



Faculty of Electrical and Electronic Engineering Technology



**DEVELOPMENT OF HOSTEL COMPLAINTS SYSTEM FOR UTeM
USING PHP MySQL DATABASE**

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

ASLIZA BINTI MD.TAHIR

Bachelor of Computer Engineering Technology (Computer Systems) with Honours

2021

**DEVELOPMENT OF HOSTEL COMPLAINTS SYSTEM FOR UTeM USING PHP
MySQL DATABASE**

ASLIZA BINTI MD.TAHIR

**A project report submitted
in partial fulfillment of the requirements for the degree of
Bachelor of Computer Engineering Technology (Computer Systems) with Honours**



Faculty of Electrical and Electronic Engineering Technology

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2021

DECLARATION

I declare that this project report entitled “Development of Hostel Complaints System for UTeM using PHP MySQL Database” is the result of my own research except as cited in the references. The project report has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

Signature

:

Asliza

Student Name

:

Asliza binti Md.Tahir

Date

:

01/01/2022

APPROVAL

I hereby declare that I have checked this project report and in my opinion, this project report is adequate in terms of scope and quality for the award of the degree of Bachelor of Computer Engineering Technology (Computer Systems) with Honours.

Signature

:


RADI HUSIN BIN RAMLEE

(Pensyarah)

Fakulti Teknologi Kejuruteraan Elektrik & Elektronik (FTKEE)

Universiti Teknikal Malaysia Melaka (Kampus Teknologi)

Hang Tuah Jaya, 76100 Durian Tunggal, Melaka

Supervisor Name

:

Radi Husin bin Ramlee

Date

:

11/01/2022

اونيورسيتي تيكنيكل مليسيا ملاك

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

DEDICATION

To begin, I would want to express my gratitude to Allah SWT for bestowing upon me strength, passion, time, mental strength, abilities, health, and protection.

This project is dedicated to my loved ones, who have been my pillars of support and inspiration throughout the most trying times of working on it. When I needed it most, they were there for me every step of the way.

Thank you to my supervisor, who has been a tremendous help to me while working on this project. He is always there for me, even when I'm feeling low, and he doesn't make me feel overwhelmed by the task at hand.



ABSTRACT

In the current era of technology, all management is typically managed by using an appropriate database. In this article, a UTeM hostel web-based complaint system was developed. This project aims to create a system called "Hostel Complaint System for UTeM" that can be accessed by student, admin and technician. This method may be helpful for users to report online regarding damaged equipment in the hostel. The idea comes when all students have difficulty complaining about the damage in their room or hostel since all students must come to the hostel office first before writing manually about the damage to the hostel. Furthermore, this outdated approach often causes delays in repair time because sometimes the complaint forms kept in the administrative office were missing or the administrator forgets to submit the complaints to the technician. Therefore, all these issues can be addressed with this system. This system uses a MySQL connection database with PHP language and use XAMPP as a server platform. This system uses XAMPP as a local host server as a testing development. This project has successfully functional for all three users which is admin, student and technician. To implement this project, it should be uploaded to the actual domain server so that it can be used for desktop and phone.

ABSTRAK

Pada era teknologi sekarang, semua pengurusan biasanya dikendalikan dengan menggunakan pangkalan data yang sesuai. Dalam artikel ini, sistem aduan berasaskan web asrama UTeM telah dibangunkan. Projek ini bertujuan untuk mewujudkan sistem yang disebut "Sistem Aduan Asrama untuk UTeM" yang dapat diakses oleh pelajar, pentadbir dan juruteknik. Kaedah ini mungkin berguna bagi pengguna untuk melaporkan secara dalam talian mengenai peralatan yang rosak di asrama. Idea ini timbul apabila semua pelajar mempunyai kesukaran untuk mengadu tentang kerosakan bilik atau asrama mereka kerana semua pelajar mesti datang ke pejabat asrama terlebih dahulu sebelum menulis secara manual mengenai kerosakan di asrama. Selain itu, pendekatan yang ketinggalan zaman ini sering menyebabkan kelewatan masa pembaikan kerana kadangkala borang aduan disimpan dalam pentadbiran pejabat hilang atau pentadbir lupa untuk mengemukakan aduan kepada juruteknik. Oleh itu, semua masalah ini dapat diatasi dengan sistem ini. Sistem ini menggunakan pangkalan data sambungan MySQL dengan bahasa PHP dan menggunakan XAMPP sebagai platform pelayan. Sistem ini menggunakan XAMPP sebagai pelayan hos tempatan sebagai pembangunan ujian. Projek ini telah berjaya berfungsi untuk ketiga-tiga pengguna iaitu pentadbir, pelajar dan juruteknik. Untuk melaksanakan projek ini, ia harus dimuat naik ke pelayan domain sebenar supaya ia boleh digunakan untuk desktop dan telefon.

ACKNOWLEDGEMENTS

Thank you to my supervisor, Mr Radi Husin bin Ramlee, for his invaluable assistance, words of wisdom and patience as we worked together in implementing this system.

In addition, I owe a debt of gratitude to UTeM and FTKEE for their assistance in making this project a reality. I would also want to thank my fellow 4BEEC students for their openness to share their ideas and opinions on the project.

In the end, I would want to express gratitude to my parents and other family members for their support and prayers during my academic career. All of my pals deserve a special mention for their encouragement and support.



TABLE OF CONTENTS

	PAGE
DECLARATION	
APPROVAL	
DEDICATIONS	
ABSTRACT	i
ABSTRAK	ii
ACKNOWLEDGEMENTS	iii
TABLE OF CONTENTS	i
LIST OF TABLES	iii
LIST OF FIGURES	iv
LIST OF APPENDICES	ix
CHAPTER 1 INTRODUCTION	1
1.1 Background	1
1.2 Problem Statement	2
1.3 Project Objective	3
1.4 Scope of Project	3
CHAPTER 2 LITERATURE REVIEW	4
2.1 Introduction	4
2.2 Past Related Project Research	4
2.2.1 Think Hostel Maintenance System (THMS) Using Dynamic Graphic – 2007	4
2.2.2 Web Application for Complaint Tracking and Resolving – 2017	7
2.2.3 Integration Telegram Bot on E-Complaint Applications in College – 2018	9
2.2.4 Hostel Management System with RFID – 2018	10
2.2.5 Smart Complaint Management System (SCMS) – 2018	11
2.2.6 The Android Based Mobile Application to Manage Student Complaints – 2020	14
2.2.7 Design and Implementation of Hostel Management System Using Java and MySQL – 2020	16
2.3 Comparison Table of the Past Project Related to Hostel Complaint System	18
2.4 Summary	19
CHAPTER 3 METHODOLOGY	20
3.1 Introduction	20
3.2 Design of Experimental	20

3.3	Elaboration of the Flowchart	22
3.3.1	Admin	23
3.3.2	Students	25
3.3.3	Technician	26
3.4	Implementations of Software and Languages	27
3.4.1	PhpMyAdmin	27
3.4.2	MySQL	28
3.4.3	XAMPP	28
3.4.4	PHP	29
3.4.5	HTML	29
CHAPTER 4	RESULTS AND DISCUSSIONS	30
4.1	Introduction	30
4.2	Interface of The System	30
4.3	Database of The System	41
4.4	Source Code of The System	44
4.5	Summary	48
CHAPTER 5	CONCLUSION AND RECOMMENDATIONS	49
5.1	Conclusion	49
5.2	Future Works	49
REFERENCES		50
APPENDICES		51



اونيورسيتي تيكنيكل مليسيا ملاك

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

LIST OF TABLES

TABLE	TITLE	PAGE
Table 2-1	Comparison table	18



LIST OF FIGURES

FIGURE	TITLE	PAGE
Figure 2.1	Warning and forecasting the weather for United State[1]	5
Figure 2.2	Maintenance by Map[1]	6
Figure 2.3	Flow of User side[2]	7
Figure 2.4	Flow of Admin/Authority side[2]	8
Figure 2.5	Complaint form for user[2]	8
Figure 2.6	Architecture of Bot token[3]	9
Figure 2.7	System Architecture[4]	11
Figure 2.8	Complaint Management Process[5]	12
Figure 2.9	System Architecture Overview[5]	12
Figure 2.10	Mobile Application of Home Page (Left) and Complaint Status Page (Right) [5]	13
Figure 2.11	Web Application of Manage Complaint Page for Engineer Department[5]	13
Figure 2.12	Comparison between existing complaint system[6]	14
Figure 2.13	An overview of PNSCares activity diagram[6]	15
Figure 2.14	Data Flowchart of registration process[7]	16
Figure 2.15	Data Flow Diagram[7]	17
Figure 3.1	Flowchart of project	21
Figure 3.2	Flowchart of Admins	23
Figure 3.3	Flowchart of Students	25
Figure 3.4	Flowchart of Technicians	26
Figure 3.5	Logo of phpMyAdmin	28
Figure 3.6	Logo of MySQL	28
Figure 3.7	Logo of XAMPP	29

Figure 4.1 Login page	30
Figure 4.2 Register Page	31
Figure 4.3 Admin Front Page	31
Figure 4.4 Admin Page - User/Student List	32
Figure 4.5 Admin Page - Technician List	32
Figure 4.6 Admin Page - Add New Account	33
Figure 4.7 Admin Page - New Complaint	33
Figure 4.8 Admin Page - Assign Task	34
Figure 4.9 Admin Page - On-going Complaint	34
Figure 4.10 Admin Page - Old Complaint	35
Figure 4.11 Admin Page – Logout	35
Figure 4.12 Student Page	36
Figure 4.13 Student Page - New Complaint	36
Figure 4.14 Student Page - Report History	37
Figure 4.15 Student Page - Edit Complaint	37
Figure 4.16 Student Page - Report History After Updated	38
Figure 4.17 Student Page – Logout	38
Figure 4.18 Technician Page - Task Assigned	39
Figure 4.19 Technician Page - Update Progress	39
Figure 4.20 Technician Page - Task Assigned History	40
Figure 4.21 Technician Page – Logout	40
Figure 4.22 Update Password Page	41
Figure 4.23 Database Table	41
Figure 4.24 Data for Account Table	42
Figure 4.25 Data for Complaint Table	42
Figure 4.26 Data for Role Table	43

Figure 4.27 Source Code of index.php for Admin	44
Figure 4.28 Source Code of index.php for Student	45
Figure 4.29 Source Code of index.php for Technician	45
Figure 4.30 Source Code of db.php	46
Figure 4.31 Source Code of usercheck.php	47
Figure 4.32 Source Code for Connect with db.php and usercheck.php	47



LIST OF APPENDICES

APPENDIX	TITLE	PAGE
Appendix A	Source Code of index.php for Admin	51
Appendix B	Source Code of index.php for Student	54
Appendix C	Source Code of index.php for Technician	59
Appendix D	Source Code of db.php	63
Appendix E	Source Code of usercheck.php	63



CHAPTER 1

INTRODUCTION

1.1 Background

A hostel complaints system is a set of procedures used by institutions to deal with complaints by students and resolve conflicts. Although managing complaints is difficult, an institution's performance can be measured by how quickly it responds to student complaints. The UTeM hostel is one of the institutions that receives complaints from students on a daily basis. They currently still use a manual form to file a complaint and the damage can take up to three days to be resolved. This might cause mistakes and make it difficult for the administrator to keep track of and manage everything in the hostel. Students have to meet with the administration at the hostel office for any complaints and wait a few days to guarantee the damage is fixed within the current system. For students with busy schedules, this is a waste of time. Although UTeM already provides full-facilities accommodations for students placed in the Durian Tunggal main campus, there are some UTeM's students still have to face the problems of damaged facilities in their hostel such as clogged sink, broken lamp and poor condition of door. To solve this problem, the project will develop a system to make it easier for UTeM students to lodge complaints online on damaged facilities in hostel. On top of that, this system will help them to make complaints through UTeM official website anytime and anywhere. Besides, it will be easier for the administration to check the list of complaints and its details and then assign the complaints to technicians. It will also lead the technicians to do the maintenance as soon as possible and update progress. Indirectly, this technique can help in saving the environment by reducing the usage of paper.

1.2 Problem Statement

The paper-based complaints system for hostels is inconvenient for both administrators and students. Keeping up with the database and responding to repair requests are both experiencing problems. As for UTeM, all students face problems to make a complaint about the problem in their room or toilet in the hostel since they have limit of time. Students have to go to the class in the morning and come back to the hostel in the evening. So, they do not have enough time to report the problems since the hostel's office only open from 8 A.M. until 5 P.M. and break during lunch time. There are also some students that have to go to their faculty which is outside of the main campus in Durian Tunggal everyday such as FTK, FKM and FPTT students. This situation also become a hassle for all those students since they have to go to the faculty as early as 7.30 A.M. when the office is not open yet since they have to follow the UTeM bus's schedule. Furthermore, usually they will return to the hostel by five or six in the evening when the office already closed. Thus, the problem here is these students do not have free time to go to the office to make complaints. Therefore, this project will utilize the web application as the medium to replace the paper-based complaint form by using PHP and MySQL web application so all students can make the complaints through UTeM official website anytime and anywhere. On the other hand, they do not have to wait for a long time for the problem to be solved since this system will make sure the technician will repair the problems in the hostel in the estimated time.

1.3 Project Objective

The main objectives of this project are:

- a) To develop a hostel complaints system for UTeM by using PHP and MySQL.
- b) To develop a web-based complaint system for hostel's students with a proper database.
- c) To give access for UTeM's hostel student to check the progress of their complaints.

1.4 Scope of Project

The scopes of work for the project include the following areas:

- a) This complaint system is specific for UTeM's hostel only.
- b) All data will be stored in MySQL and phpMyAdmin.
- c) The system can be accessed by students, admin and technician.
- d) Only admin can view the complaint list by all students, assign the complaints to certain technician and view the progress done by technician.
- e) Technicians can only view the tasks that already assigned for them and its details then update the progress.
- f) Students can make complaint, edit and delete the complaints that have been made and check the complaint's status

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

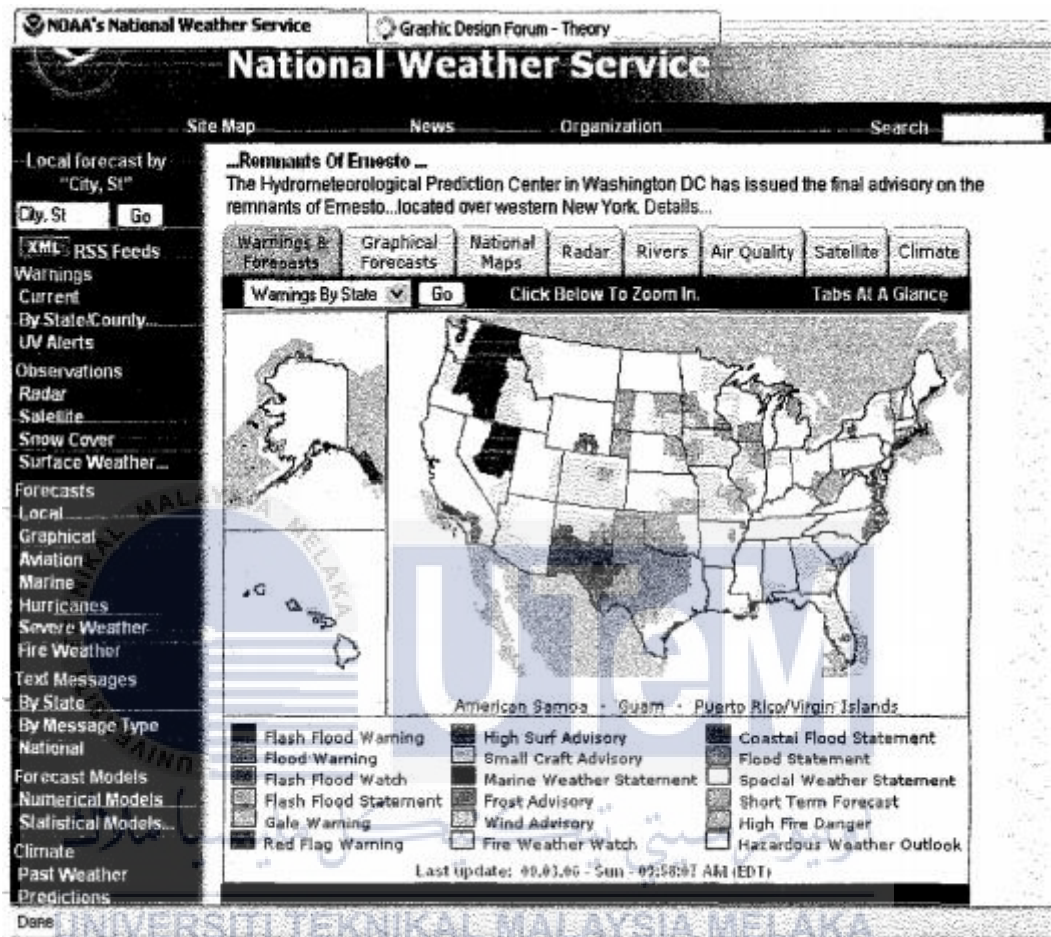
Analysis from different sources linked to a given research subject may be discussed as part of the literature review. This chapter also contains current knowledge, theoretical, and methodological contributions on a specific issue for this project. As part of the research brand, this chapter will explain the relationship between the previous study and the current study. Prior research in the field is researched in order to gain a better understanding and a suitable plan on how to improve this project. Articles, journals, theses, books, internet tools, newspapers, and other related sources that can be cited for this issue are the focus of the literature review.

2.2 Past Related Project Research

2.2.1 Think Hostel Maintenance System (THMS) Using Dynamic Graphic – 2007

Saifulnizam (2007) proposed a paper on the use of dynamic graphics as a medium of issue reporting instead of utilizing the text. The goal of this research is to use graphical images to report problems in the hostel maintenance system. This system might be used to record accommodation issues, assign tasks to the appropriate person, and track the issue's resolution. The idea of this medium comes since dynamic graphic has been commonly applied by the other web based system notably in weather and natural catastrophe prediction system[1]. According to the author, web pages that combine text with pictures is the most effective way to improve the user's interest[1]. The text-based system only benefits the people who know the technical thing in one field of technology and some other people might

be not understand what is all about[1]. National Weather Services is one system that uses dynamic visual mapping. The National Weather Service has utilised this type of technology to predict weather and sound warnings in earthquake-prone areas in the United States.



Source: www.nws.noaa.gov

Figure 2.1 Warning and forecasting the weather for United State[1]

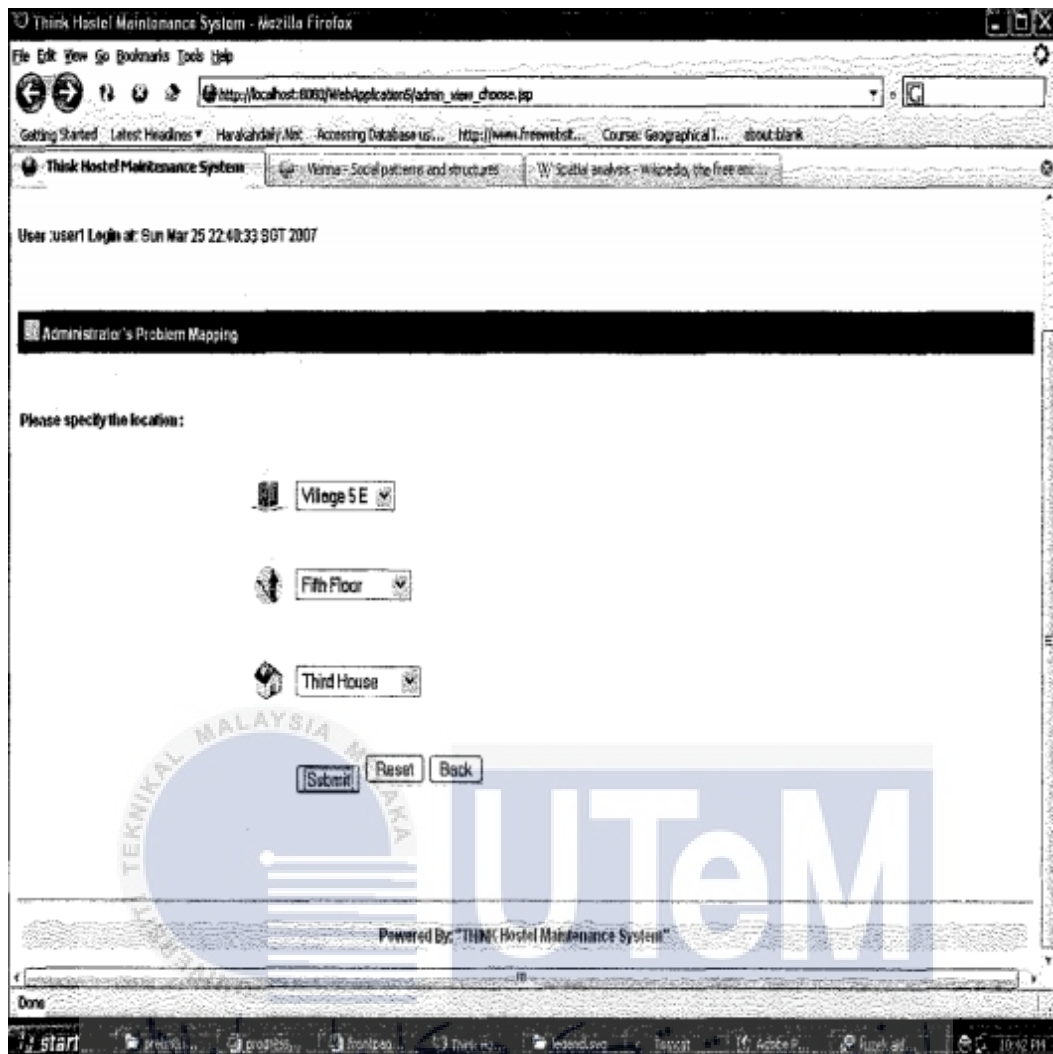


Figure 2.2 Maintenance by Map[1]

Based on Figure 2.2 above, this is one of the results of this system which is for administrator module. Administrator was able to choose which block, floor, house and room to be viewed by map. The system will display a map of the specific house when the administrator selects the relevant block, floor, and home. As a conclusion, this THMS were successfully build and able to transform the usage of paper-based into electronic means and the reports could be reviewed graphically and dynamically. Even though this THMS were success, there are several aspects of the project could be improved to make it more robust in terms of usability and functionality such as mobile integration was recommended by author to enable students to view and use the whole system via PDA or other mobile gadgets.

2.2.2 Web Application for Complaint Tracking and Resolving – 2017

This research focus on how to track complaints by develop web application to handle a variety of complaints at the hostel and college. The system's objective is to simplify the organizing, monitoring, management, and handling of complaints while also providing enterprises with a helpful tool for tracking complaints, identifying problem areas, and improving service. This method is internet-based. This system is divided into three components, each of which allows users to submit a complaint along with a photo, which is then handled by the appropriate authorities[2]. Users may view the progress of past complaints and provide comments to the system based on the quality of the service. According to Alve et al., (2017), sometimes responsible persons fail to settle the issues that have been reported in a timely manner, and higher authorities are unaware of this. So, this web application allows administrators to access to all complaints from all categories as well as feedbacks given by users and can take necessary action to enhance the service's quality.

This web application has two modules: one for the user and another for the administrator/authority. The Figure 2.3 below shows the flow of user side from log in to check status and the Figure 2.4 shows the flow of Admin/Authority side from selection to update.

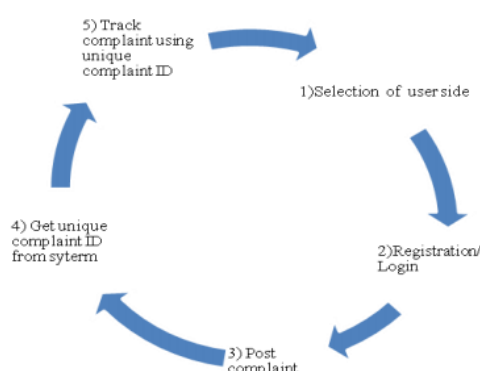


Figure 2.3 Flow of User side[2]

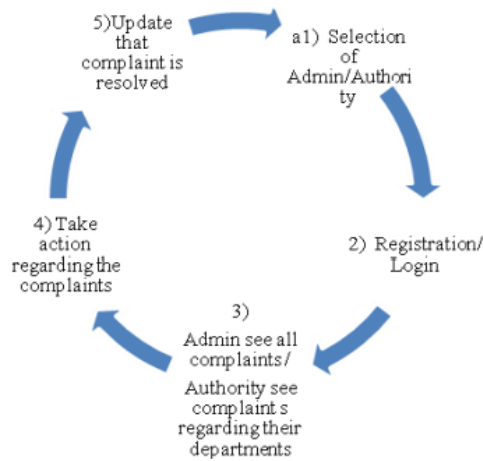


Figure 2.4 Flow of Admin/Authority side[2]

After user register and log in into the system, user can make and post complaint from the complaint form given as shows in Figure 2.3.

The screenshot shows a web form titled 'Complaint' with a 'UTeM' logo in the background. The form includes the following fields and options:

- Complaint Type***: A dropdown menu.
- Hostel**: A text input field.
- Complaint Subtype***: A dropdown menu with 'Electrical' selected.
- Complaint Discription***: A text input field.
- Date When Problem Occured**: A date picker showing 'mm/dd/yyyy'.
- Hostel Name***: A dropdown menu with 'Girls Hostel FE' selected.
- Room No***: A dropdown menu with '1' selected.
- Upload Photo Here**: A section with a 'Browse...' button and the text 'No file selected.'
- Post Complaint**: A blue button at the bottom right.

Figure 2.5 Complaint form for user[2]

Conclusively, this system considered successful since it achieves the objectives but this web applications is too simple and basic.

2.2.3 Integration Telegram Bot on E-Complaint Applications in College – 2018

M A Rosid et. al, (2018) proposed a research attempt to connect a telegram sender's application service with an e-complaint application at a college. The benefit of this application is users can easily submit a complaint via Telegram without have to visit the E-compliant application URL since the Telegram will then be routed to the E-complaint application[3]. IoT which represent for Internet of Things play the main role for this application to be successful since it allows the connection between Telegram smartphone messenger and the Arduino platform using Telegram Bot[3]. This paper's major contribution was to show the possibilities of Telegram apps built on IoT using the Bot Telegram API on various platforms for communication between web platforms and Telegram.

Based on the experimental methods in this article, Telegram API was used to build the Telegram Bot. Then, the Bot token (UmsidaBot) was used to allow the communications between the Telegram Bot and e-Complaint application. Referring to the Figure 2.6, we can see that the system is divided into two flows. The user's complaint information was transmitted to the telegram bot for the first flow. The telegram is then used to convey the message to the e-complaint server. To ensure that complaints are regularly updated, the e-complaint system must be updated periodically. When the e-complaint server gets a message from the user, it sends a response to the telegram, which then delivers the response to the user.

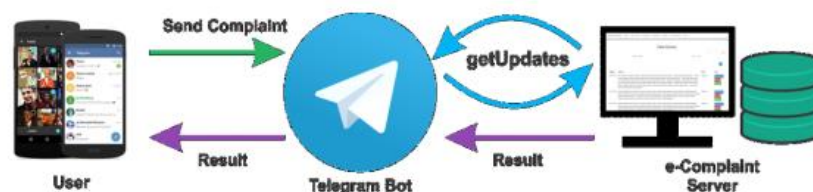


Figure 2.6 Architecture of Bot token[3]

2.2.4 Hostel Management System with RFID – 2018

This project is anticipated to minimize people's labor and make the distribution of hostels simpler for students and hostel managers with the aid of the hostel web application, pick students from the waiting list and mess accounts, discharge generation, complaint registration and so on[4]. The Hostel Management System is an internet website designed to manage various activities in the hostel. With a user-friendly GUI, it is effortless to automate, organize and manage all operations to administrate hostel bureau[4]. The website will be a massive relief to the staff. It is advantageous in big institutional organizations with a high number of hostels. In this system, four languages were used to develop the website which is HTML, CSS, JavaScript and PHP. Analyzing the inadequacies of the current system and designing a computerized system with a user-friendly GUI and efficient performance will assist in the resolution of the current system's challenges.

Radio Frequency Identification (RFID) labels were first developed to replace barcode readers in supply chains. RFID tags function as the RFID device's receiver, emitting electromagnetic radiation that induces a current in the tag's antenna. RFID technology has the potential to become widely used sooner rather than later. It is now being used efficiently as a component of inventory network management to monitor. It enables us to monitor any material, object, or human person by applying it correctly, and it also allows us to save time without the participation of any human effort. The planned system's primary goal is to assist in the automation of fundamental hostel administration functions. The actual hostel administration operations include entering student information and other data, enrolling complaints on the complaints page, and monitoring student mobility using RFID.

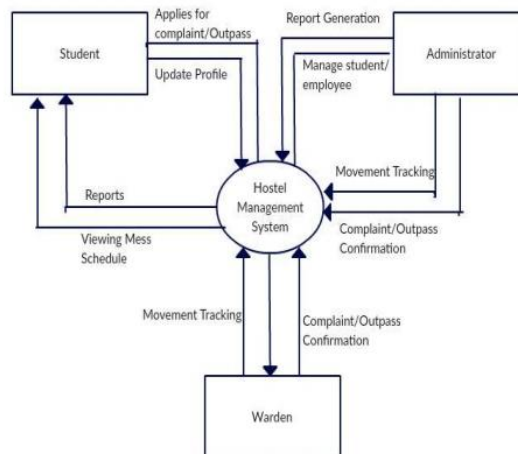


Figure 2.7 System Architecture[4]

Based on the Figure 2.7 above, there are three different modules which is administrator, employee/warden and student. Some of the functionalities can also be used in my project since there are three users and there is complaint function. The only downside of this system is the current programming's functioning demands an appropriate approach to programming progress

2.2.5 Smart Complaint Management System (SCMS) – 2018

Some customers may feel unsatisfied with the service when they receive the service delay, do not know how the complaint can be filed, and also have difficulties with the present complaint processing within businesses. The creators of this project have thus built the Smart Complaint Management System (SCMS), which includes the mobile, chat board and online application, to resolve customer disappointment. Furthermore, the SCMS features a service that categorizes the complaint and then automatically sends it to the appropriate department and service that finds a similar issue to avoid a repeated complaint.

Customers' complaints are increasingly crucial to the firm. Thus it is imperative that it pay attention to them and deal with any issues quickly.

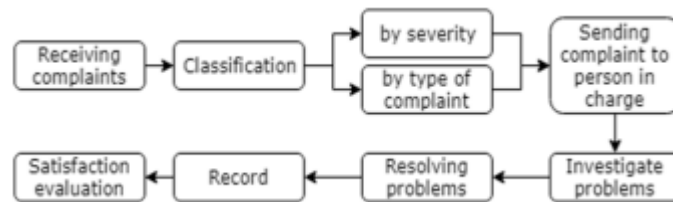


Figure 2.8 Complaint Management Process[5]

A complaint management process is a series of procedures used in businesses to address complaints and address problems. Figure 2.8 depicts the complaint handling process.

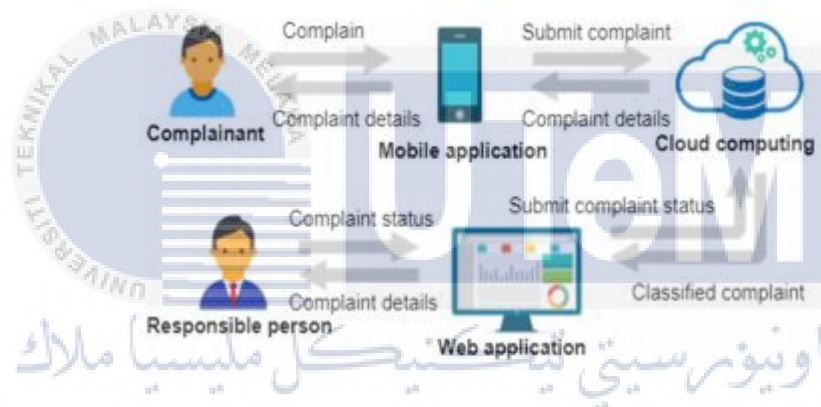


Figure 2.9 System Architecture Overview[5]

The complainant reports the problem or complaints using the mobile app or chatbot. The complaint was then directed to the cloud computing system, which analysed it and saved the data in a database. The web application retrieves all classified complaints from the database on the staff side, allowing the responsible individual to view all open complaints via the system dashboard. Additionally, the liable individual may take notes on the remedial procedure and update the complaint status before returning to the complainant. Figures below are some of the interface designs for this system.

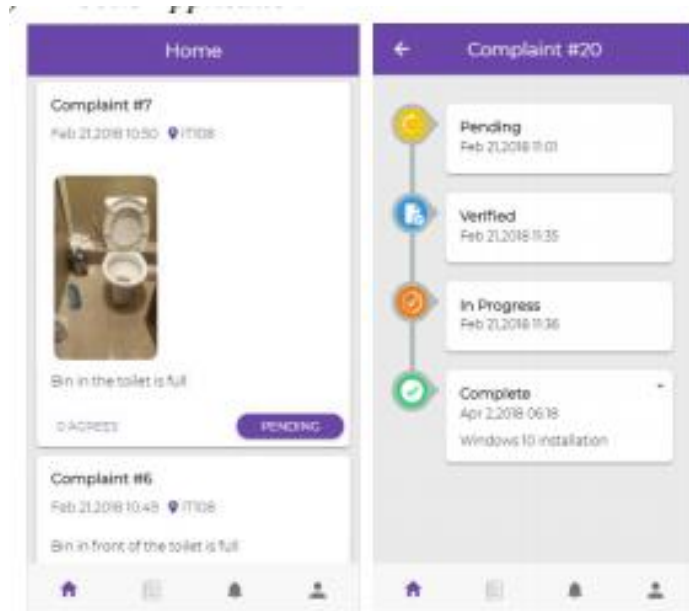


Figure 2.10 Mobile Application of Home Page (Left) and Complaint Status Page (Right) [5]

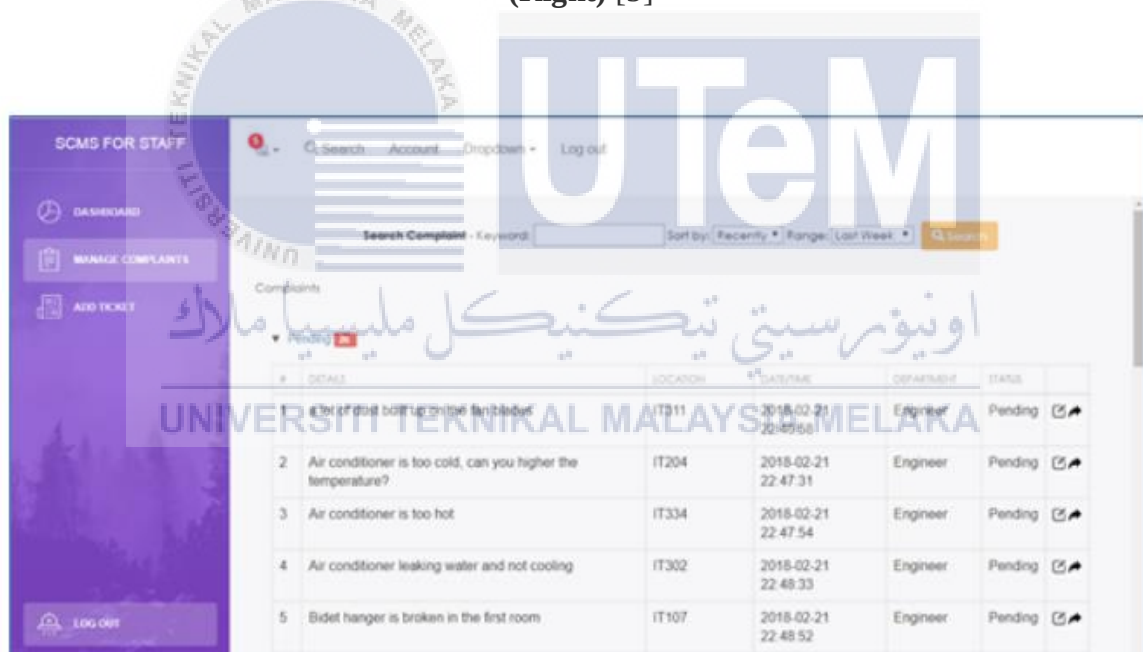


Figure 2.11 Web Application of Manage Complaint Page for Engineer Department[5]

As a conclusion for this system, the test results indicate that it has the potential to significantly reduce complaint processing time and procedures, increase the channel for filing complaints, and enhance the channel for progress reporting and monitoring the status of the complaint.

2.2.6 The Android Based Mobile Application to Manage Student Complaints – 2020

The researchers created a mobile Android application to manage student complaints in this study. With mobile applications, the administration can avoid the loss of complaint forms and keep track of the registrations. This technology provides comments and alerts directly to students' cell phones. The mobile phone is a frequently used device, and it does not take much time to use the mobile app. The biggest difference between manual and automated complaints is throughout the procedure. The existing approach is longer and more inefficient than mobile apps, which will shorten the length of time and give the students a faster answer. The creation of PNSCares, a mobile app for managing student concerns is discussed in this paper. They used Android to construct PNSCares as a mobile platform. Android is commonly utilised at Politeknik Nilai as a result of a brief survey conducted by the HEP officer himself[6]. Figure 2.1.2 shows the comparison between the three existing complaint systems and Figure 2.1.3 is an overview of PNSCares activity diagram.

No.	Functions	eAduan (UiTM)	eAduan (UMK)	eAduan (USM)
1	Sign Up	X	X	X
2	Login	√	X	X
3	Make complaint	√	√	√
4	Choose category of complaint	√	√	√
5	Upload picture	X	X	X
6	Auto date/time	√	X	X
7	List of history	√	X	X
8	Received notification	X	X	X
9	Check feedback	√	√	√

Figure 2.12 Comparison between existing complaint system[6]

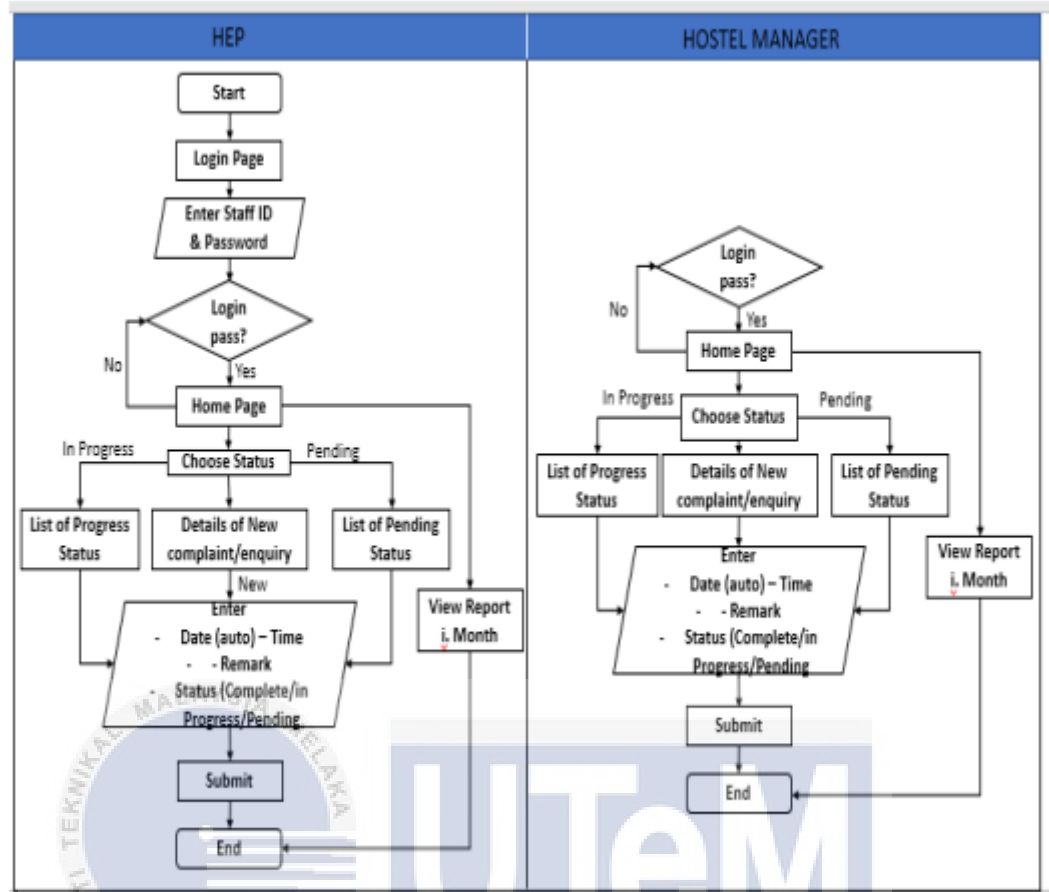


Figure 2.13 An overview of PNSCares activity diagram[6]

Based on the SUS results and user testing, they determined that PNSCares is a must-have mobile app at the institution. The applications are simple to use, user-friendly, and well-functioning. The developers were able to create a prototype of PNSCares by adhering to the ten (10) guidelines for successful complaint processing. User Focused, Visibility, Accessibility, Responsiveness, Objectivity and Fairness, Confidentiality, Remedy, Review, Accountability, and Continuous Improvement are the principles[6]. The eight (8) previously listed principles do not include the Remedy and Review.

2.2.7 Design and Implementation of Hostel Management System Using Java and MySQL – 2020

This project encourages students and the management group to save records of all things in hostels. This application needs students and administrators to log in data via the application dashboard to quickly view information on their hostel room registration, fees payment, student records and update as needed. This project works in the languages JAVA and SQL. JAVA is an object-oriented programming language that has minimal implementing requirements and SQL is a structured query language which allows you to access and manage a database[7].

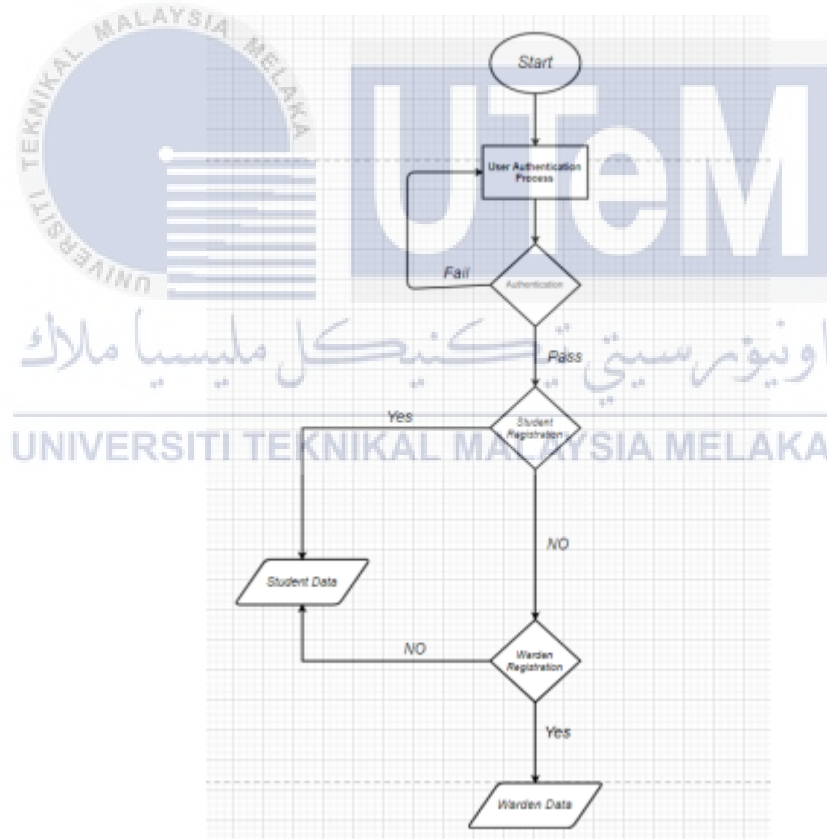


Figure 2.14 Data Flowchart of registration process[7]

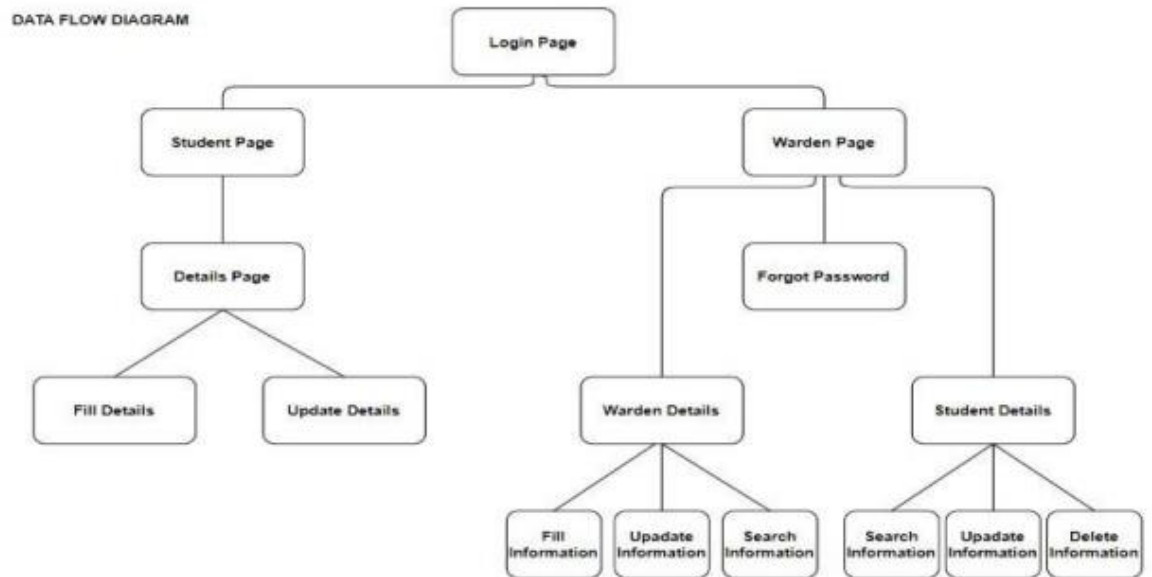


Figure 2.15 Data Flow Diagram[7]

The proposed system consists of three different modules which are administrator module, warden module and student module. One of the advantages of the suggested system is that the existing Hostel Management System is highly user friendly and efficient. In addition, the database use makes it easy and fluent to retrieve data. Eventually, this project also meets the aims.

2.3 Comparison Table of the Past Project Related to Hostel Complaint System

Table 2-1 Comparison table

No.	Title	Year	Characteristics				
			Method used	Interface	User friendly	Security	Feedback Space
1.	Think Hostel Maintenance System (THMS) Using Dynamic Graphic	2007	Dynamic Graphic	Good	Yes	Yes	No
2.	Web Application for Complaint Tracking and Resolving	2017	<i>Not stated</i>	Too simple	Yes	No	Yes
3.	Integration Telegram Bot on E-Complaint Applications in College	2018	Telegram Bot	Fixed	Yes	Yes	No
4.	Hostel Management System with RFID	2018	PHP, MySQL, RFID	<i>Not displayed</i>	Yes	Yes	No
5.	Smart Complaint Management System (SCMS)	2018	Mobile application, web application	Interesting	Yes	Yes	No
6.	The Android Based Mobile Application to Manage Student Complaints	2020	Android based	<i>Not displayed</i>	Yes	Yes	Yes
7.	Design and Implementation of Hostel Management System Using Java and MySQL	2020	Java, MySQL	Too Simple	No	No	No

2.4 Summary

Online complaints system is one of the crucial technologies that should be implemented to all organizations out there. By reading the complaints, action can be taken and some improvement to the organization can be applied. Based on the past related project research above, some systems can be applied in my hostel complaints system project. As example, the web-based interface by using PHP and MySQL as mentioned in the title. There are some functionalities that I can add to my system for the admin, student and technician such as managing password, feedback space and dynamic graphic to send the pictures of the location or damaged facilities. More details about my system will be discussed in the next chapter.



CHAPTER 3

METHODOLOGY

3.1 Introduction

In this chapter, we will go through the methods that we have been conducted to get the preliminary results of this project. All work process, methodology and techniques which were implemented in developing the hostel complaint system will be describe in this chapter. The objective of this project is to develop a hostel complaints system for UTeM by using PHP and MySQL. Basically, four types of software were used to build this system which are PHP, MySQL and XAMPP and HTML. The next section will provide an overview of the work process before delving into the specifics of software development.

3.2 Design of Experimental

The flowchart of Figure 3.1 shows that the flow of process of this project. The project is started by doing some research about the title which is ‘Development of Hostel Complaints System for UTeM using PHP MySQL Database’. Many articles have been found about this system, but only a few of them were about hostel complaints. Those articles have been reviewed in the chapter 2, literature review. The next step after doing research is the problem statements and objectives have been decided and was written in chapter 1, introduction. After I understand the flow of this project by referring to a few videos in YouTube, the flowchart was built to make us more understand about the whole process and how the system works. Then, I have been looking for suitable codes that can be used for this project by using ‘bottom-up approach’ method. After the codes were found, I will discuss about the flowchart and codes with my supervisor to ask for his guidance and troubleshoots

any errors before the codes were modified and run corresponding to the title. Then, I will troubleshoot the codes again and again until the codes were successfully function. Lastly, the results were analyzed with its data and report was written.

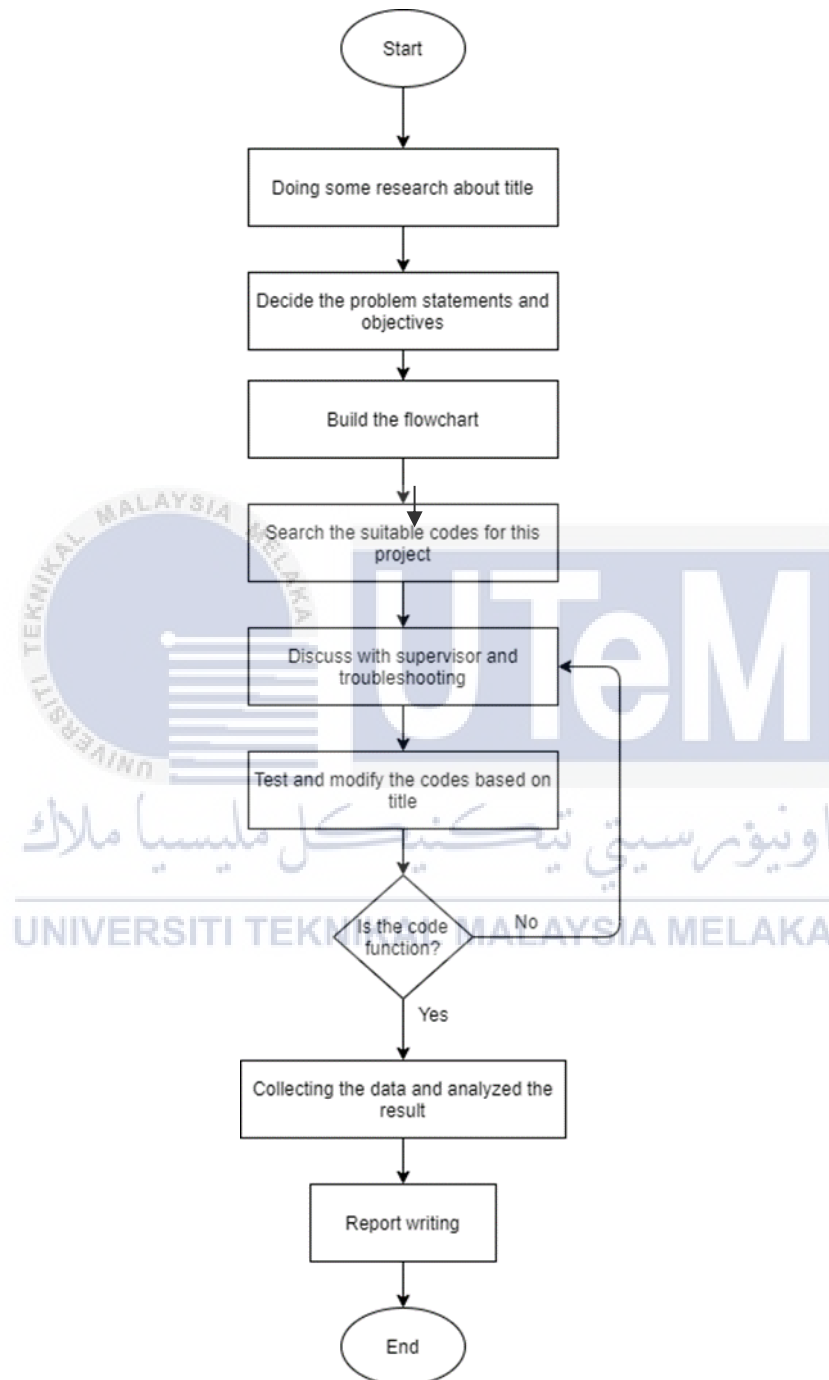


Figure 3.1 Flowchart of project

3.3 Elaboration of the Flowchart

This system consists of three users which is admins, students and technicians. Each flowchart will be elaborated below to make sure we become more understand about each user's page.



3.3.1 Admin

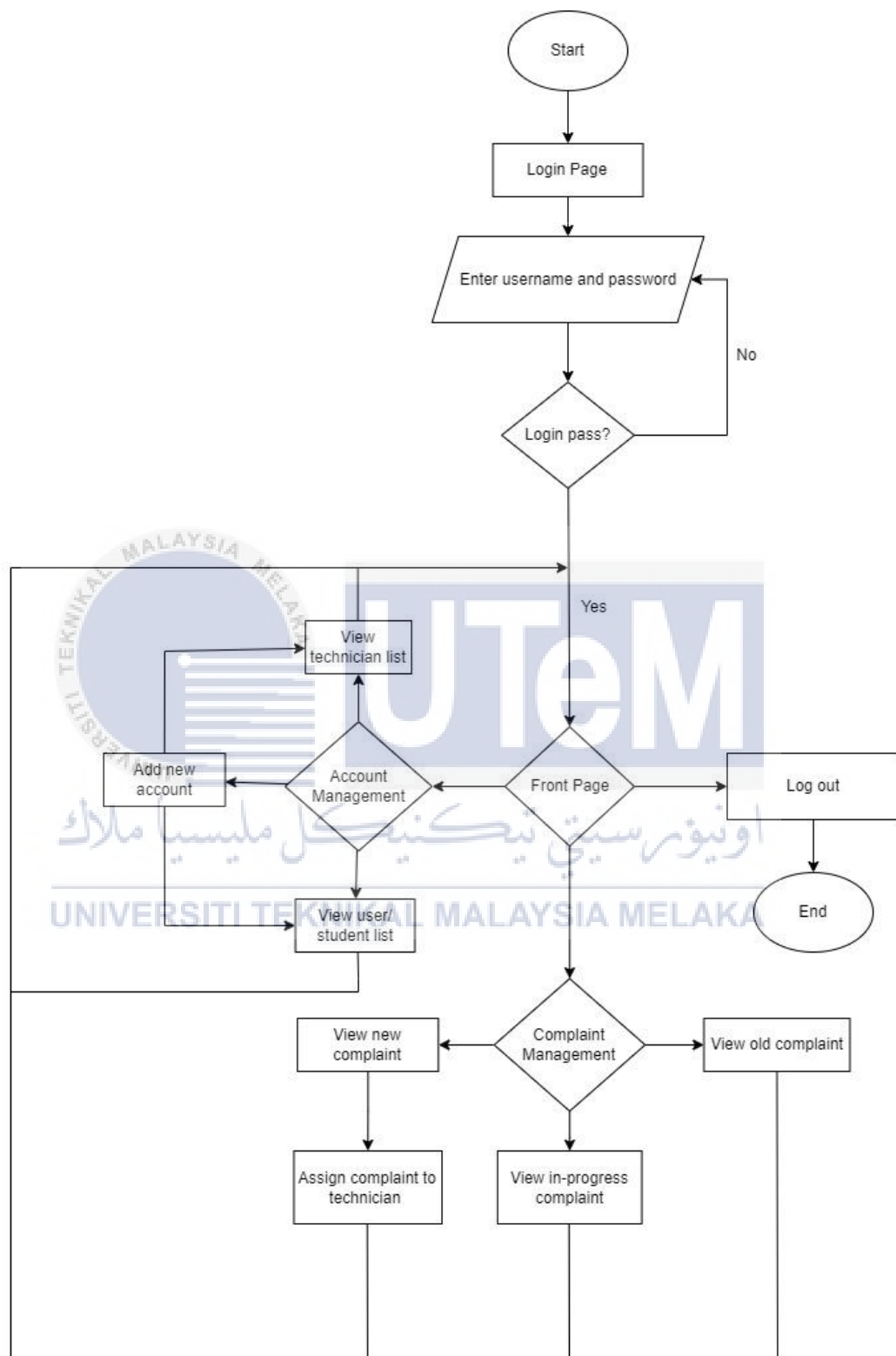


Figure 3.2 Flowchart of Admins

Based on Figure 3.2, to log in the system, admins have to enter the username and password. Then, if both username and password is correct, the system will display the front page or otherwise the admins have to log in again with the correct username and password. At the front page, admins have to choose between three options which is the account management, the complaint management and log out. If the admins choose account management, admin can view the technician list, user or student list and add new account. If admin choose the complaint management, admin can view new complaint, in-progress complaint and old complaint. For the new complaint part, admin have to assign the complaint to the certain technician suitable with their job scope.



3.3.2 Students

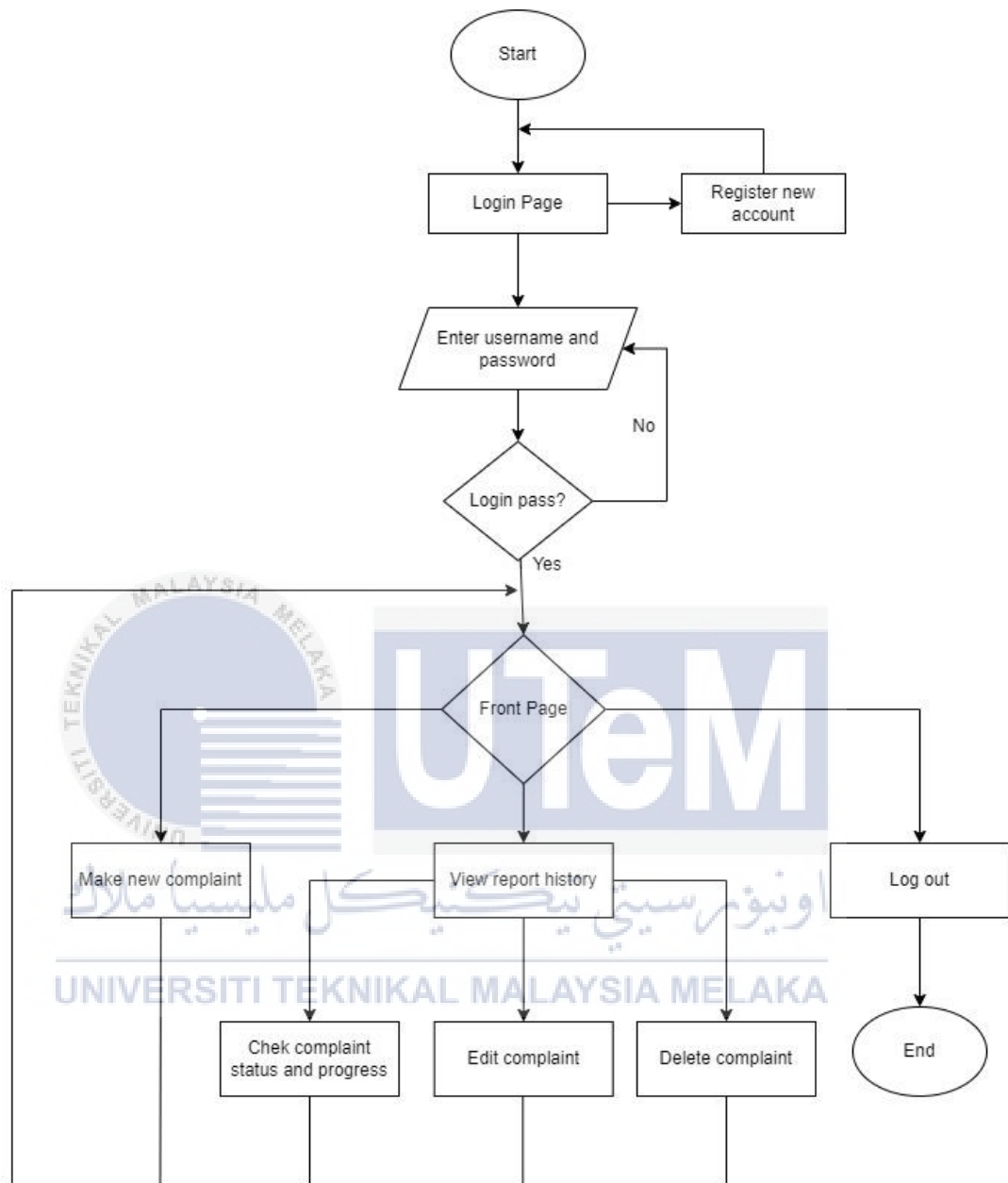


Figure 3.3 Flowchart of Students

As for students, they have to use their username and their password to log in. After they already log in into the system, the front page will be shown with three options which is make new complaint, view report history and log out. To make a complaint, they have to fill the complaint's details in the form such as email, date of complaint, phone number, location

and problem description. After the complaint has successfully submitted, they were brought back to the front page and they can choose another option which is view report history or just log out. In the view report history part, they can also check the complaint status and progress updated by technician, edit the old complaint or delete complaint.

3.3.3 Technician

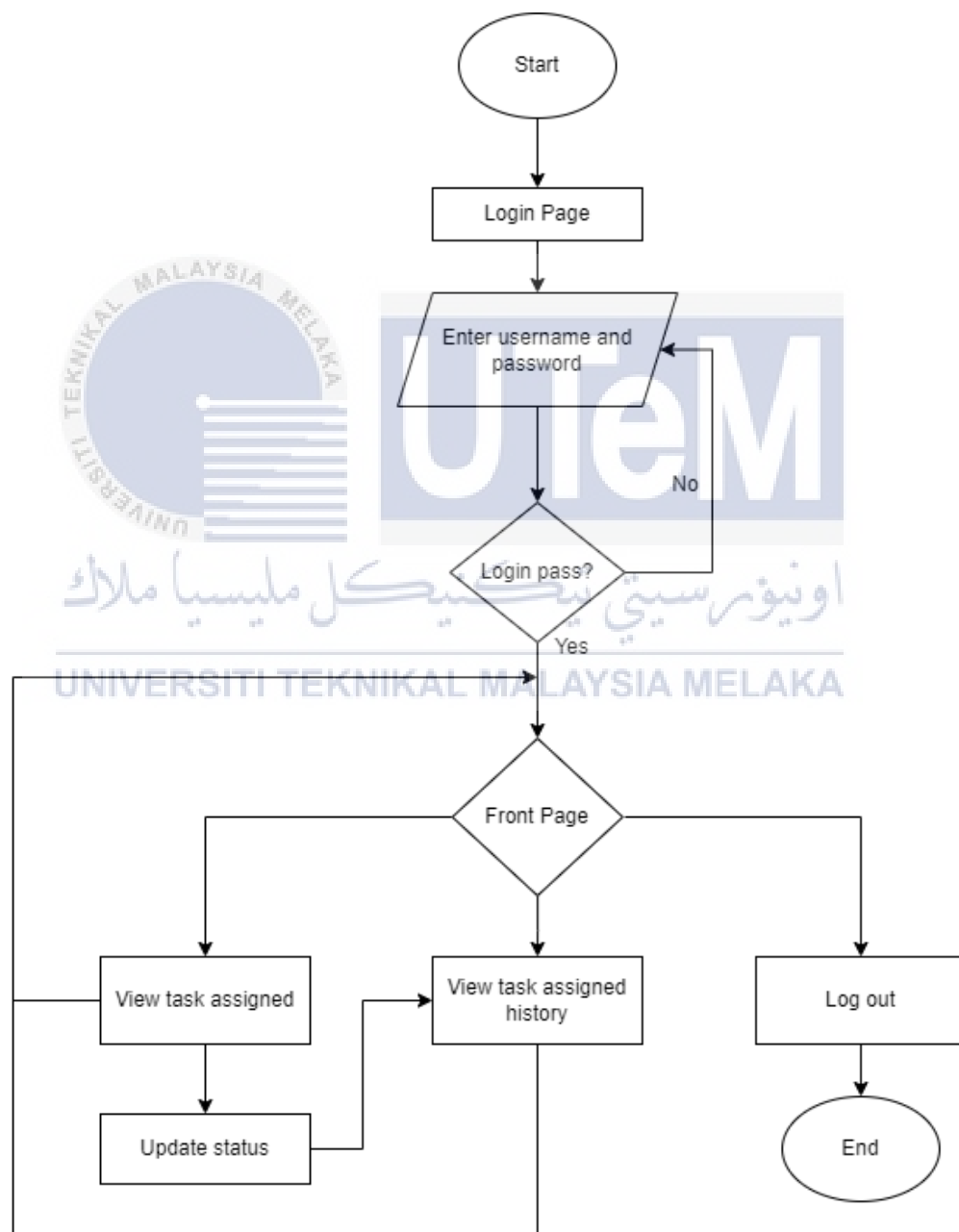


Figure 3.4 Flowchart of Technicians

The same way used for technicians where they have to log in using their username and password. Then, the front page will also show three options which is to view task assigned, view task assigned history, and log out. Their tasks will be shown when admins have assigned the complaint to them and they need to do the maintenance as soon as possible since admin and student can check their progress day by day. Besides, technician also have to update their progress for the maintenance.

3.4 Implementations of Software and Languages

There are three types of softwares were used to build this system which are phpMyAdmin, MySQL and XAMPP and two types of languages which are PHP and HTML. All these softwares must be installed to make sure the system can be run and function well.

3.4.1 PhpMyAdmin

PhpMyAdmin is the PHP-written open-source software utility. It is a third-party tool for managing tables and data in the database that supports several MariaDB and MySQL operations. The primary function of phpMyAdmin is to manage MySQL web management. PhpMyAdmin is the most common MySQL database administration tool. We may use this tool to create, update, change, delete, import, and export MySQL database tables with this program. These actions can be performed via a user interface while any SQL query is still being executed. PhpMyAdmin provides a wide variety of operations on MySQL and MariaDB, such as managing databases, relationships, tables, columns, indexes, permissions and users.



Figure 3.5 Logo of phpMyAdmin

3.4.2 MySQL

MySQL is a relational database management system (DBMS) that stores data in tables, which are database objects. A table is a collection of linked data items comprised of columns and rows. When it comes to classifying data, databases come in useful. We may query a database for detailed information and get a record set in response using MySQL. MySQL is advantageous in that it can be scaled down to suit embedded database applications. MySQL is the de-facto industry standard database for websites that handle large quantities of data and have many end-users (such as Yahoo and Google). As a consequence of this reputation, many people think MySQL can only operate small to medium-sized systems.

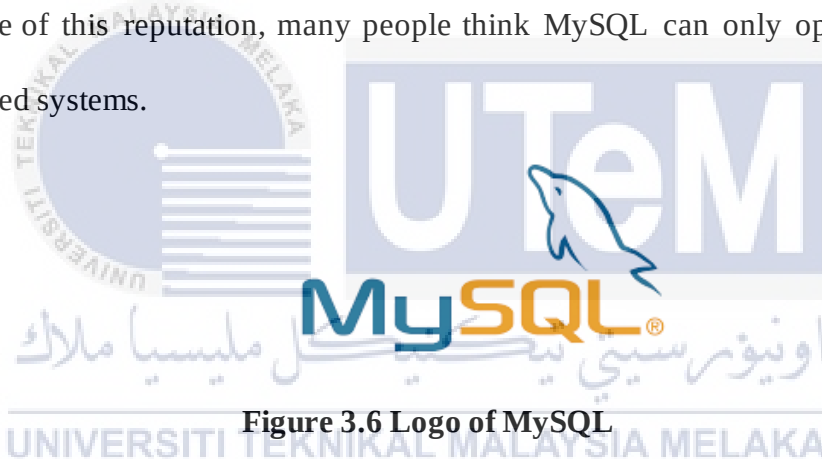


Figure 3.6 Logo of MySQL

3.4.3 XAMPP

XAMPP are the acronyms for Cross-Platform (X), Apache (A), MySQL (M), PHP (P), and Perl (P). It is a compact and lightweight Apache package that simplifies the process of setting up a local web server for testing purposes. It is a development tool that enables web designers and programmers to test their work on their computers without an internet connection. Additionally, XAMPP is cross-platform, meaning it may operate on any operating system, including Linux, Mac OS X, and Windows. Since most existing server installation systems share many of the same components as XAMPP, migrating from a local test server to a live server is simple.



Figure 3.7 Logo of XAMPP

3.4.4 PHP

PHP is a server-side programming language that is used to create both static and dynamic webpages as well as online applications. PHP is an acronym for Hypertext Preprocessor, which was previously abbreviated for Personal Home Pages. To run a PHP script, we must have PHP installed on the server where we want to run it. A web browser is all that is needed to access the PHP scripts on the client machines. The file ending ".php" signifies a PHP file, which contains PHP tags.

3.4.5 HTML

HTML, or Hypertext Markup Language, is a markup language used to describe the structure of publications for retrieval over the Internet using World Wide Web technologies (WWW). HTML utilizes a range of tags and attributes to define the web page structure and layout. The head and body are the two main elements of every HTML page. The HTML tags are not displayed in the browser but are used to understand the website's content. A browser would not be able to format an HTML page. HTML is the World Wide Web Consortium (W3C), which is usually adopted by major browsers such as Internet Explorer.

CHAPTER 4

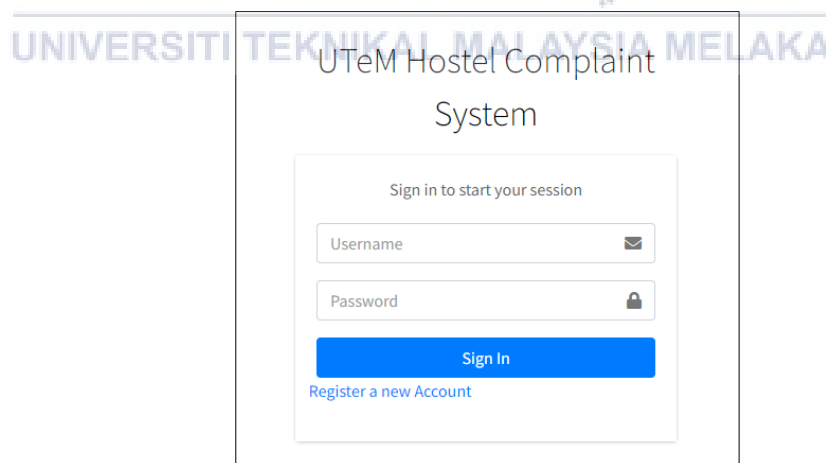
RESULTS AND DISCUSSIONS

4.1 Introduction

This chapter presents on how to use Hostel Complaints System for UTeM and how it works for all three users which is admin, technician and student. Besides that, this chapter will also show and explains the database and some source codes of the systems.

4.2 Interface of The System

Firstly, we have to make sure all the files and database that we need to run this system is complete and put in the htdocs folder in xampp files. Then we can run it by using link <http://localhost/compsys/index.php>. Noted that the 'compsys' is the example of the folder name in htdocs and it need to be included in the source codes.



UTeM Hostel Complaint System

Sign in to start your session

Username 

Password 

Sign In

[Register a new Account](#)

Figure 4.1 Login page

To login, all the users need to insert their username and password. For students, they needs to register their account first before they login if they do not have any account yet.

UTeM Hostel Complaint System

Register

Username

Password

Retype password

[Register](#)

[I already have a account](#)

Figure 4.2 Register Page

For register page, students just need to input their username, password and retype the password and make sure both of the passwords is same.

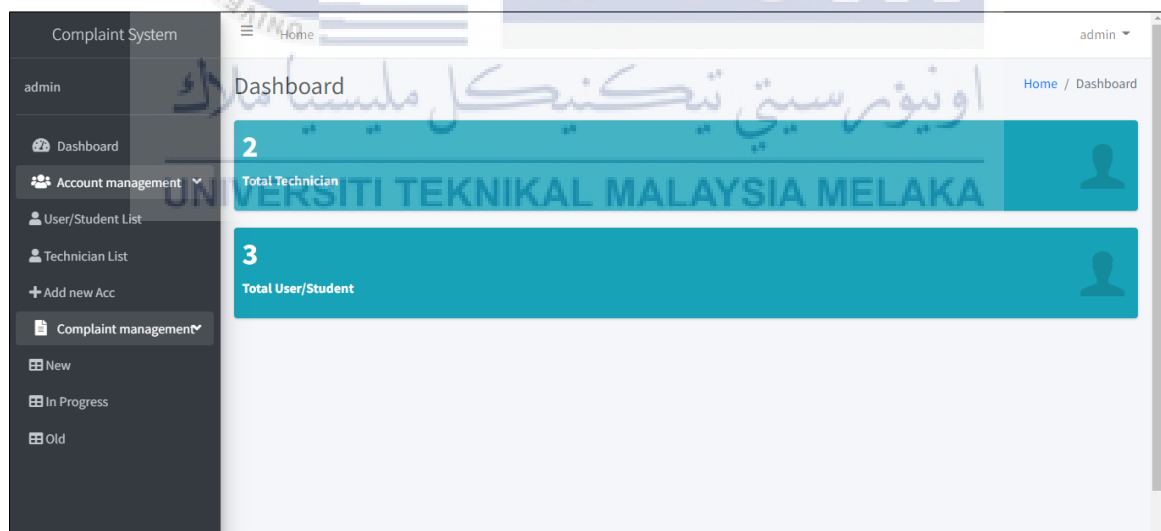


Figure 4.3 Admin Front Page

Based the figure 4.2 above, admin can see total numbers of technician and students at the admin front page. Then, admin can choose either to click the account management, complaint management or log out.

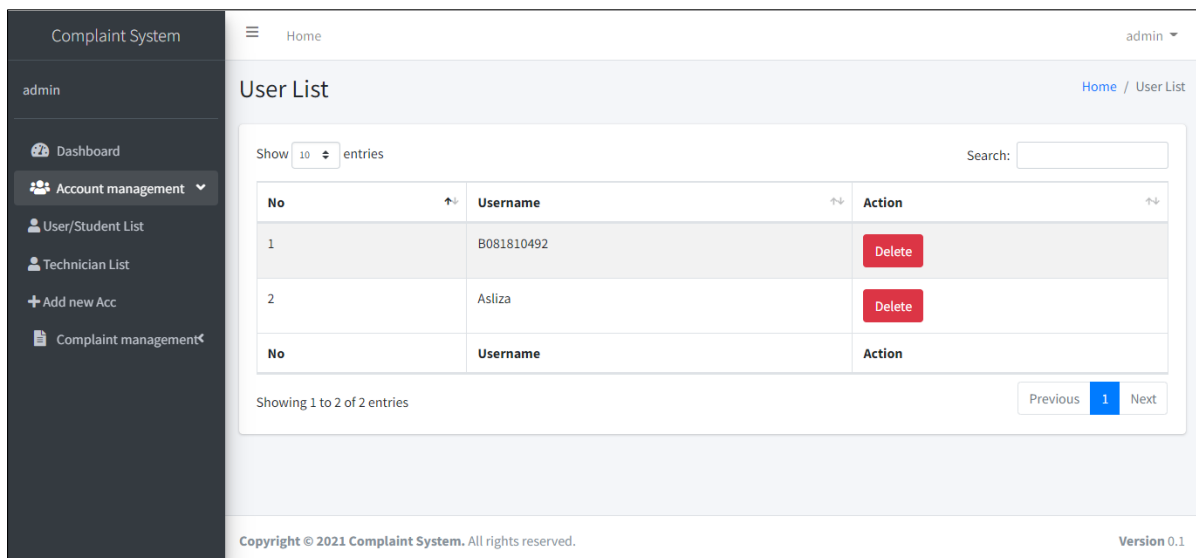


Figure 4.4 Admin Page - User/Student List

For figure 4.3 above, it show the page of account management in admin page for user/student list. Admin can delete the user or student from the list.

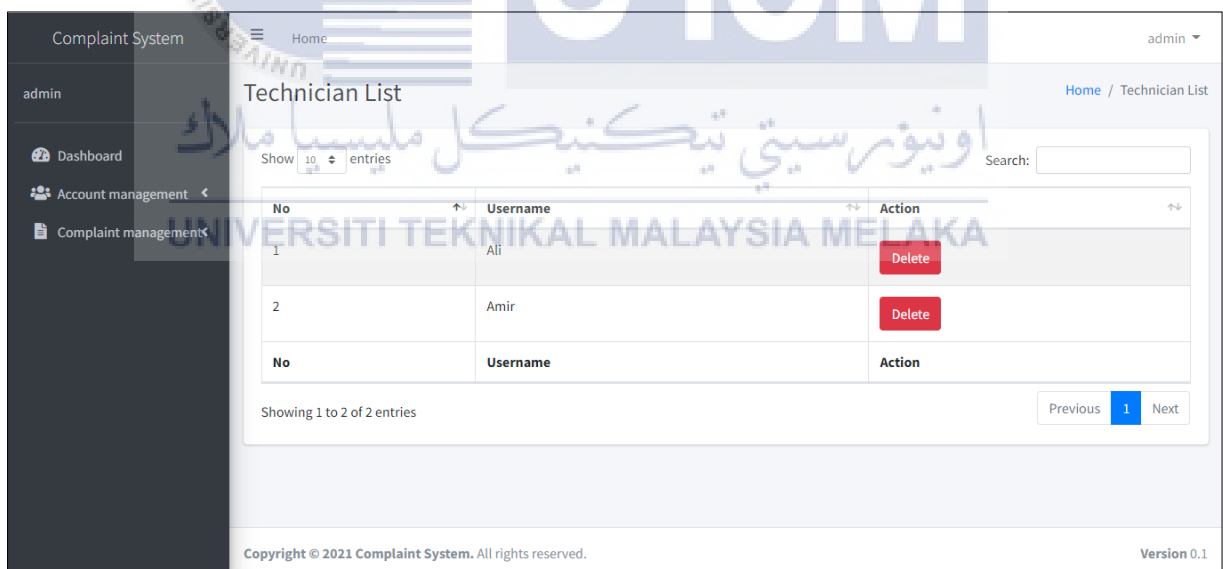


Figure 4.5 Admin Page - Technician List

Same as the user/student list, admin can also view the list of technician and delete their account. Technician account can only be registered by admin. Technician cannot register the account by themselves.

Figure 4.6 Admin Page - Add New Account

If admin click add new account, admin can register a new account and select a role which is either for technician or student.

No	User	Email	Date Complaint	Phone No	Residential	Location	Complaint Category
1	B081810492	B081810492@student.utem.edu.my	03 Jan 2022 : 11:01:02	0183597921	Satria	Lekiu SQ-Q-3-2.E	FACILITY

Problem Description Door cannot be locked

Status Assign

No	User	Email	Date Complaint	Phone No	Residential	Location	Complaint Category
Showing 1 to 1 of 1 entries							

Previous 1 Next

Copyright © 2021 Complaint System. All rights reserved. Version 0.1

Figure 4.7 Admin Page - New Complaint

For the new complaint part, admin have to assign the complaint made by students to the certain technician based on their job scope.

Assign Task

Task

Residential

Satria

Location

Lekiu SQ-Q-3-2.E

Complaint Category

FACILITY

Problem Description

Door cannot be locked

Assign To:

Technician

Amir

Figure 4.8 Admin Page - Assign Task

Based on the figure 4.7, admin can view the details of each complaint made by students and then assign the complaint to suitable technician based on the complaint category.

On-Going Complaint

Show 10 entries

Search:

No	User	Email	Date Complaint	Phone No	Residential	Location	Complaint Category
1	B081810492	B081810492@student.utem.edu.my	03 Jan 2022 : 11:01:02	0183597921	Satria	Lekiu SQ-Q-3-2.E	FACILITY

Problem Description Door cannot be locked

Status On-Going

Progress Door has been checked

Person In Charge Amir

No	User	Email	Date Complaint	Phone No	Residential	Location	Complaint Category
1	B081810492	B081810492@student.utem.edu.my	03 Jan 2022 : 11:01:02	0183597921	Satria	Lekiu SQ-Q-3-2.E	FACILITY

Showing 1 to 1 of 1 entries

Previous 1 Next

Figure 4.9 Admin Page - On-going Complaint

After admin have assigned the complaint to certain technician, technician can view the task assigned to them and update the progress. Then, the updated progress will be shown on the admin page at on-going complaint section.

No	User	Email	Date Complaint	Phone No	Residential	Location	Complaint Category
1	B081810492	B081810492@student.utem.edu.my	03 Jan 2022 : 11:01:02	0183597921	Satria	Lekiu SQ-Q-3-2.E	FACILITY

Problem Description Door cannot be locked

Status Work Done

Done By Amir

Solution Door knob has been replaced with the new one

No	User	Email	Date Complaint	Phone No	Residential	Location	Complaint Category
----	------	-------	----------------	----------	-------------	----------	--------------------

Showing 1 to 1 of 1 entries

Figure 4.10 Admin Page - Old Complaint

After technician choose ‘work done’ on the progress part, the complaint will be considered done and the complaint will be listed on the old complaint section of admin page.

Category	Count
Total Technician	2
Total User/Student	2

Figure 4.11 Admin Page – Logout

For the last part of admin page, admin can also logout and update password for the ‘admin’ account. After logout, admin will be back to the login page.

Complaint System

Home

Asliza

Report History

Home / Report History

Show 10 entries

Search:

No	Email	No Phone	Date Complaint	Residential	Location	Complaint Category	Problem Description
1	B081810492@student.utem.edu.my	0183597921	03 Jan 2022 : 06:01:38	Satria	Lekiu SQ-Q-3-2.E	TOILET	Flush cannot be used

Status: Work Done

Progress: Flush has been repaired

Action: [Edit](#) [Delete](#)

No	Email	No Phone	Date Complaint	Residential	Location	Complaint Category	Problem Description
----	-------	----------	----------------	-------------	----------	--------------------	---------------------

Showing 1 to 1 of 1 entries

Previous 1 Next

Figure 4.12 Student Page

For student page, after student has login into their account, the page will show the report history where the old complaint that the student has made. Student still can edit or delete the complaint's details.

New Complaint

Email: B081810492@student.utem.edu.my

Phone number: 0183597921

Residential: Satria

Location: Lekiu SQ-Q-3-2.E

Complaint category: ELECTRICITY

Problem Description: Lamp keep blinking

[Send Report](#)

Figure 4.13 Student Page - New Complaint

To make a new complaint, students have to fill the details in the form such as in the figure above. For the complaint category, student can choose either electricity, toilet, facility or others. Then, student can send the report to be checked by admin.

No	Email	No Phone	Date Complaint	Residential	Location	Complaint Category
1	B081810492@student.utem.edu.my	0183597921	03 Jan 2022 : 06:01:38	Satria	Lekiu SQ-Q-3-2.E	TOILET
2	B081810492@student.utem.edu.my	0183597921	03 Jan 2022 : 07:01:31	Satria	Lekiu SQ-Q-3-2.E	ELECTRICITY

Problem Description Lamp keep blinking Status Pending Progress Action Edit Delete						
--	--	--	--	--	--	--

No	Email	No Phone	Date Complaint	Residential	Location	Complaint Category
Showing 1 to 2 of 2 entries						

Figure 4.14 Student Page - Report History

After student has successfully send the report of the new complaint, student will be brought again to the report history and the new complaint will be shown with the pending status. Admin have to assign the complaint to technician first so that the technician can update the progress.

Edit Complaint	
Email	B081810492@student.utem.edu.my
No Phone	0183597921
Residential	Satria
Location	Lekiu SQ-Q-3-2.E
Complaint Category	ELECTRICITY
Problem Description	Lamp keep blinking
Update	

Figure 4.15 Student Page - Edit Complaint

Student can also edit the complaint's details as shown in the figure above and delete the complaint if they want.

Asliza		Report History						Home / Report History
Dashboard		Show 10 entries						Search:
New Complaint								
No	Email	No Phone	Date Complaint	Residential	Location	Complaint Category	Problem Description	
1	B081810492@student.utem.edu.my	0183597921	03 Jan 2022 : 06:01:38	Satria	Lekiu SQ-Q-3-2.E	TOILET	Flush cannot be used	
Status		Work Done						
Progress		Flush has been repaired						
Action		Edit Delete						
2	B081810492@student.utem.edu.my	0183597921	03 Jan 2022 : 07:01:31	Satria	Lekiu SQ-Q-3-2.E	ELECTRICITY	Lamp keep blinking	
Status		On-Going						
Progress		Check completed						
Action		Edit Delete						

Figure 4.16 Student Page - Report History After Updated

After the assigned technician has update the progress for the complaint, the status and progress will be shown as in figure 4.16. If the complaint is still in progress, it will state the status is still on-going with it progress statement. If the technician has complete the maintenance and update the status to 'Work Done', the status will be change to work done.

Complaint System		Home						Asliza
Asliza		Report History						Update Password Logout
Dashboard		Show 10 entries						Search:
New Complaint								
No	Email	No Phone	Date Complaint	Residential	Location	Complaint Category	Problem Description	
1	B081810492@student.utem.edu.my	0183597921	03 Jan 2022 : 06:01:38	Satria	Lekiu SQ-Q-3-2.E	TOILET	Flush cannot be used	
Status		Work Done						
Progress		Flush has been repaired						
Action		Edit Delete						
2	B081810492@student.utem.edu.my	0183597921	03 Jan 2022 : 07:01:31	Satria	Lekiu SQ-Q-3-2.E	ELECTRICITY	Lamp keep blinking	
Status		On-Going						
Progress		Check completed						

Figure 4.17 Student Page – Logout

As in figure 4.17, student can also logout and update their password.

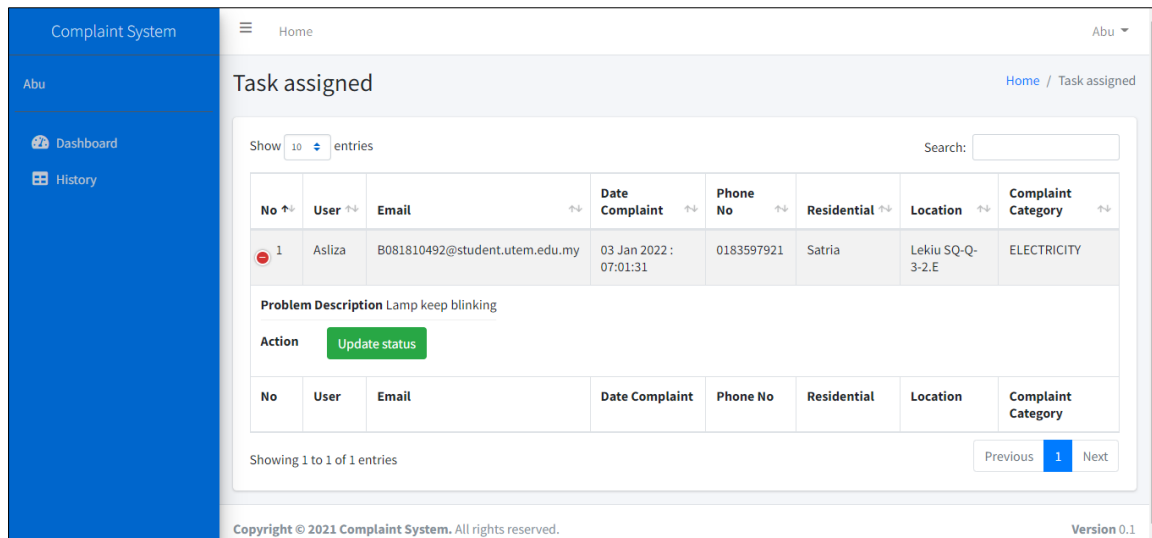


Figure 4.18 Technician Page - Task Assigned

For technician, after they has been login by using username and password given by the admin, they can see the front page with task that have been assigned for them. Then, they can read the details and upgrade the progress as many times as they want.

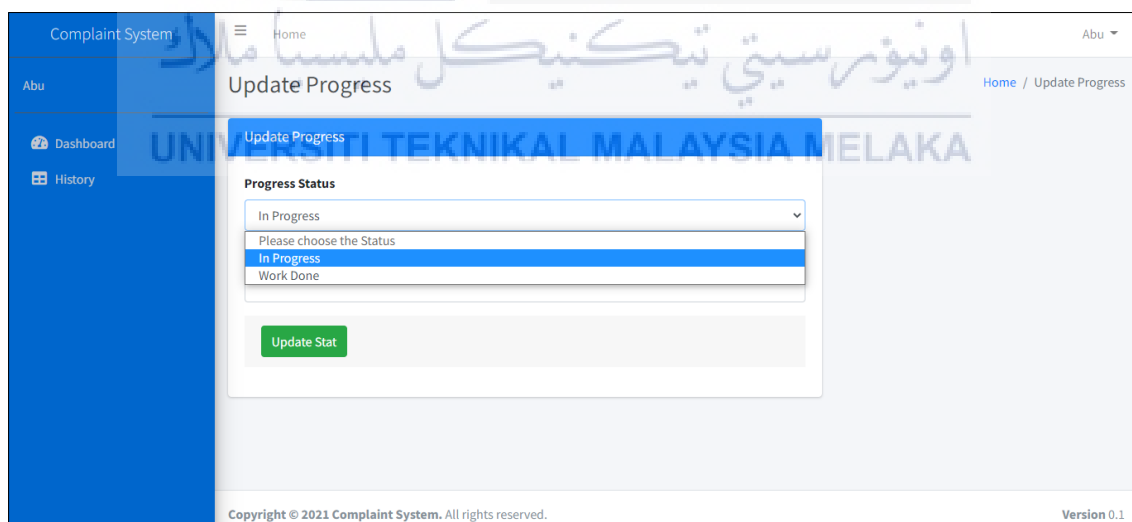


Figure 4.19 Technician Page - Update Progress

To update the progress, technician can choose between two options of the progress status which is 'In Progress' or 'Work Done'. After that, they can type the progress description in the box given. This progress description will be display on the admin and student page.

Complaint System

Home

Abu

Task Assigned History

Show 10 entries

Search:

No	User	Email	Date Complaint	Phone No	Residential	Location	Complaint Category
1	Asliza	B081810492@student.utem.edu.my	03 Jan 2022 : 07:01:31	0183597921	Satria	Lekiu SQ-Q-3-2.E	ELECTRICITY

Problem Description Lamp keep blinking
Solution Starter has been replaced with the new one

No	User	Email	Date Complaint	Phone No	Residential	Location	Complaint Category
----	------	-------	----------------	----------	-------------	----------	--------------------

Showing 1 to 1 of 1 entries

Previous 1 Next

Copyright © 2021 Complaint System. All rights reserved. Version 0.1

Figure 4.20 Technician Page - Task Assigned History

Complaint that has been updated as a 'Work Done' will be reviewed in the task assigned history section in the technician page. Technician cannot edit the progress anymore once they have chosen 'Work Done' for the progress status.

Complaint System

Home

Abu

Task assigned

Show 10 entries

Search:

No	User	Email	Date Complaint	Phone No	Residential	Location	Complaint Category	Problem Description	Action
No data available in table									

Showing 0 to 0 of 0 entries

Previous Next

Update Password

Logout

Copyright © 2021 Complaint System. All rights reserved. Version 0.1

Figure 4.21 Technician Page – Logout

Lastly, if all the tasks that have been assigned for them are completed, the task assigned table will be empty. On the other hand, technician can also logout and update their password.

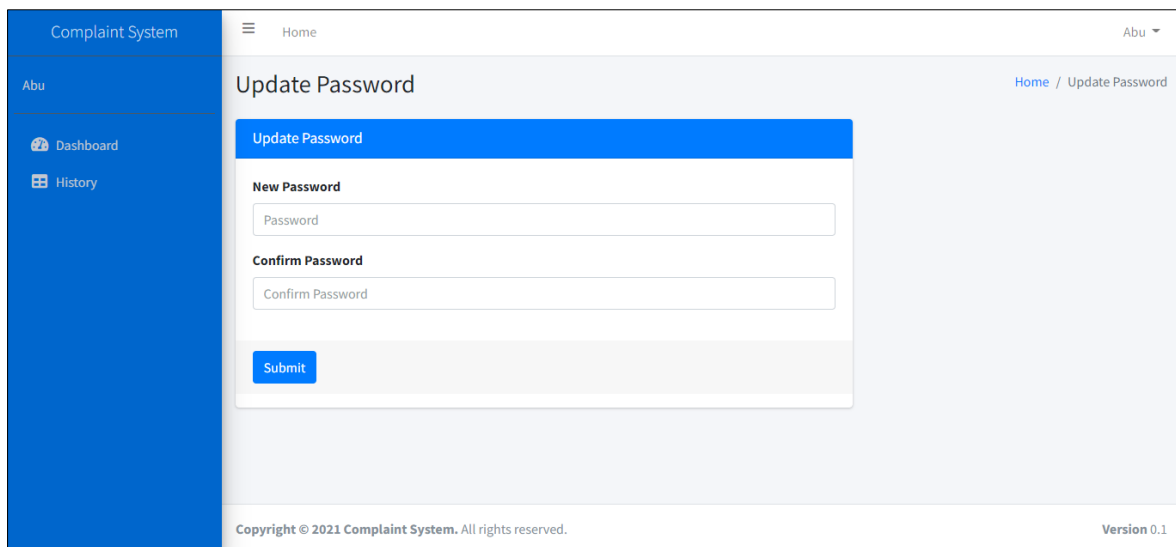


Figure 4.22 Update Password Page

The figure above is the example of update password page. All users which is admin, student and technician can change their password.

4.3 Database of The System

Before proceeding to the source codes, first and foremost, a database of the system must be created to be implemented into the codes. The database for this system has been created by using phpMyAdmin. The figure below shows the tables of 'compsys' database that have been created to be implemented in the Hostel Complaints System for UTeM. The database consists of four tables which is *acc* that stand for account, complaint, *kediaman* which represent residential and role.

Table	Action	Rows	Type
☐ acc	★ Browse Structure Search Insert Empty Drop	6	InnoDB
☐ complaint	★ Browse Structure Search Insert Empty Drop	3	InnoDB
☐ kediaman	★ Browse Structure Search Insert Empty Drop	0	InnoDB
☐ role	★ Browse Structure Search Insert Empty Drop	3	InnoDB
4 tables	Sum	12	InnoDB

Figure 4.23 Database Table

				id	username	password	role	timecreated
☐	Edit	Copy	Delete	1	admin	admin	1	2022-01-10 23:08:29
☐	Edit	Copy	Delete	8	Ali	12345	2	2022-01-10 23:08:21
☐	Edit	Copy	Delete	9	Amir	12345	2	2022-01-10 23:08:15
☐	Edit	Copy	Delete	10	B081810492	12345	3	2022-01-10 23:08:08
☐	Edit	Copy	Delete	11	Asliza	12345	3	2022-01-10 23:07:54
☐	Edit	Copy	Delete	12	Abu	12345	2	2022-01-10 23:07:40

Figure 4.24 Data for Account Table

Figure 4.24 above shows the data for account table. The table consists of five columns which is id, username, password, role and timecreated. This account table is for all the users account which is admin, students and technicians. When students register their account or when admin add new users in the admin page, the account will be added at this database table. The id will be the primary key where it will uniquely identify each row in a table.

id	id_user	email	date_of_complaint	phone_no	kolej_kediaman	Location	Complain_category	Problem_discription	status	id_technician	solution	timecreated
6	10	B081810492@student.utem.edu.my	2022-01-03 11:34:02	0183597921	Satria	Leklu SQ-Q-3-2 E	FACILITY	Door cannot be locked	4	9	Door knob has been replaced with the new one	2022-01-03 18:33:10
9	11	B081810492@student.utem.edu.my	2022-01-03 06:46:38	0183597921	Satria	Leklu SQ-Q-3-2 E	TOILET	Flush cannot be used	4	8	Flush has been repaired	2022-01-03 18:47:35
10	11	B081810492@student.utem.edu.my	2022-01-03 07:10:31	0183597921	Satria	Leklu SQ-Q-3-2 E	ELECTRICITY	Lamp keep blinking	4	12	Starter has been replaced with the new one	2022-01-03 19:55:26

Figure 4.25 Data for Complaint Table

Based on figure 4.25 above, there are three columns for complaint table which is id, id_user, email, date_of_complaint, phone_no, kolej_kediaman, location, complaint_category, problem_description, status, id_technician, solution and timecreated. This data will be used when there is new complaint made by student. Other than that, the id_technician column will be updated when admin has assigned the task to certain technician. Lastly, the solution column will also be updated when technician has update the progress for each complaint.

			id	role_discription	timecreated
Edit	Copy	Delete	1	Admin	2021-12-19 17:51:02
Edit	Copy	Delete	2	Technician	2021-12-19 17:51:02
Edit	Copy	Delete	3	User	2021-12-19 17:51:08

Figure 4.26 Data for Role Table

The last table that will be used for this system is role table. This table consists of three columns which is id, role_discription and timecreated. The role_discription column is very important as it will be used when new account has been registered by student or added by admin. This three roles represent the users of this system which is admin, user(student) and technician. The id will be linked with account table and complaint table.



4.4 Source Code of The System

In this subchapter, some of the source codes will briefly discussed and explained. Full codes will be included in the appendices.

```
78      <?php
79      $usql3= "SELECT * FROM acc where role='2' ";
80      $uresult3 = mysqli_query($conn, $usql3) or die(mysqli_error($conn));
81      $totaluser3 = mysqli_num_rows($uresult3);
82
83      ?>
84      <h3> <?php echo $totaluser3; ?><h3>
85
86      <p>Total Technician</p>
87      </div>
88      <div class="icon">
89      <i class="ion ion-person"></i>
90
91      </div>
92      </div>
93      <div class="small-box bg-info">
94      <div class="inner">
95      <?php
96      $usql3= "SELECT * FROM acc where role='3' ";
97      $uresult3 = mysqli_query($conn, $usql3) or die(mysqli_error($conn));
98      $totaluser3 = mysqli_num_rows($uresult3);
99
100     ?>
101     <h3> <?php echo $totaluser3; ?><h3>
102
103     <p>Total User/Student</p>
104     </div>
105     <div class="icon">
106     <i class="ion ion-person"></i>
107
108     </div>
109     </div>
```

Figure 4.27 Source Code of index.php for Admin

Figure 4.27 shows some of the codes of index.php for admin. Based on lines 78 until 83, its show the codes to run the SQL query/queries on table compsys.acc. Codes *role='2'* is represent the role for technician. When the SQL database can be connected with the system, it will show the total technician (line 86) at admin page. Lines 95 until 100 is to run the SQL query/queries on table compsys.acc for user since the code used is *role='3'*. This part will be display the total user/student (line 103) in admin page when the system can be connected with the SQL database.

Figure 4.28 below shows some of the codes for index.php for student where lines 75 until line 78 show the codes to run SQL query on table compsys.complaint using id_user. When the query has successfully connected with the SQL database, it will fetch all the data that have been filled by students in the complaint report. Next, codes line 93 until line 103 shows the status that will be updated by technician. There are four status were used which is 1 is for 'Pending', 2 is for 'Still in progress', 3 is for 'On-Going' or 4 is for 'Work Done'.

```

75      <?php
76      $sql = "SELECT * FROM `complaint` Where id_user='$_idu' ";
77      $result = $conn->query($sql);
78      $count=1;
79
80      while($row = $result->fetch_assoc()) {
81      ?>
82      <tr class="tr-shadow">
83          <td class="desc"><?php echo $count;?></td>
84          <td class="desc"><?php echo $row["email"];?></td>
85          <td class="desc"><?php echo $row["phone_no"];?></td>
86          <td class="desc"><?php echo date ( "d M Y : h:m:s" , strtotime ( $row["date_of_complaint"] ))?></td>
87          <td class="desc"><?php echo $row["kolej_kediaman"];?></td>
88          <td class="desc"><?php echo $row["Location"];?></td>
89          <td class="desc"><?php echo $row["Complain_category"];?></td>
90          <td class="desc"><?php echo $row["Problem_discription"];?></td>
91          <td class="desc"><?php
92
93          if($row["status"]==1)
94          {
95              echo "Pending";
96          }elseif($row["status"]==2)
97          {
98              echo "Still in progress";
99          }elseif($row["status"]==3)
100          {
101              echo "On-Going";
102          }else{
103              echo "Work Done";
104

```

Figure 4.28 Source Code of index.php for Student

```

74      <?php
75      $sql = "SELECT * FROM `complaint` where id_technician='$_idu' and status='2'or status='3'";
76      $result = $conn->query($sql);
77      $count=1;
78
79      while($row = $result->fetch_assoc()) {
80      ?>
81      <tr class="tr-shadow">
82          <td class="desc"><?php echo $count;?></td>
83          <td class="desc"><?php $uid= $row["id_user"];
84
85          $qry4="SELECT * FROM acc WHERE id='$_uid'";
86          $result4=mysqli_query($conn,$qry4);
87          $row4=mysqli_fetch_array($result4,MYSQLI_ASSOC);
88          {
89              echo $row4['username'];
90          }
91          ?></td>
92
93          <td class="desc"><?php echo $row["email"];?></td>
94          <td class="desc"><?php echo date ( "d M Y : h:m:s" , strtotime ( $row["date_of_complaint"] ))?></td>
95          <td class="desc"><?php echo $row["phone_no"];?></td>
96          <td class="desc"><?php echo $row["kolej_kediaman"];?></td>
97          <td class="desc"><?php echo $row["Location"];?></td>
98          <td class="desc"><?php echo $row["Complain_category"];?></td>
99          <td class="desc"><?php echo $row["Problem_discription"];?></td>
100          <td class="desc">
101
102          <a href="updatestat.php?idr=<?php echo $row['id']; ?>&idu=<?php echo $idu; ?>" >button type="button" class
            ="btn btn-success" >Update status</button>

```

Figure 4.29 Source Code of index.php for Technician

Figure 4.29 shows some of the codes in index.php for technician. When technician want to update status of the task which is 'Still in progress' or 'On-Going', the database in complaint table will use id_technician and status columns and connect it with id variable in account table. Hence, it will fetch all the complaint's details from the required username.

```
1 <?php
2 $servername = "localhost";
3 $username = "root";
4 $password = "";
5 $database = "compsys";
6
7 // Create connection
8 $conn = mysqli_connect($servername, $username, $password,$database);
9
10 // Check connection
11 if (!$conn){
12     die("Connection failed: " . mysqli_connect_error());
13 }
14 //echo "Connected successfully";
15 ?>
```

Figure 4.30 Source Code of db.php

The db.php is really essential since all of other php codes for this system require to include this part at the beginning of the codes which is '*include 'process/db.php';*'. If we forgot to include this codes for each php file, the database cannot be connected and the system will be error when we run it. The servername at line 2 shows the server that we use for this system which is localhost. To log in, we need both username and password. The database name that was built in this system is compsys.

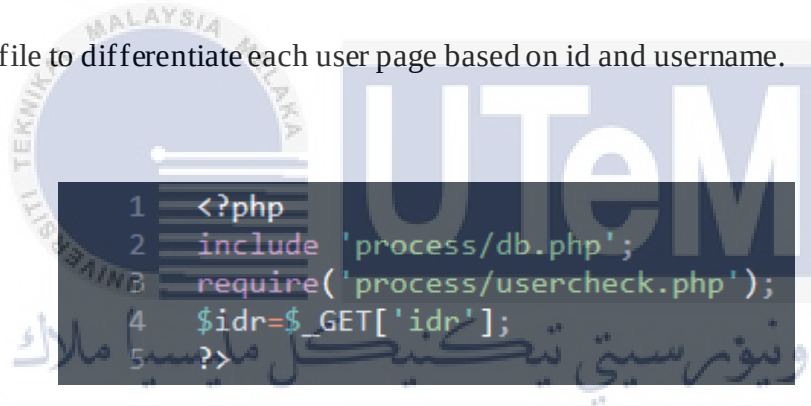
```

1  <?php
2  session_start();
3  include ("db.php");
4  $uname=$_SESSION['username'];
5  $qry="SELECT * FROM acc WHERE username='$uname'";
6  $result=mysqli_query($conn,$qry);
7  $row=mysqli_fetch_array($result,MYSQLI_ASSOC);
8  {
9  $idu=$row['id'];
10 $name=$row['username'];
11 }
12 if(!isset($uname)!=''){
13 echo "error ek";
14 header("location: ../index.php");
15 }
16 ?>

```

Figure 4.31 Source Code of usercheck.php

Besides db.php, usercheck.php is also one of required source code that should be included in each php file to differentiate each user page based on id and username.



```

1  <?php
2  include 'process/db.php';
3  require('process/usercheck.php');
4  $idr=$_GET['idr'];
5  ?>

```

Figure 4.32 Source Code for Connect with db.php and usercheck.php

Figure 4.32 shows the source code used in the beginning of each php file that must be included so that the system can be connected with SQL database in phpMyAdmin and all the users can function as required.

4.5 Summary

This chapter explains the design of the Hostel Complaints System for UTeM and how to use it. Since this system is work without a domain, so we need to make sure we have XAMPP software and the related files with its database. Then, we must start the Apache and MySQL using the XAMPP software. By doing this, we can run the system smoothly using localhost. In other words, everyone that want to access this system must have the files and database. Without it, they cannot run this system at all. Thus, to make it easier for everyone to use this system, this system should be built with its domain and online server so that it can be used by everyone through the website link.



CHAPTER 5

CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

This system should be implemented by UTeM hostel admin since they are still using a manual way for students to make complaint. There are some advantages of this system which is this system can help student to make a complaint easier and student can also view the status of the complaints that have been submitted to the system. In this way, student don't need to go to the administrative office to take a complaint form and fill it. This system also helps the technicians to view all the complaints that have been assigned for them properly. Moreover, this system also helps to prevent the losing of data because all the complaints that have been submit by the complainant will be keep directly to the database and this can help admin to manage all of the complaints properly without any data loss problem.

5.2 Future Works

For future improvements, this system can be implemented as follows:

- i) Student can include images in their complaint form so that admin and technician can see the problem through the complaint form and technician can do the maintenance faster.
- ii) This system should be implemented for mobile applications so this system can be used for both Android and IOS.
- iii) This system should be upgrade to add a notification and alert functions for all the technician after each new complaint that have been received.

REFERENCES

- [1] M. Saifulnizam and M. Jaini, "Think Hostel Maintenance System (THMS)," vol. 1.
- [2] M. S. Alve, M. V. Babardesai, M. S. Bhosale, M. S. Kapadi, and P. A. B, "Web Application for Complaint Tracking and Resolving," *Int. Res. J. Eng. Technol.*, vol. 4, no. 4, 2017, [Online]. Available: <https://www.irjet.net/archives/V4/i4/IRJET-V4I4289.pdf>.
- [3] M. A. Rosid, A. Rachmadany, M. T. Multazam, A. B. D. Nandiyanto, A. G. Abdullah, and I. Widiaty, "Integration Telegram Bot on E-Complaint Applications in College," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 288, no. 1, 2018, doi: 10.1088/1757-899X/288/1/012159.
- [4] R. K. Bista, A. J. Karki, and S. Das, "Hostel Management System with RFID," *Int. J. Trend Sci. Res. Dev.*, vol. Volume-2, no. Issue-4, pp. 856–862, 2018, doi: 10.31142/ijtsrd14110.
- [5] P. Kormpho, P. Liawsomboon, N. Phongoen, and S. Pongpaichet, "Smart Complaint Management System," *Proceeding 2018 7th ICT Int. Student Proj. Conf. ICT-ISPC 2018*, 2018, doi: 10.1109/ICT-ISPC.2018.8523949.
- [6] N. Illias, N. H. Abdul Hamid, and Z. A. Shaffiei, "The android based mobile application to manage student complaints," *Bull. Electr. Eng. Informatics*, vol. 9, no. 3, pp. 1276–1283, 2020, doi: 10.11591/eei.v9i3.2065.
- [7] P. Nagpal, N. Goel, S. Sangwan, and H. Dixit, "Design and Implementation of Hostel Management System Using Java and Mysql," pp. 63–74, 2020.

APPENDICES

Appendix A Source Code of index.php for Admin

```
<?php
require('process/db.php');
require('process/usercheck.php');

?>
<html lang="en">
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <?php include "title.php"; ?>

  <!-- Google Font: Source Sans Pro -->
  <link rel="stylesheet"
href="https://fonts.googleapis.com/css?family=Source+Sans+Pro:300,400,400i,700&display=fallback">
  <!-- Font Awesome -->
  <link rel="stylesheet" href="plugins/fontawesome-free/css/all.min.css">
  <!-- Ionicons -->
  <link rel="stylesheet"
href="https://code.ionicframework.com/ionicons/2.0.1/css/ionicons.min.css">
  <!-- Tempusdominus Bootstrap 4 -->
  <link rel="stylesheet" href="plugins/tempusdominus-bootstrap-4/css/tempusdominus-bootstrap-4.min.css">
  <!-- iCheck -->
  <link rel="stylesheet" href="plugins/ichack-bootstrap/ichack-bootstrap.min.css">
  <!-- JQVMap -->
  <link rel="stylesheet" href="plugins/jqvmap/jqvmap.min.css">
  <!-- Theme style -->
  <link rel="stylesheet" href="dist/css/adminlte.min.css">
  <!-- overlayScrollbars -->
  <link rel="stylesheet"
href="plugins/overlayScrollbars/css/OverlayScrollbars.min.css">
  <!-- Daterange picker -->
  <link rel="stylesheet" href="plugins/daterangepicker/daterangepicker.css">
  <!-- summernote -->
  <link rel="stylesheet" href="plugins/summernote/summernote-bs4.min.css">
</head>
<body class="hold-transition sidebar-mini layout-fixed">
<div class="wrapper">

  <!-- Navbar -->
```

```

<?php include "nav.php"; ?>

<!-- /.navbar -->

<!-- Main Sidebar Container -->
<?php include "sidebarmenu.php"; ?>

<!-- Content Wrapper. Contains page content -->
<div class="content-wrapper">
  <!-- Content Header (Page header) -->
  <div class="content-header">
    <div class="container-fluid">
      <div class="row mb-2">
        <div class="col-sm-6">
          <h1 class="m-0 text-dark">Dashboard</h1>
        </div><!-- /.col -->
        <div class="col-sm-6">
          <ol class="breadcrumb float-sm-right">
            <li class="breadcrumb-item"><a href="#">Home</a></li>
            <li class="breadcrumb-item active">Dashboard</li>
          </ol>
        </div><!-- /.col -->
      </div><!-- /.row -->
    </div><!-- /.container-fluid -->
  </div>
  <!-- /.content-header -->

  <!-- Main content -->
  <section class="content">
    <div class="container-fluid">
      <!-- Small boxes (Stat box) -->

      <!-- /.row -->
      <!-- Main row -->

      <div class="small-box bg-info">
        <div class="inner">
          <?php
            $usql3= "SELECT * FROM acc where role='2' ";
            $uresult3 = mysqli_query($conn, $usql3) or die(mysqli_error($conn));
            $totaluser3 = mysqli_num_rows($uresult3);

            ?>
            <h3> <?php echo $totaluser3; ?></h3>
            <p>Total Technician</p>
          </div>
          <div class="icon">
            <i class="ion ion-person"></i>

```

```

    </div>
  </div>
  <div class="small-box bg-info">
    <div class="inner">
      <?php
        $usql3= "SELECT * FROM acc where role='3'";
        $uresult3 = mysqli_query($conn, $usql3) or die(mysqli_error($conn));
        $totaluser3 = mysqli_num_rows($uresult3);

      ?>
      <h3> <?php echo $totaluser3; ?><h3>

      <p>Total User/Student</p>
    </div>
    <div class="icon">
      <i class="ion ion-person"></i>
    </div>
  </div>
</div>
<!-- /.row (main row) -->
</div><!-- /.container-fluid -->
</section>
<!-- /.content -->
</div>
<!-- /.content-wrapper -->
<?php include "footer.php";?>

<!-- Control Sidebar -->
<aside class="control-sidebar control-sidebar-dark">
  <!-- Control sidebar content goes here -->
</aside>
<!-- /.control-sidebar -->
</div>
<!-- /.wrapper -->

<!-- jQuery -->
<script src="plugins/jquery/jquery.min.js"></script>
<!-- jQuery UI 1.11.4 -->
<script src="plugins/jquery-ui/jquery-ui.min.js"></script>
<!-- Resolve conflict in jQuery UI tooltip with Bootstrap tooltip -->
<script>
  $.widget.bridge('uibutton', $.ui.button)
</script>
<!-- Bootstrap 4 -->
<script src="plugins/bootstrap/js/bootstrap.bundle.min.js"></script>
<!-- ChartJS -->

```

```

<script src="plugins/chart.js/Chart.min.js"></script>
<!-- Sparkline -->
<script src="plugins/sparklines/sparkline.js"></script>
<!-- JQVMap -->
<script src="plugins/jqvmap/jquery.vmap.min.js"></script>
<script src="plugins/jqvmap/maps/jquery.vmap.usa.js"></script>
<!-- jQuery Knob Chart -->
<script src="plugins/jquery-knob/jquery.knob.min.js"></script>
<!-- daterangepicker -->
<script src="plugins/moment/moment.min.js"></script>
<script src="plugins/daterangepicker/daterangepicker.js"></script>
<!-- Tempusdominus Bootstrap 4 -->
<script src="plugins/tempusdominus-bootstrap-4/js/tempusdominus-bootstrap-4.min.js"></script>
<!-- Summernote -->
<script src="plugins/summernote/summernote-bs4.min.js"></script>
<!-- overlayScrollbars -->
<script
src="plugins/overlayScrollbars/js/jquery.overlayScrollbars.min.js"></script>
<!-- AdminLTE App -->
<script src="dist/js/adminlte.js"></script>
<!-- AdminLTE for demo purposes -->
<script src="dist/js/demo.js"></script>
<!-- AdminLTE dashboard demo (This is only for demo purposes) -->
<script src="dist/js/pages/dashboard.js"></script>
</body>
</html>

```

Appendix B Source Code of index.php for Student

```

<?php
include 'process/db.php';
require('process/usercheck.php');
?>
<html lang="en">
<head>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<?php include "title.php"; ?>
<!-- Google Font: Source Sans Pro -->

<link rel="stylesheet"
href="https://fonts.googleapis.com/css?family=Source+Sans+Pro:300,400,400i,700&display=fallback">
<!-- Font Awesome -->
<link rel="stylesheet" href="plugins/fontawesome-free/css/all.min.css">
<!-- DataTables -->

```

```

<link rel="stylesheet" href="plugins/datatables-
bs4/css/dataTables.bootstrap4.min.css">
<link rel="stylesheet" href="plugins/datatables-
responsive/css/responsive.bootstrap4.min.css">
<!-- Theme style -->
<link rel="stylesheet" href="dist/css/adminlte.min.css">
</head>
<body class="hold-transition sidebar-mini">
<div class="wrapper">
<!-- Navbar -->
<?php include "nav.php"; ?>
<!-- /.navbar -->

<!-- Main Sidebar Container -->
<?php include "sidebarmenu.php"; ?>

<!-- Content Wrapper. Contains page content -->
<div class="content-wrapper">
<!-- Content Header (Page header) -->
<section class="content-header">
<div class="container-fluid">
<div class="row mb-2">
<div class="col-sm-6">
<h1>Report History</h1>
</div>
<div class="col-sm-6">
<ol class="breadcrumb float-sm-right">
<li class="breadcrumb-item"><a href="index.php">Home</a></li>
<li class="breadcrumb-item active">Report History</li>
</ol>
</div>
</div>
</div><!-- /.container-fluid -->
</section>

<!-- Main content -->
<section class="content">
<div class="container-fluid">
<div class="row">
<div class="col-12">
<div class="card">
<!-- /.card-header -->

<div class="card-body">
<table id="example1" class="table table-bordered table-striped">
<thead>
<tr>
<th>No</th>
<th>Email</th>

```

```

<th>No Phone</th>
<th>Date Complaint</th>
<th>Residential</th>
<th>Location</th>
<th>Complaint Category</th>
<th>Problem Description</th>
<th>Status</th>
<th>Progress</th>
<th>Action</th>

</tr>
</thead>
<tbody>
    <?php
        $sql = "SELECT * FROM `complaint` Where id_user='$idu' ";
        $result = $conn->query($sql);
        $count=1;

        while($row = $result->fetch_assoc()) {
            ?>
            <tr class="tr-shadow">
                <td class="desc"><?php echo $count;?></td>
                <td class="desc"><?php echo $row["email"];?></td>
                <td class="desc"><?php echo $row["phone_no"];?></td>
                <td class="desc"><?php echo date ( "d M Y : h:m:s" , strtotime (
$row["date_of_complaint"] ) );?></td>
                <td class="desc"><?php echo $row["kolej_kediaman"];?></td>
                <td class="desc"><?php echo $row["Location"];?></td>
                <td class="desc"><?php echo $row["Complain_category"];?></td>
                <td class="desc"><?php echo
$row["Problem_discription"];?></td>
                <td class="desc"><?php

                    if($row["status"]==1)
                    {
                        echo "Pending";
                    }elseif($row["status"]==2)
                    {
                        echo "Still in progress";
                    }elseif($row["status"]==3)
                    {
                        echo "On-Going";
                    }
                    else{
                        echo "Work Done";
                    }

                ?></td>
                <td class="desc"><?php echo $row["solution"];?></td>

```

```

        <td class="desc">
            <a href="edit_comp.php?id=?php echo $row["id"];?>"><button
type="button" class="btn btn-primary">Edit</button></a>

            <a href="process/deletecomp.php?id=?php echo
$row["id"];?>"><button type="button" class="btn btn-
danger">Delete</button></a>

        </td>

    </tr>
</?php

$count+=1;}
$conn->close();?>

</tr>
</tbody>

<tfoot>
<tr>
    <th>No</th>
<th>Email</th>
<th>No Phone</th>
<th>Date Complaint</th>
<th>Residential</th>
<th>Location</th>
<th>Complaint Category</th>
<th>Problem Description</th>
<th>Status</th>
<th>Progress</th>
<th>Action</th>

</tr>
</tfoot>
</table>
</div>
<!-- /.card-body -->
</div>
<!-- /.card -->
</div>
<!-- /.col -->
</div>

<!-- /.row -->
</div>
<!-- /.container-fluid -->
</section>
<!-- /.content -->
</div>

```



```

<!-- /.content-wrapper -->
<?php include "footer.php"; ?>

<!-- Control Sidebar -->
<aside class="control-sidebar control-sidebar-dark">
  <!-- Control sidebar content goes here -->
</aside>
<!-- /.control-sidebar -->
</div>
<!-- ./wrapper -->

<!-- jQuery -->
<script src="plugins/jquery/jquery.min.js"></script>
<!-- Bootstrap 4 -->
<script src="plugins/bootstrap/js/bootstrap.bundle.min.js"></script>
<!-- DataTables -->
<script src="plugins/datatables/jquery.dataTables.min.js"></script>
<script src="plugins/datatables-bs4/js/dataTables.bootstrap4.min.js"></script>
<script src="plugins/datatables-responsive/js/dataTables.responsive.min.js"></script>
<script src="plugins/datatables-responsive/js/responsive.bootstrap4.min.js"></script>
<!-- AdminLTE App -->
<script src="dist/js/adminlte.min.js"></script>
<!-- AdminLTE for demo purposes -->
<script src="dist/js/demo.js"></script>
<!-- Page specific script -->
<script>
$(function () {
  $("#example1").DataTable({
    "responsive": true,
    "autoWidth": false,
  });
  $('#example2').DataTable({
    "paging": true,
    "lengthChange": false,
    "searching": false,
    "ordering": true,
    "info": true,
    "autoWidth": false,
    "responsive": true,
  });
});
</script>
</body>
</html>

```

Appendix C Source Code of index.php for Technician

```
<?php
include 'process/db.php';
require('process/usercheck.php');
?>
<html lang="en">
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <?php include "title.php"; ?>

  <!-- Google Font: Source Sans Pro -->
  <link rel="stylesheet"
href="https://fonts.googleapis.com/css?family=Source+Sans+Pro:300,400,400i,700&display=fallback">
  <!-- Font Awesome -->
  <link rel="stylesheet" href="plugins/fontawesome-free/css/all.min.css">
  <!-- DataTables -->
  <link rel="stylesheet" href="plugins/datatables-bs4/css/dataTables.bootstrap4.min.css">
  <link rel="stylesheet" href="plugins/datatables-responsive/css/responsive.bootstrap4.min.css">
  <!-- Theme style -->
  <link rel="stylesheet" href="dist/css/adminlte.min.css">
</head>
<body class="hold-transition sidebar-mini">
<div class="wrapper">
  <!-- Navbar -->
  <?php include "nav.php"; ?>
  <!-- /.navbar -->

  <!-- Main Sidebar Container -->
  <?php include "sidebarmenu.php"; ?>

  <!-- Content Wrapper. Contains page content -->
  <div class="content-wrapper">
    <!-- Content Header (Page header) -->
    <section class="content-header">
      <div class="container-fluid">
        <div class="row mb-2">
          <div class="col-sm-6">

            <h1>Task assigned</h1>
          </div>
          <div class="col-sm-6">
            <ol class="breadcrumb float-sm-right">
              <li class="breadcrumb-item"><a href="index.php">Home</a></li>
            </ol>
          </div>
        </div>
      </div>
    </section>
  </div>
</div>
```

```

        <li class="breadcrumb-item active">Task assigned</li>
    </ol>
</div>
</div>
</div><!-- /.container-fluid -->
</section>

<!-- Main content -->
<section class="content">
    <div class="container-fluid">
        <div class="row">
            <div class="col-12">
                <div class="card">
                    <!-- /.card-header -->
                    <div class="card-body">
                        <table id="example1" class="table table-bordered table-striped">
                            <thead>
                                <tr>
                                    <th>No</th>
                                    <th>User</th>
                                    <th>Email</th>
                                    <th>Date Complaint</th>
                                    <th>Phone No</th>
                                    <th>Residential</th>
                                    <th>Location</th>
                                    <th>Complaint Category</th>
                                    <th>Problem Description</th>
                                    <th>Action</th>
                                </tr>
                            </thead>
                            <tbody>
                                <?php
                                    $sql = "SELECT * FROM `complaint` where id_technician='$idu' and
status='2'or status='3'";
                                    $result = $conn->query($sql);
                                    $count=1;

                                    while($row = $result->fetch_assoc()) {
                                        ?>
                                        <tr class="tr-shadow">
                                            <td class="desc"><?php echo $count;?></td>
                                            <td class="desc"><?php $uid= $row["id_user"];

                                            $qry4="SELECT * FROM acc WHERE id='$uid'";
                                            $result4=mysqli_query($conn,$qry4);
                                            $row4=mysqli_fetch_array($result4,MYSQLI_ASSOC);
                                            {
                                                echo $row4['username'];

```

```

    }
    ?></td>

    <td class="desc"><?php echo $row["email"];?></td>
    <td class="desc"><?php echo date ( "d M Y : h:m:s" , strtotime (
$row["date_of_complaint"] ));?></td>
    <td class="desc"><?php echo $row["phone_no"];?></td>
    <td class="desc"><?php echo $row["kolej_kediaman"];?></td>
    <td class="desc"><?php echo $row["Location"];?></td>
    <td class="desc"><?php echo $row["Complain_category"];?></td>
    <td class="desc"><?php echo
$row["Problem_discription"];?></td>
    <td class="desc">

    <a href="updatestat.php?idr=<?php echo $row['id'];
?>&idu=<?php echo $idu; ?>"> <button type="button" class="btn btn-success"
>Update status</button>

    </td>
  </tr>
<?php
$count+=1;}
$conn->close();?>
</tr>
</tbody>

<tfoot>
<tr>
  <th>No</th>
<th>User</th>
<th>Email</th>
<th>Date Complaint</th>
<th>Phone No</th>
<th>Residential</th>
<th>Location</th>
<th>Complaint Category</th>
<th>Problem Description</th>
<th>Status</th>
</tr>
</tfoot>
</table>
</div>

<!-- /.card-body -->
</div>
<!-- /.card -->
</div>
<!-- /.col -->

```

```

        </div>
        <!-- /.row -->
    </div>
    <!-- /.container-fluid -->
</section>
<!-- /.content -->
</div>
<!-- /.content-wrapper -->
<?php include "footer.php"; ?>
<!-- Control Sidebar -->
<aside class="control-sidebar control-sidebar-dark">
    <!-- Control sidebar content goes here -->
</aside>
<!-- /.control-sidebar -->
</div>
<!-- ./wrapper -->

<!-- jQuery -->
<script src="plugins/jquery/jquery.min.js"></script>
<!-- Bootstrap 4 -->
<script src="plugins/bootstrap/js/bootstrap.bundle.min.js"></script>
<!-- DataTables -->
<script src="plugins/datatables/jquery.dataTables.min.js"></script>
<script src="plugins/datatables-bs4/js/dataTables.bootstrap4.min.js"></script>
<script src="plugins/datatables-responsive/js/dataTables.responsive.min.js"></script>
<script src="plugins/datatables-responsive/js/dataTables.bootstrap4.min.js"></script>
<!-- AdminLTE App -->
<script src="dist/js/adminlte.min.js"></script>
<!-- AdminLTE for demo purposes -->
<script src="dist/js/demo.js"></script>
<!-- Page specific script -->
<script>
$(function () {
    $('#example1').DataTable({
        "responsive": true,
        "autoWidth": false,
    });
    $('#example2').DataTable({
        "paging": true,
        "lengthChange": false,
        "searching": false,
        "ordering": true,
        "info": true,
        "autoWidth": false,
        "responsive": true,
    });
});

```

```
});
</script>
</body>
</html>
```

Appendix D Source Code of db.php

```
<?php
$servername = "localhost";
$username = "root";
$password = "";
$database = "compsys";

// Create connection
$conn = mysqli_connect($servername, $username, $password,$database);

// Check connection
if (!$conn) {
    die("Connection failed: " . mysqli_connect_error());
}
//echo "Connected successfully";
?>
```

Appendix E Source Code of usercheck.php

```
<?php
session_start();
include ("db.php");
$username=$_SESSION['username'];
$query="SELECT * FROM acc WHERE username='$username'";
$result=mysqli_query($conn,$query);
$row=mysqli_fetch_array($result,MYSQLI_ASSOC);
{
    $id=$row['id'];
    $name=$row['username'];
}
if(!isset($username)!=""){
    echo "error ek";
    header("location: ../index.php");
}
?>
```