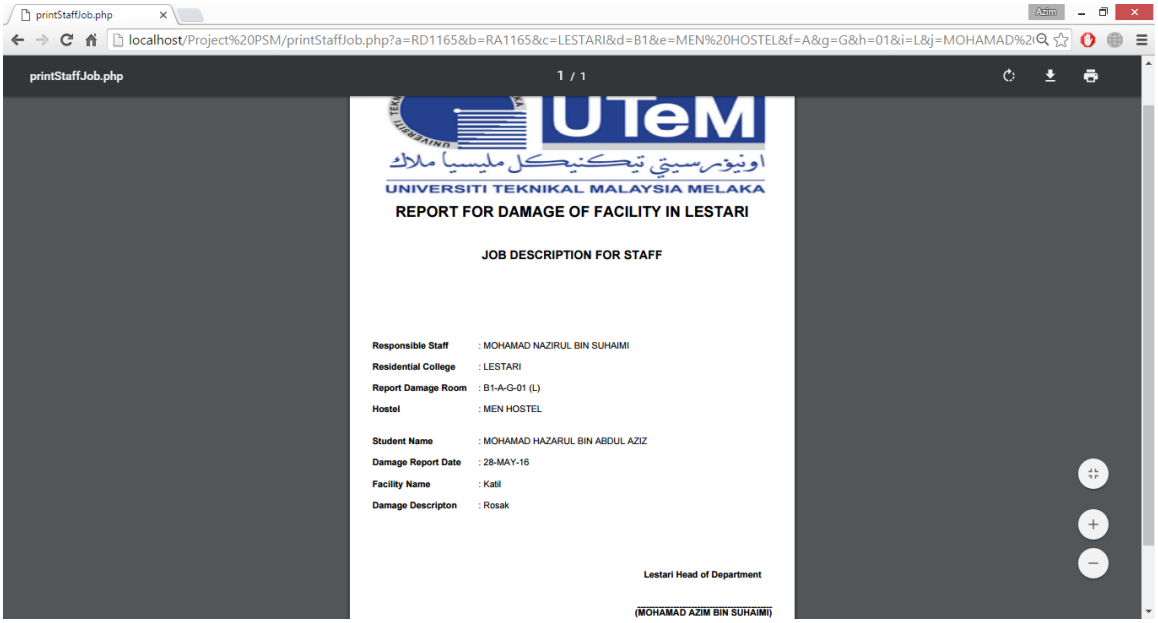


Figure 4.16: GUI for Print Job Description of Staff (Technicians)

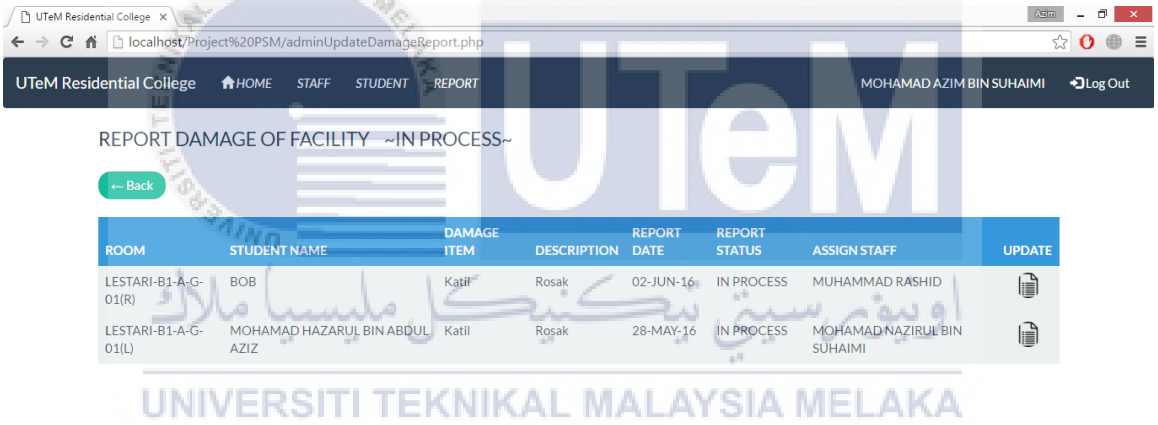


Figure 4.17: GUI for Display in Process Damage Status

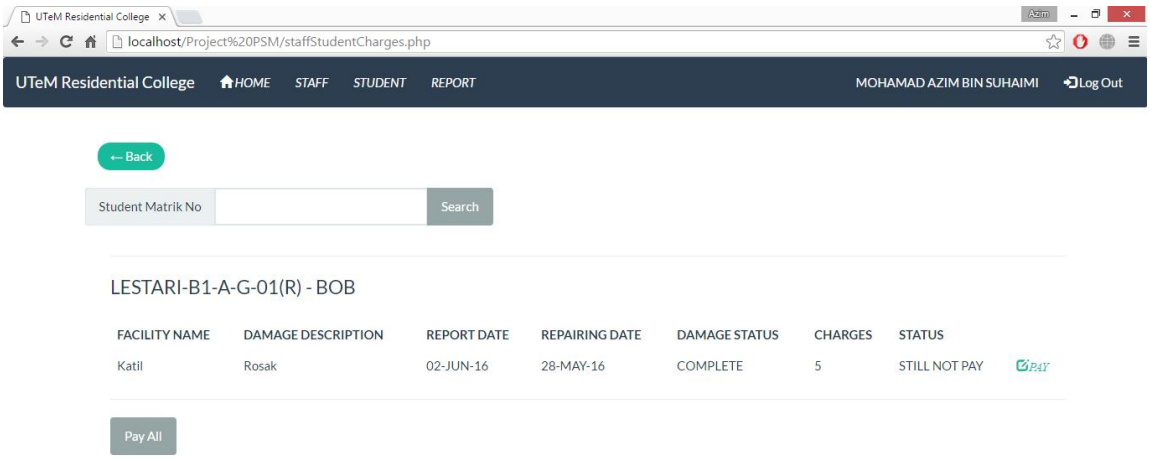


Figure 4.18: GUI for Display Student Charges "Still not paid"

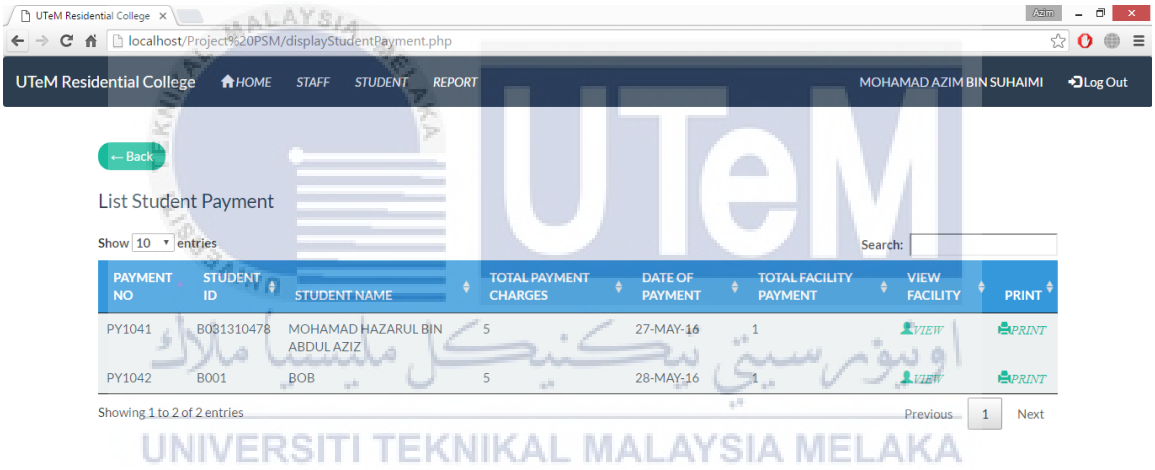


Figure 4.19: GUI for View Student That "Already Paid" Charges

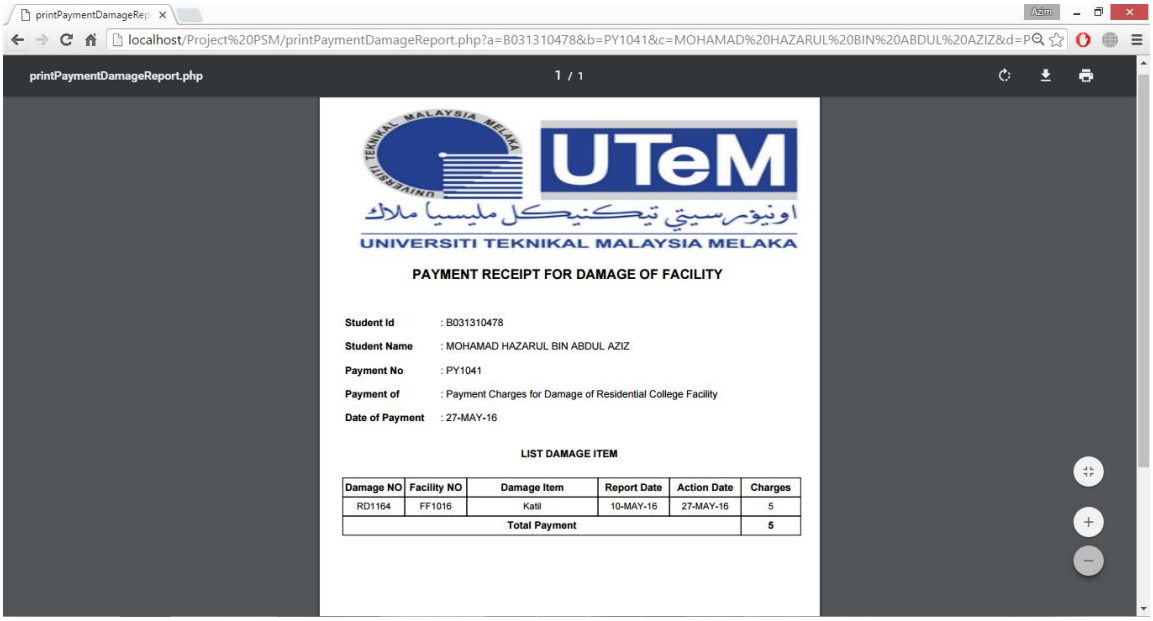


Figure 4.20: GUI for Print Student Payment Charges Receipt

Figure 4.21: GUI for Add Student Form

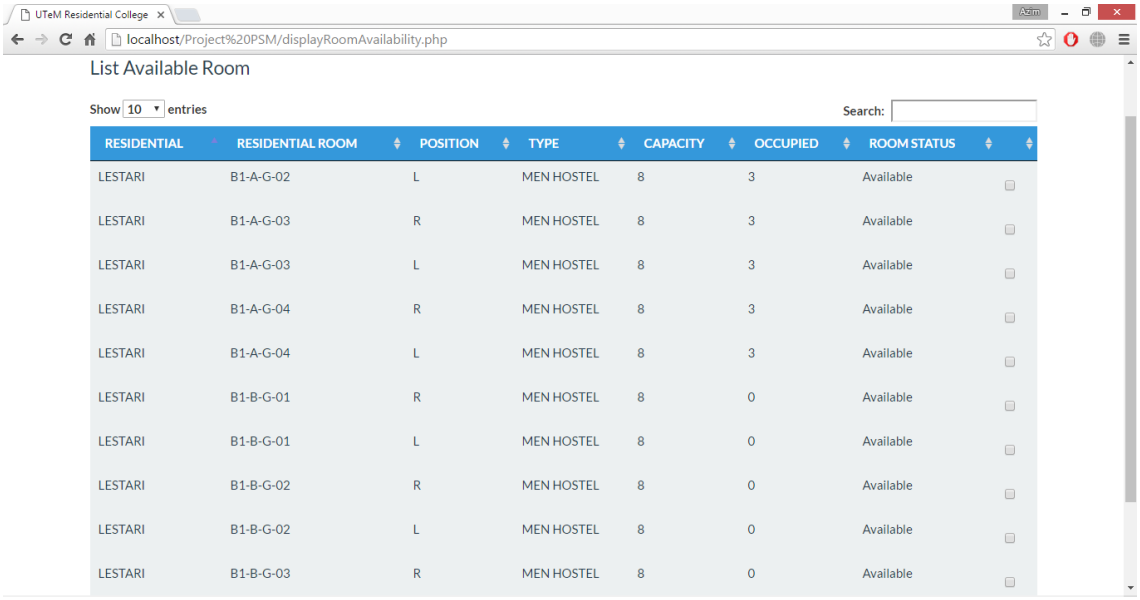


Figure 4.22: GUI for Display Room Availability According to Student Gender

Figure 4.23: GUI for List before Delete Student Form

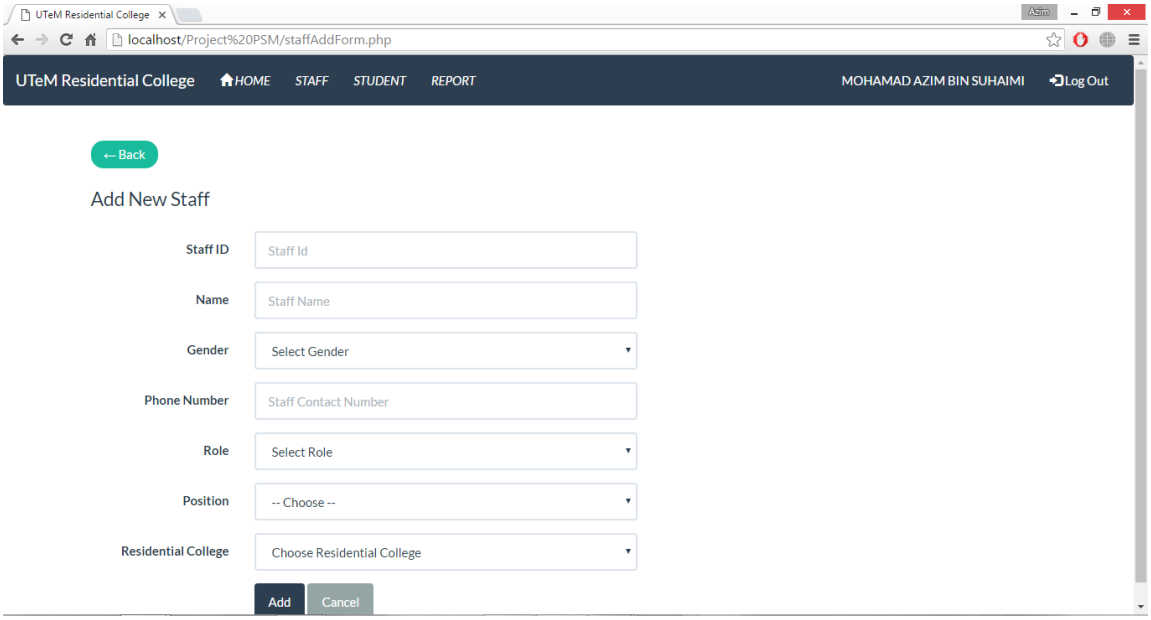


Figure 4.24: GUI for Add Staff Form

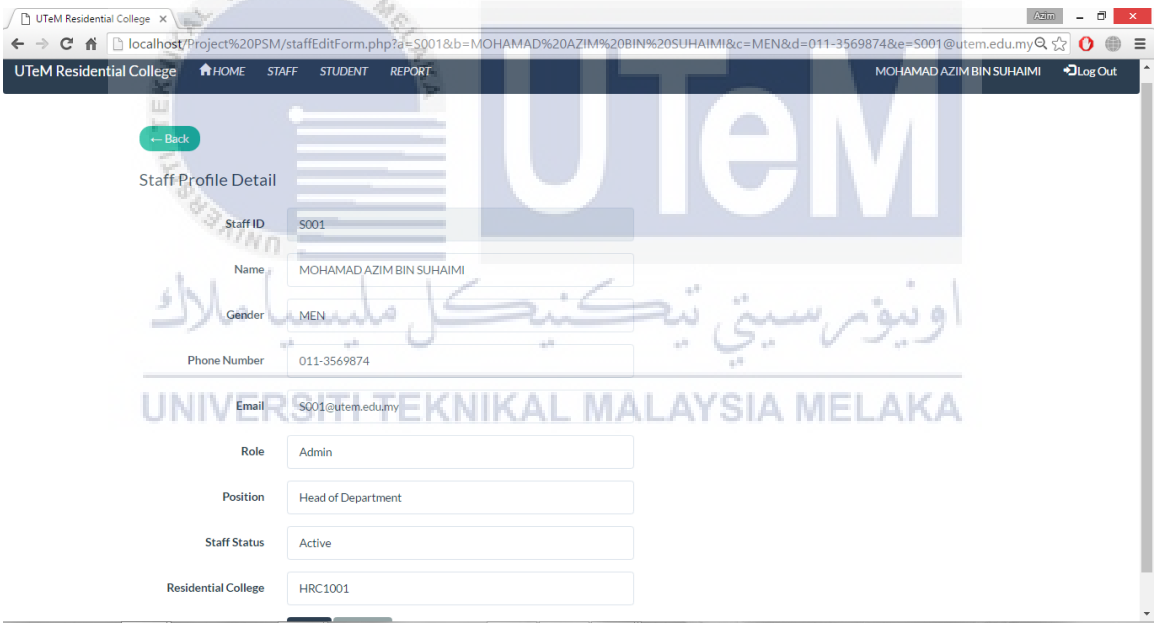


Figure 4.25: GUI for Edit Staff Form

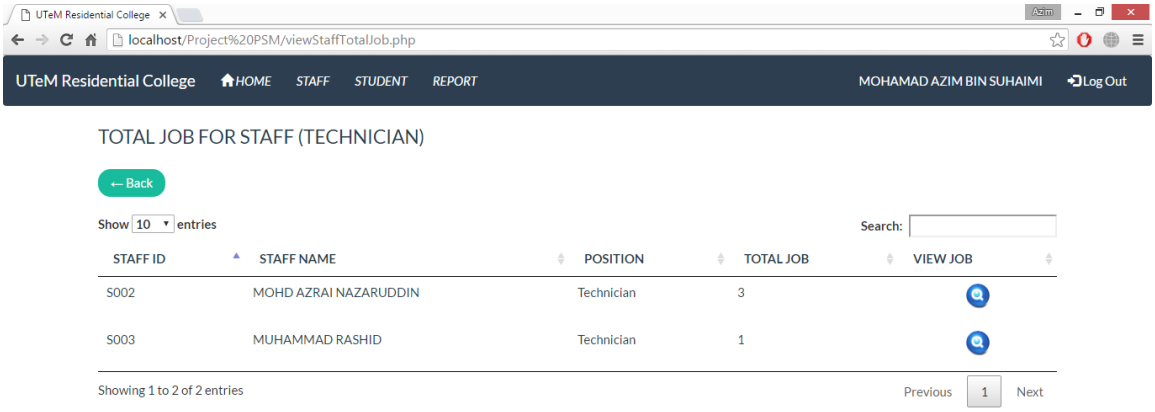


Figure 4.26: GUI for View Staff Total Job

APPENDIX B

TEST DATA FIGURE

1. Test Data for Login Module

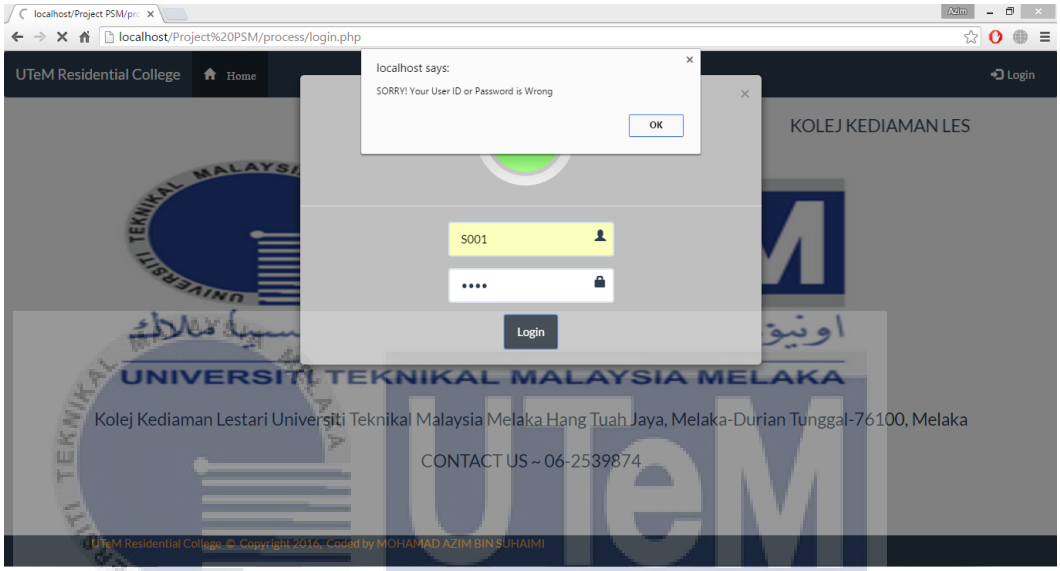


Figure 6.1: Wrong User ID or Password

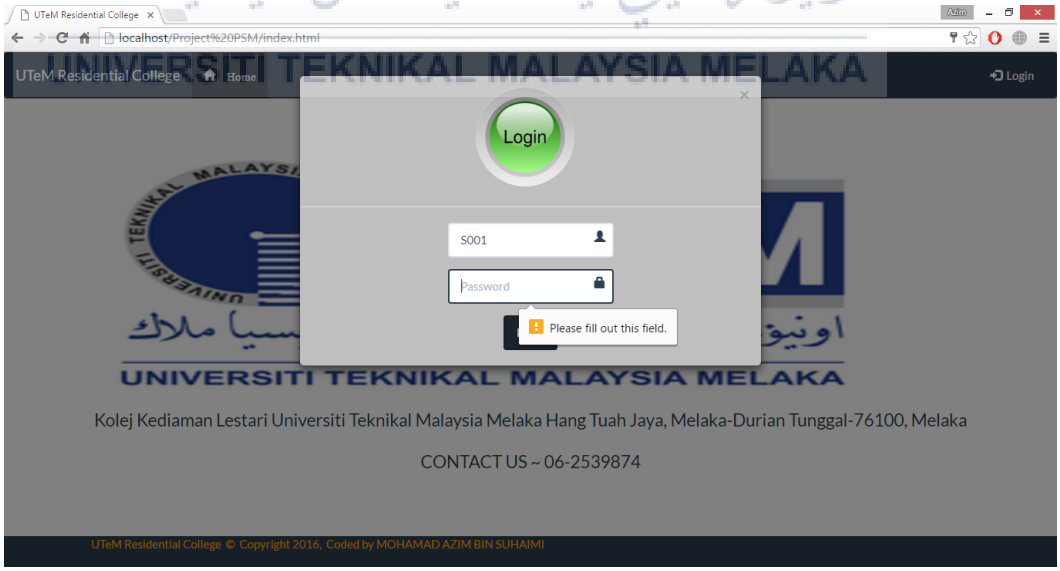


Figure 6.2: Blank Field Either User ID or Password

2. Test Data for Student Registration Module

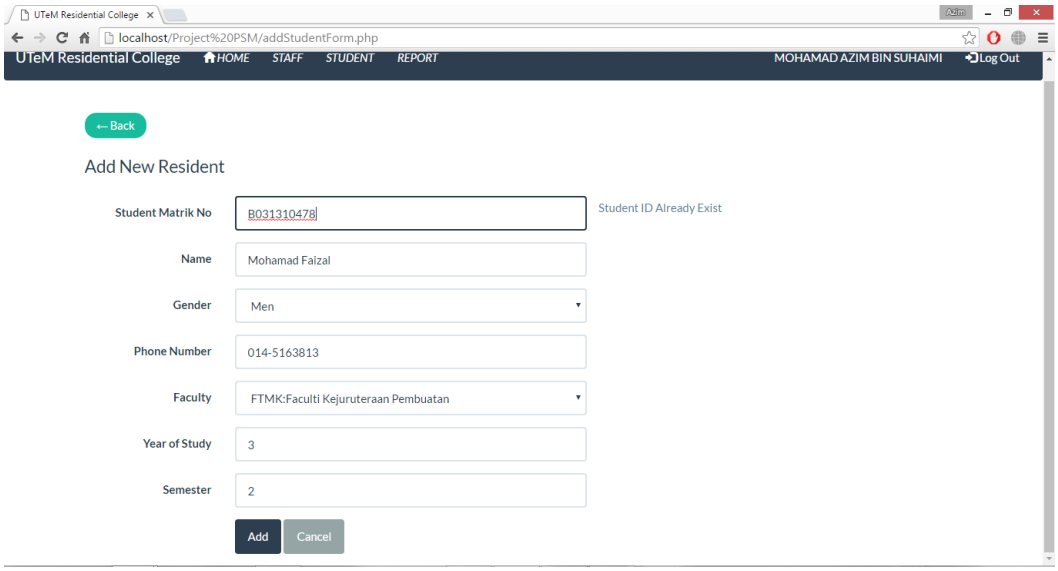


Figure 6.3: Student ID Already Exist

Figure 6.4: Either Field Blank for Student Registration Form

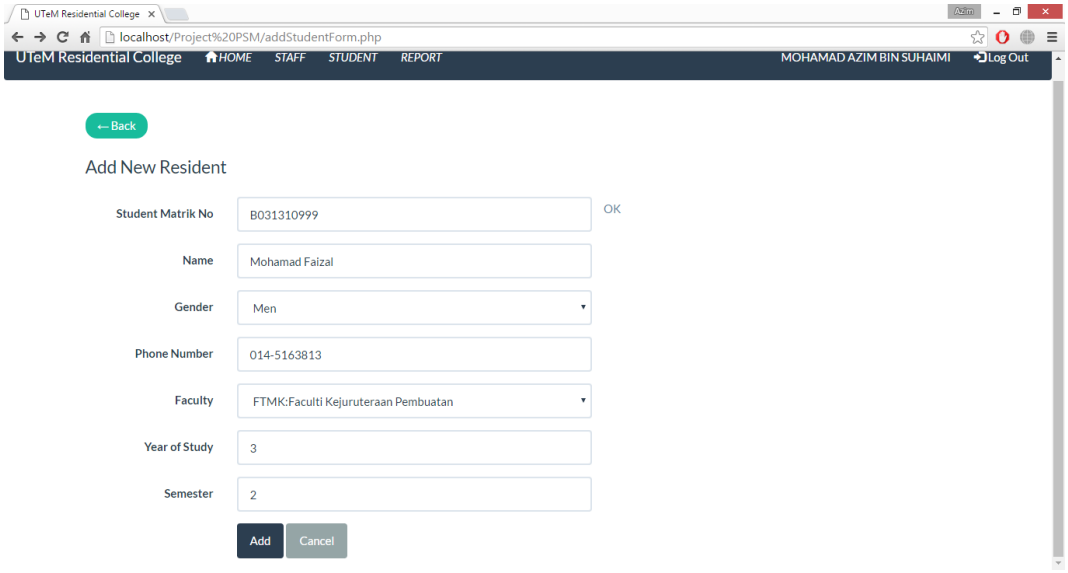


Figure 6.5: Valid Input for Each Field

3. Test Data for Staff Registration Module

Figure 6.6: Staff ID Already Exist

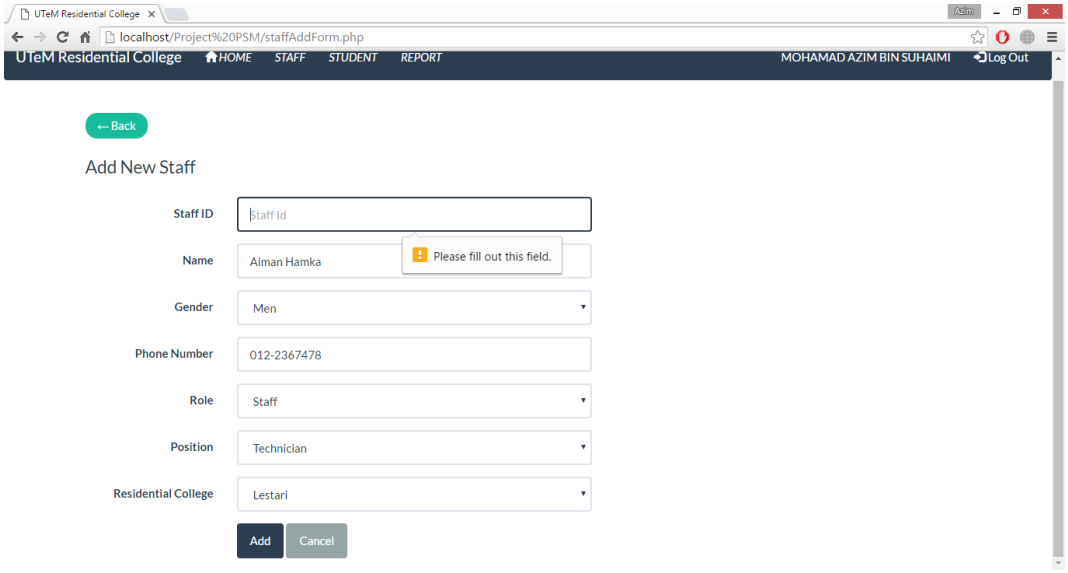


Figure 6.7: Either Field Blank for Staff Registration Form

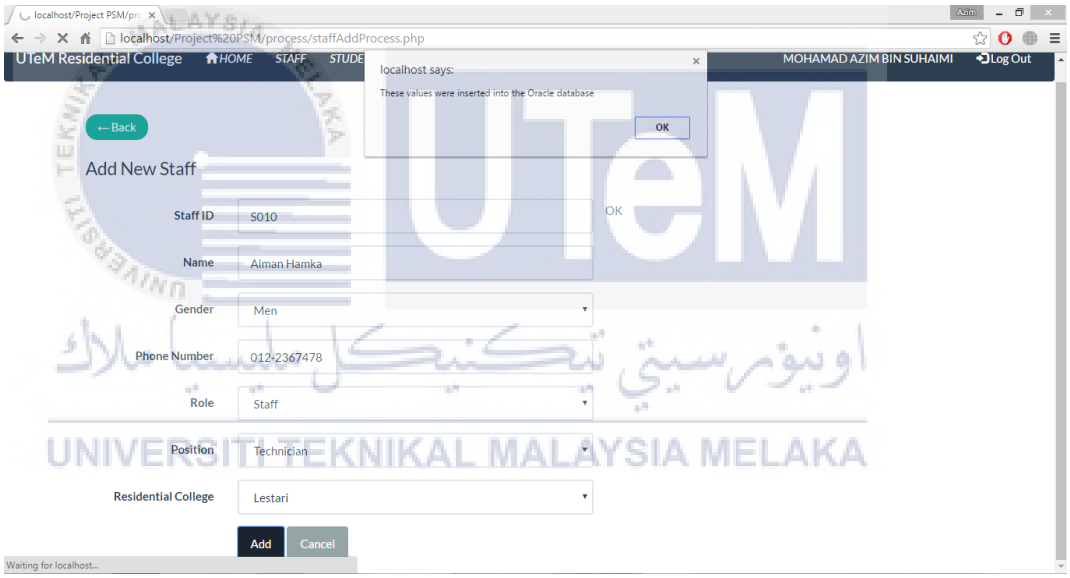


Figure 6.8: Valid Input for Each Field

4. Test Data for Delete Student Module

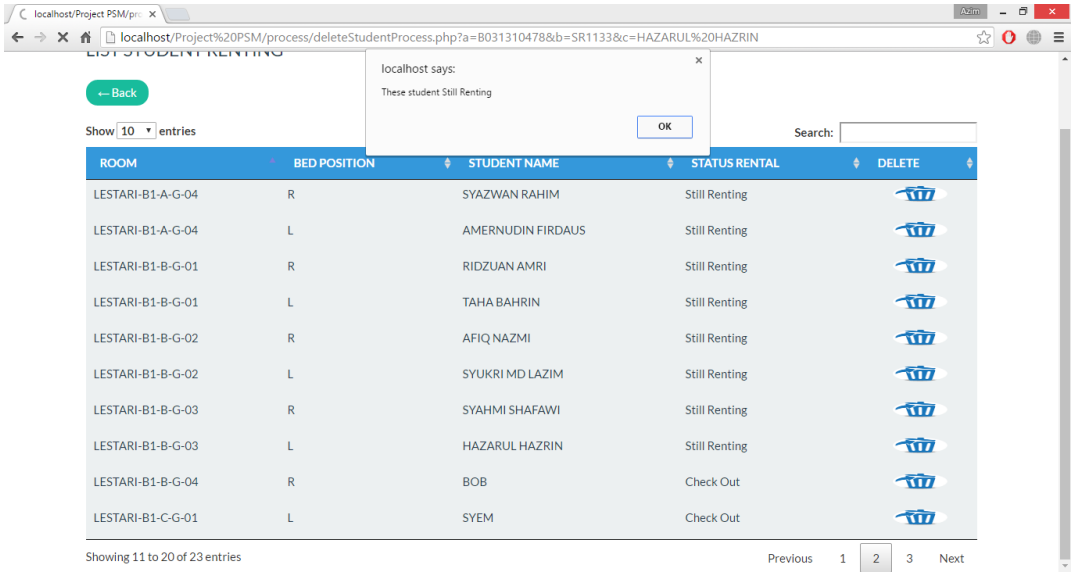


Figure 6.9: Student Still Renting

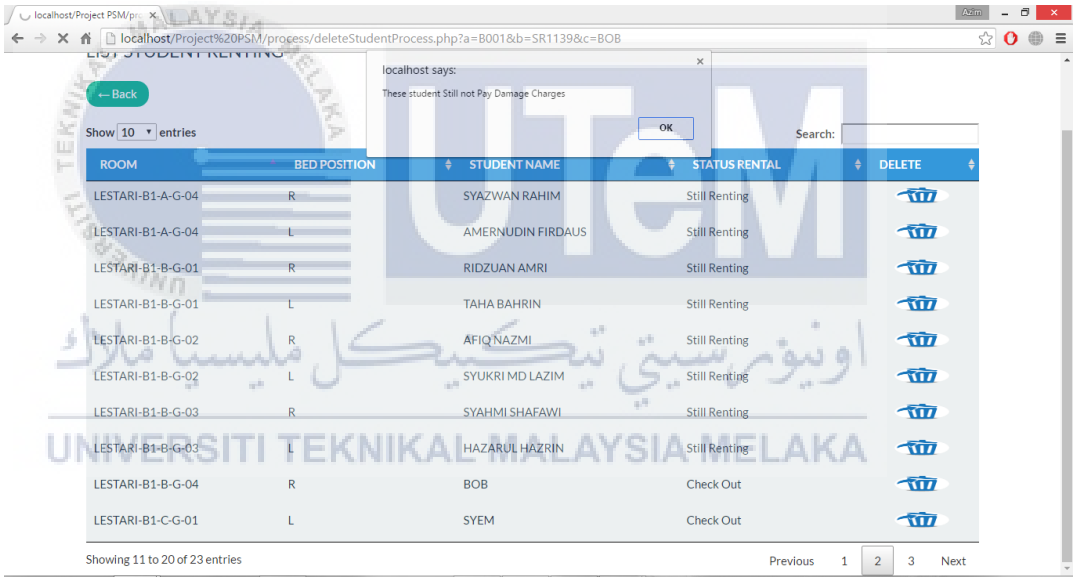


Figure 6.10: Student Still Not Paid the Charges of Damage

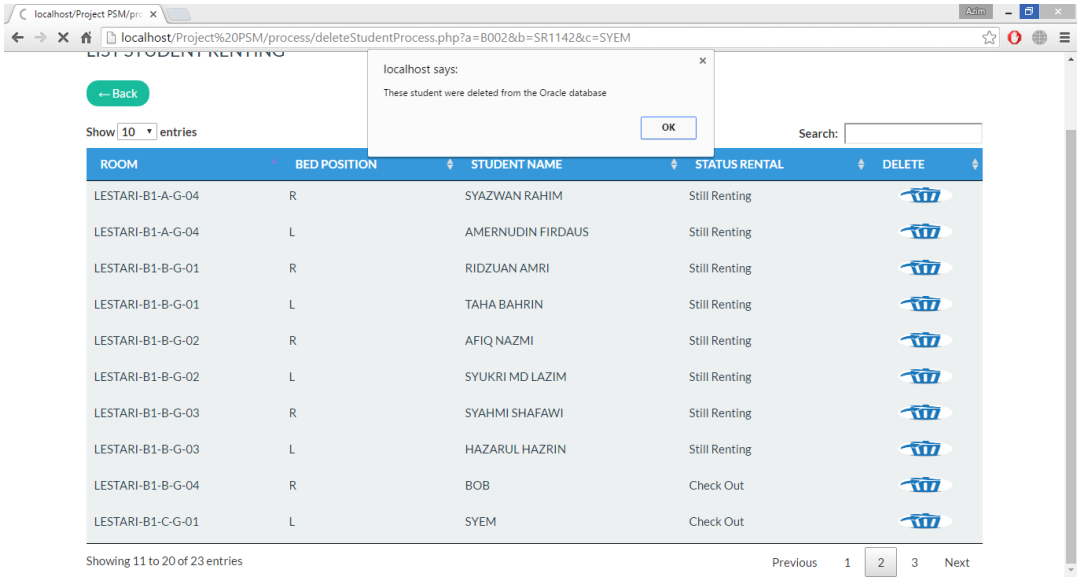


Figure 6.11: Valid Input for Deleting Info