HIBAH MANAGEMENT SYSTEM

MUHAMMAD SYAHMI BIN ABDUL JALIL

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

HIBAH MANAGEMENT SYSTEM

MUHAMMAD SYAHMI BIN ABDUL JALIL

This report is submitted in partial fulfillment of the requirements for the
Bachelor of Computer Science (Software Development) with Honours.
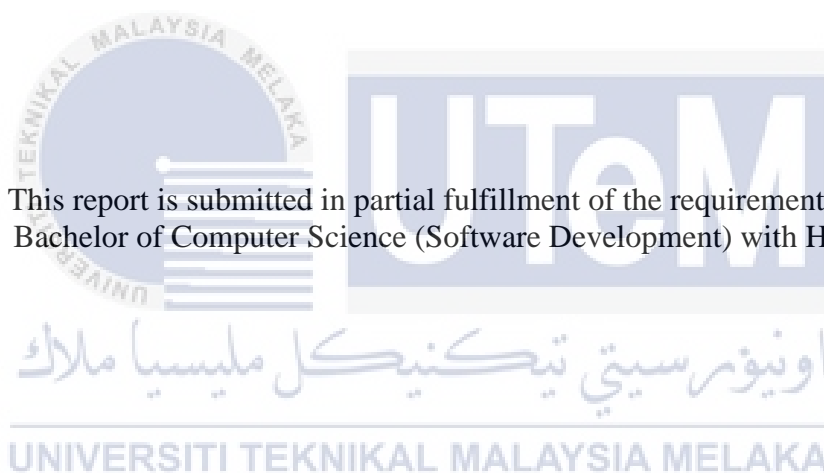
FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2021

DECLARATION

I hereby declare that this project report entitled

HIBAH MANAGEMENT SYSTEM

is written by me and is my own effort and that no part has been plagiarized

without citations.

STUDENT: Muhammad Syahmi Bin Abdul Jalil   Date : 9 September 2021

I hereby declare that I have read this project report and found

this project report is sufficient in term of the scope and quality for the award of

Bachelor of Computer Science (Software Development) with Honours.

SUPERVISOR: Madam Zarita Binti Mohd Kosnin   Date : 9 September 2021

DEDICATION

This study is wholeheartedly dedicated to my beloved parents, who have been my source of inspiration and gave me strength when I thought of giving up, who continually provide their moral, spiritual, emotional, and financial support.

To my brothers, sisters, relative, teachers, friends, and classmates who shared their words of advice and encouragement to finish this study.

And lastly, I dedicated this work to the Almighty Allah. Thank you for the guidance, strength, power of the mind, protection, and skills for giving me a healthy life.

ACKNOWLEDGEMENTS

ABSTRACT

Hibah Management System is an entirely web-based application that was initially developed to assist and ease the execution of the hibah process from the online application to the operation of the application by the Majlis Agama Islam Negeri Sembilan staff. This project development's primary purpose is to systematically convert manual hibah system to computerize and manage hibah processes. Such as updating current hibah status, managing their documents, generating documents and client information efficiently, and tracking all hibah history with ease. Lastly, make the hibah process is more organized and user-friendly. With the implementation of a payment gateway, it is easier for clients to make the application online and staff easier to manage. This system provides an excellent report feature. As a result, Hibah Management System has been developed to overcome the problem faced by Majlis Agama Islam Negeri Sembilan. Interviews are used to gather the information for Hibah Management System development. As a result, the system can help Majlis Agama Islam Negeri Sembilan simplify and process their business.

ABSTRAK

Sistem Pengurusan Hibah adalah aplikasi berasaskan web sepenuhnya yang pada mulanya dibangunkan untuk membantu dan memudahkan pelaksanaan proses hibah dari membuat permohonan atas talian sehinggalah pengoperasian aplikasi oleh kakitangan Majlis Agama Islam Negeri Sembilan. Tujuan utama pembangunan sistem ini adalah untuk menukarkan sistem hibah secara manual kepada komputerisasi dan juga menguruskan proses hibah secara komputerisasi. Contoh proses yang ditukar ke komputerisasi adalah mengemas kini status hibah semasa, menguruskan dokumen permohonan, menghasilkan dokumen secara atas talian dan maklumat pelanggan dengan cekap, dan mengesan semua sejarah hibah dengan mudah. Akhir sekali, ia akan menjadikan proses hibah lebih teratur dan mesra pengguna. Dengan pelaksanaan gerbang pembayaran juga lebih mudahkan bagi para pelanggan untuk membuat permohonan hibah dalam talian dan yang paling utama, kakitangan lebih mudah untuk menguruskan sesuatu permohonan. Sistem ini mampu memberikan ciri laporan yang sangat baik. Hasilnya, Sistem Pengurusan Hibah telah dibangunkan untuk mengatasi masalah yang dihadapi oleh Majlis Agama Islam Negeri Sembilan. Temu ramah digunakan untuk mengumpulkan maklumat untuk pembangunan Sistem Pengurusan Hibah. Oleh itu, sistem ini dapat membantu Majlis Agama Islam Negeri Sembilan mempermudahkan dan memproses perniagaan mereka.
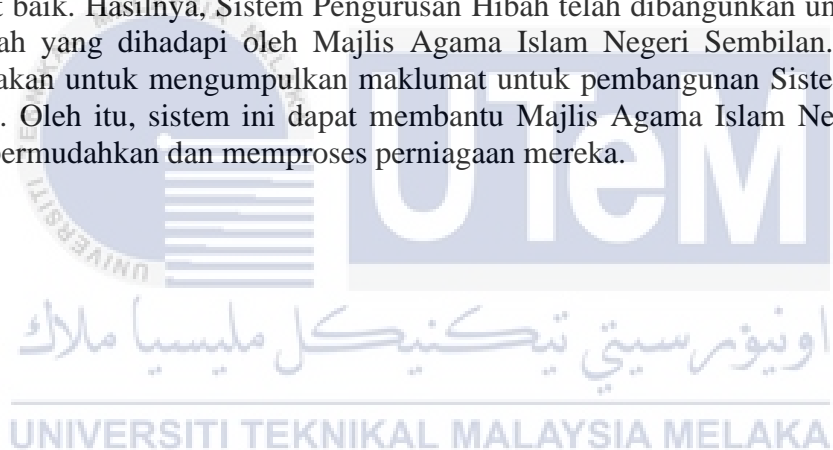
TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

FYP   -   **Final Year Project**

HMS   -   **Hibah Management System**

LIST OF ATTACHMENTS

CHAPTER 1: INTRODUCTION

1.1 Introduction

Hibah is the granting ownership of the property from one party to another without any consideration that occurs during the life of a hibah provider, made voluntarily, not meant to glorify anybody, and given by reciting an ijab and qabul or any such expressions. In Majlis Agama Islam Negeri Sembilan, an increasing number of people want to make hibah, either from Negeri Sembilan or another state. Typically, Majlis Agama Islam Negeri Sembilan operates using the manual system. The client needs to fill the documents and need to submit them to the Department of Hibah. Although these methods work well, they are not very efficient because they do not have a systematic workflow to facilitate the work of staff and clients, which will often redo the submission due to mistakes when receiving or making the hibah. With Hibah Management System, the client will be able to make and managing the hibah application efficiently. Staffs can manage the hibah with a very systematic flow. This system provides functions that can help them to improve their business process.

1.2 Problem Statements

Department Hibah and Wasiat Majlis Agama Islam Negeri Sembilan are having trouble taking or managing the hibah document and knowing the client's application status. Some issues exist where the person who conducts the process makes mistakes

or human errors, such as data loss or unchecking documents. With the standard procedure, they need to check each record before proceeding to other processes. This process involves many steps before the hibah process finishes. So, the percentage of human error for each process increase. We can reduce the problem if they have an excellent system to run their operation. They are facing a problem to produce an excellent informative report. With immense client's document, they need more time and regularly check all documents before creating a report.

1.3 Objective

The main purpose of the Hibah management system is to overcome the problem forced by the Department Hibah dan Wasiat Majlis Agama Islam Negeri Sembilan. In distinctive, the objectives of Hibah Management System.

- To make the hibah process is more organized and user-friendly. Clients easier to make the application online and staffs easier to manage the applications.

- To provide a system that have excellent report feature.

1.4 Scope

This project is for the creation of a Hibah Management System for a web-based application. The system develops for Department of Hibah and Wasiat Majlis Agama Islam Negeri Sembilan. There are three types of users in this system, which are admin, staff, and client.

1.4.1 Module

1.4.1.1 Admin and Staff

- User management

    a. Admin can create an account for staff, and the system will send an email with the username and password to the staff. For the first-time login, they need to change the password.

    b. Admin can manage the user.

    c. Admin and staff able to change their profile information.

    d. Forgot password feature by using email. The system will send a link to the user's email to change their password.

- Client management

    a. This module stores the settlor information, beneficiary information, asset information, and witness information by staff(walk-in).

    b. The user can manage the data.

- Hibah

    a. This module can store and manage all documents involved in all process.

    b. Staff can update and know the current status of the application.

    c. Generate documents.

- Report

    a. Exportable report.

    b. Generate a report for the month, particular time range and other factors.

    c. Generate a report for each client.

1.4.1.2 Client

- Profile management

    a. The client can register their account from the website, and the system will send an email for the verification before they can log in to the system.

    b. The client able to change their profile information.

    c. Forgot password feature by using email. The system will send a link to the user's email to change the password.

- Client management

    a. This module stores the settlor information, beneficiary information, asset information, and witness information by the client(online).

    b. The client can manage the data before they sign the hibah agreement.

- Hibah

    a. The client can access the related documents that involve during the hibah process.

    b. The client gets an update the current status of the application or can check it on the website.
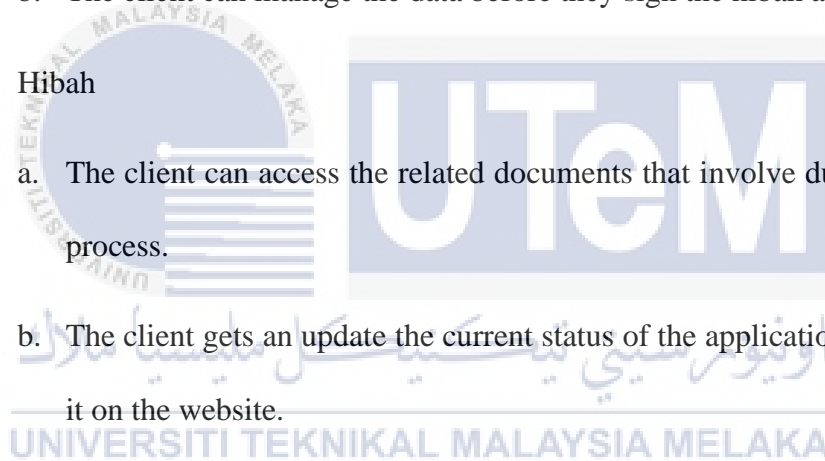
1.4.2   Target user

This project will focus primarily on two users which is,

   i.   Staff of Department Hibah and Wasiat

  ii.   Client

1.5   Project Significance

Following is the significance of this project,

    i.   The rationale of this project is to help Department of Hibah and Wasiat Majlis Agama Islam Negeri Sembilan to have their own management system and online platform.

    ii.   It also helps Department of Hibah and Wasiat Majlis Agama Islam Negeri Sembilan to organize their operation efficiently and effectively.

    iii.   This application makes it easier to client to find and make application of hibah.

1.6   Expected Outcome

The Hibah Management System will convert the manually hibah system to computerized and make the system more organized. By using this system, staff can save time and energy to process the document in large numbers. Besides, the record can be kept in the database orderly. Staff no longer needed to make documents, report, and update status manually, but five can do or get all of that from the system. Finally, this system provides analysis to the staff on all the application data required by them. This system can reduce workload and increase the system's efficiency to make all the processes run smoothly.

1.7   Conclusion

In designing this application, I cooperate with the guide of the supervisor Madam Zarita Binti Mohd Kosnin. The discussion between the supervisor and me will continue every week or every two weeks to ensure the system's progress runs smoothly without any problems. I sincerely hope that this project will be implemented effectively and provide benefits to the public.

CHAPTER 2: LITERATURE REVIEW AND PROJECT METHODOLOGY

2.1   Introduction

This chapter describes the method used in developing Hibah Management System. The methodology is commonly used to solve problems with specific components such as phases, tasks, methods, technique, and tools. With the uses of methodology, the development of an application will be more systematic and organize.

In this project, I chose agile as my methodology because agile is best suited for this iterative and incremental project. It is a type of process where demands any solutions evolve through the collaborative effort of self-organizing. I will apply the concept during the project progress.

2.2   Facts and findings

2.2.1   Domain

2.2.2   Existing System

Hibah Management System specifically developed for managing hibah application. The existing system uses a manual approach from the existence of the hibah department, which is not systematic and inefficient for both users (staff and client) to use, especially in matters involving a lot of document and transaction operations.

The existing system requires the client to get the application document and then apply it to the review process. If the application has a mistake, they need to resubmit the application until the application accepted. This system also made clients face difficulties to up to date with the status of the application. They need to contact the staff to up to date with their application status.

Furthermore, the staff also face difficulties to manage the application because this process required many documents and need to manage them properly to avoid a mistake that can cause a client to create a new application. The staff store all the document in the filing cabinet. They need to find one by one if they want to retrieve a specific document. They also need to generate the documents manually, for example, the agreement document that has all information from the hibah application and need to update new report monthly or yearly to track the process.

2.2.3 Technique

Table 2.1: Case study

Case Study	Case Study 1: Current System	Case Study 2: Hibah Management System
Provide details information of hibah	✔	✔
Manage application data	✘	✔
Able to make online payment	✘	✔
Generate documents	✘	✔

Automate report	✗	✓
Provide current status of each application	✗	✓

2.3 Project Methodology

Figure 2.1 illustrates the Agile Methodology Approach software development, model. One software life cycle is not enough to produce a complete product, but each iteration demonstrates a portion of the software features that can be checked or modified. After each cycle, it can be summarizing and obtain new project specifications, so any changes can easily be made to the software development plan.
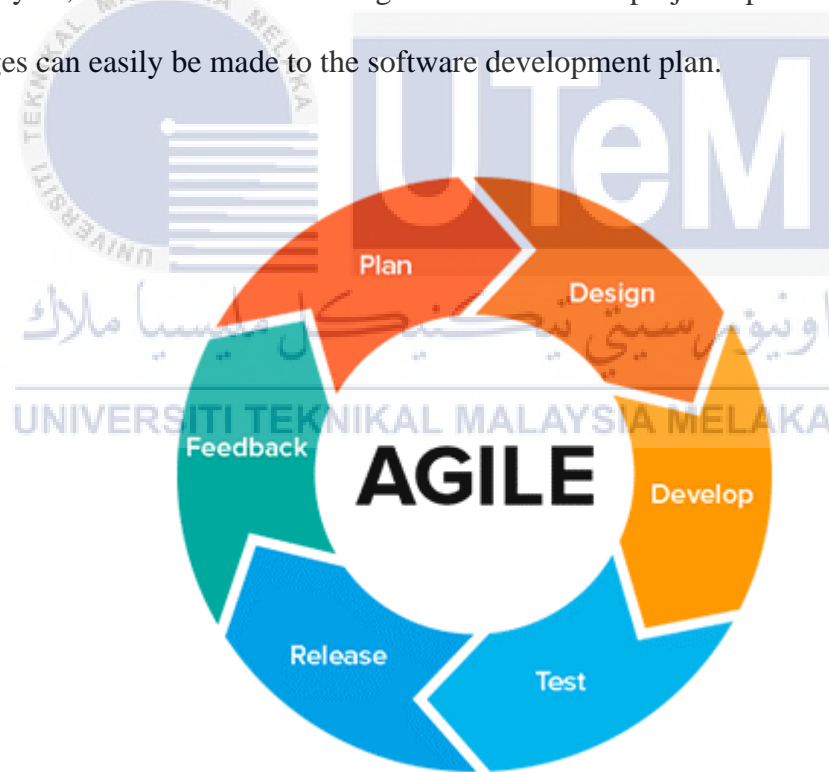
Figure 2.1: Agile methodology model

The Agile SDLC model is a mixture of iterative and incremental process models that concentrate on process adaptability through rapid product implementation

with working applications. Customer demands, especially in the field of software development, are evolving at a rapid pace. With an iterative approach to application design and architecture, agile development promotes a continuous transformation. This model consists of six stages: planning, design, build, testing, review, and launch.

In the planning phase, It can assess the time and work resources necessary to complete the project. So, in order to complete the project, I made some preparation concerning the project title, length and work resources for each job. The next step will be the concept phase. In this step, I need to work with stakeholders to identify all the requirements. In this step, we use high-level UML diagrams to show how the new feature should function.

The design or construction process then starts after the specifications are specified for the initial sprint based on the needs of the stakeholders. In this project, the system designed by splitting the system feature into several modules—next, the test process. In the Agile Lifecycle, each system will undergo multiple iterations and perform tests after each cycle. Therefore, I evaluate the functionality of the device after the complete combination of the module.

Review is the next step. In this step, I will test the features, detect bugs, record winning and losing. I will also finalize user documentation and the system. Finally, the process of launch is the finishing stage of the course of Agile life. Customers are advised and notified of new launches and alternatives.

2.4   Project Requirements

2.4.1   Software Requirement

- Visual Studio Code

- Laragon

- Github Desktop

- Microsoft Office Word

- UML

2.4.2   Hardware Requirement

- HP Pavilion 15

- Realme 3 pro

2.5   Project Schedule and Milestones
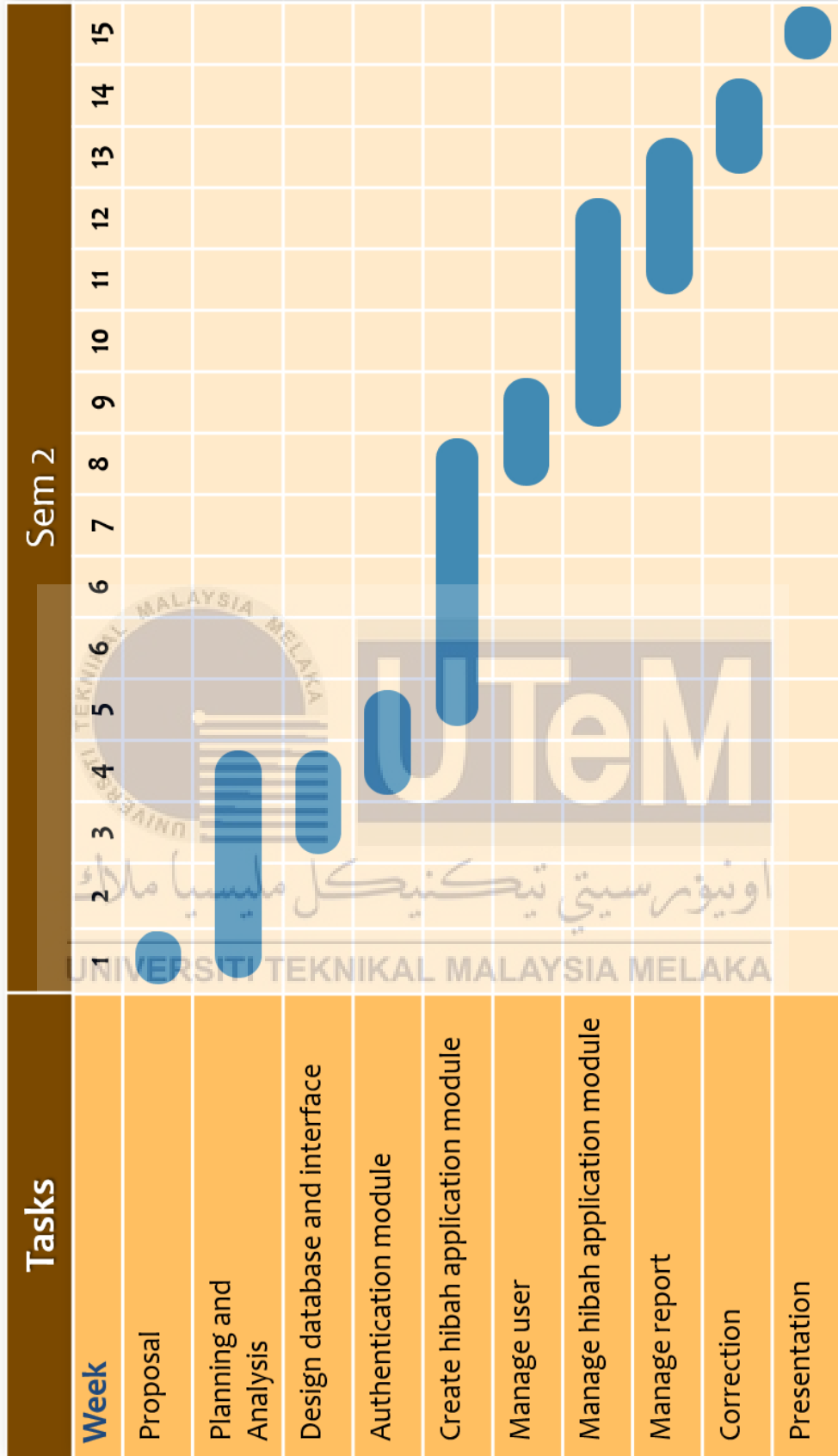
Figure 2.2: Gantt chart

2.6   Conclusion

Throughout this chapter, it can conclude that the process of the project development is executed systematically, which follows the milestones with the help of methodology on keeping things on track. The agile Model methodology is an iterative and incremental process model prioritizing process adaptability and satisfaction by rapid delivery of the working product. Suitable for a large project.

CHAPTER 3: ANALYSIS

3.1   Introduction

This section summarizes the requirement gathered for the current and proposed system

Hibah Management System in terms of business process, functional requirement, non-

functional requirement, and business flow diagram.

3.2   Problem Analysis

3.2.1   The current business flow for staff

Figure 3.1 and figure 3.2 shows the business flow of the current system (as-is system)

for staff.
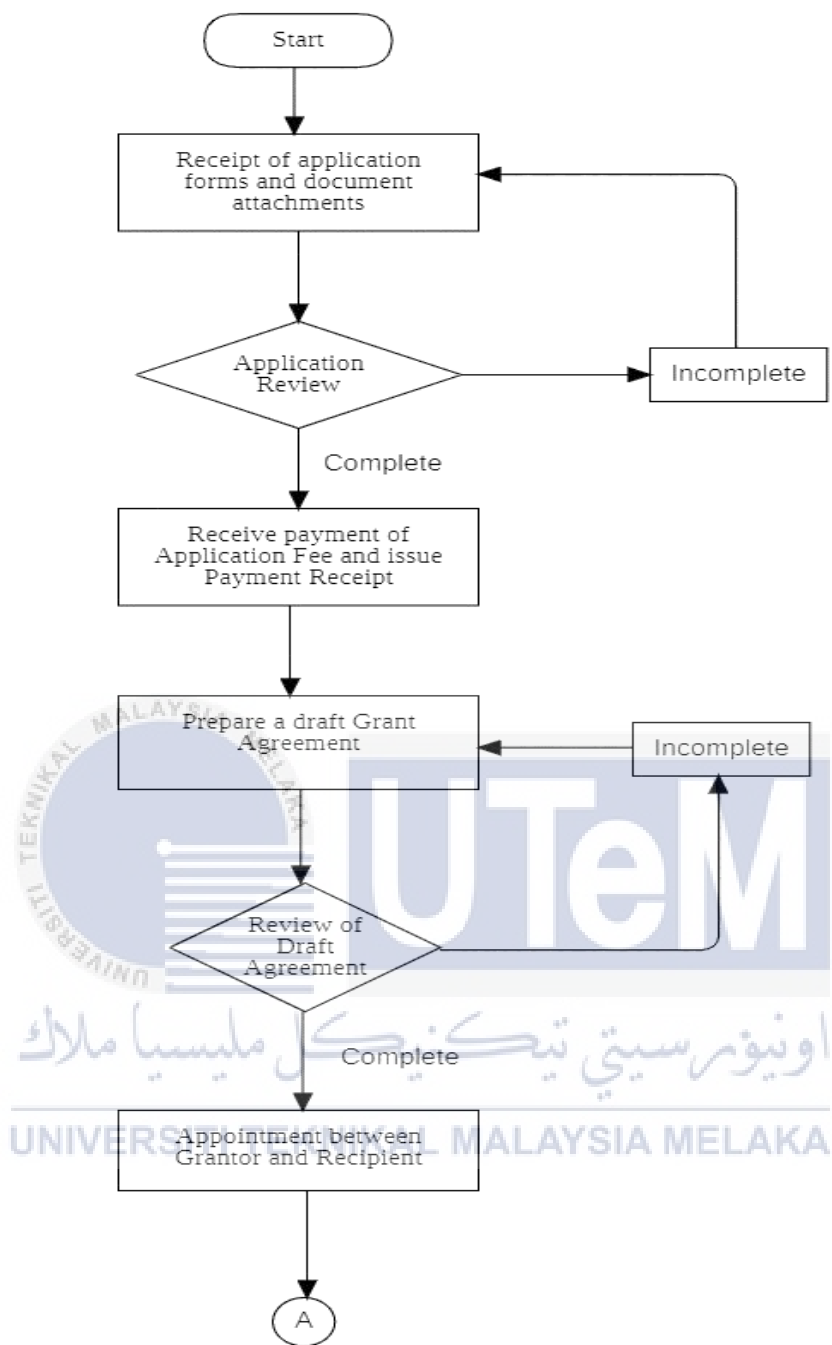
Figure 3.1: Flowchart of current system for staff part 1
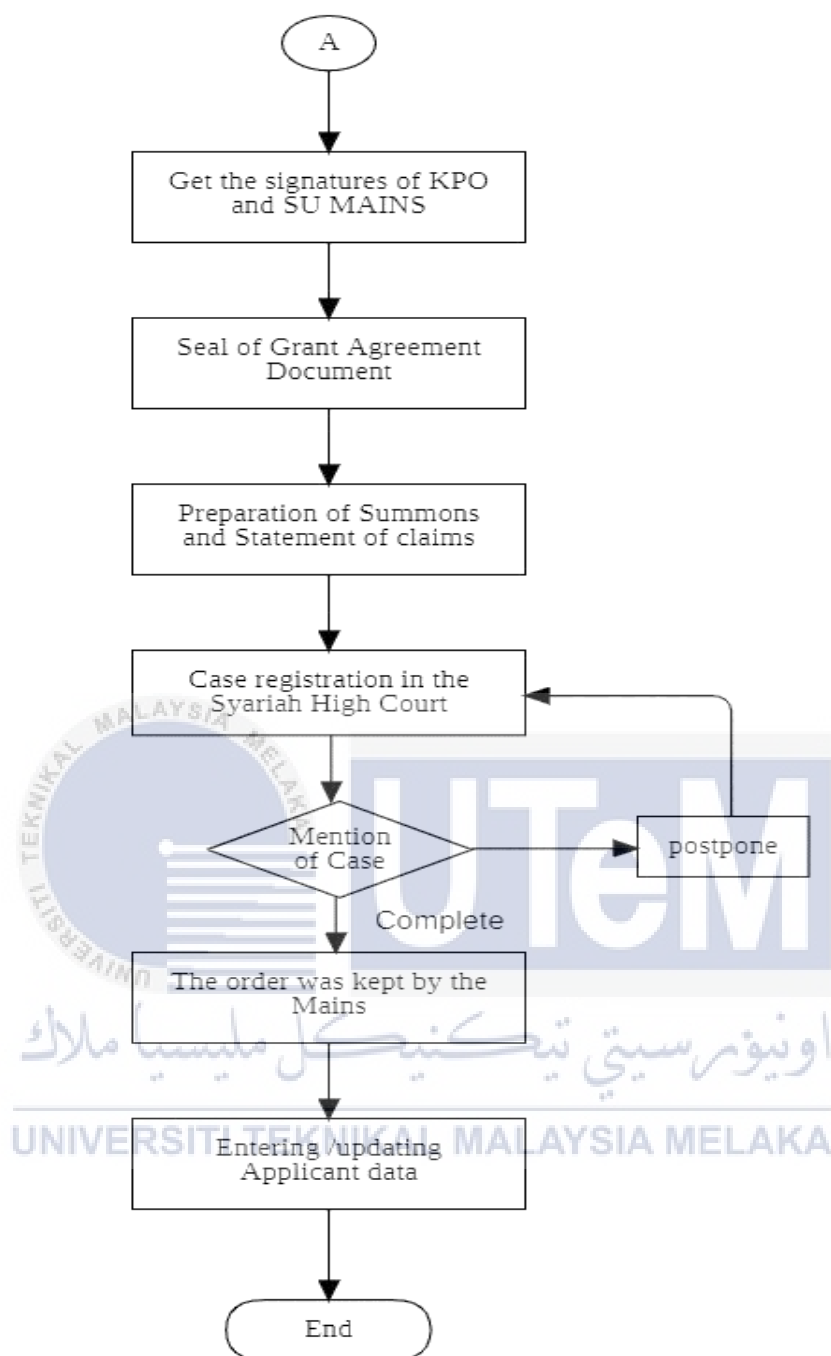
Figure 3.2: Flowchart of current system for staff part 2
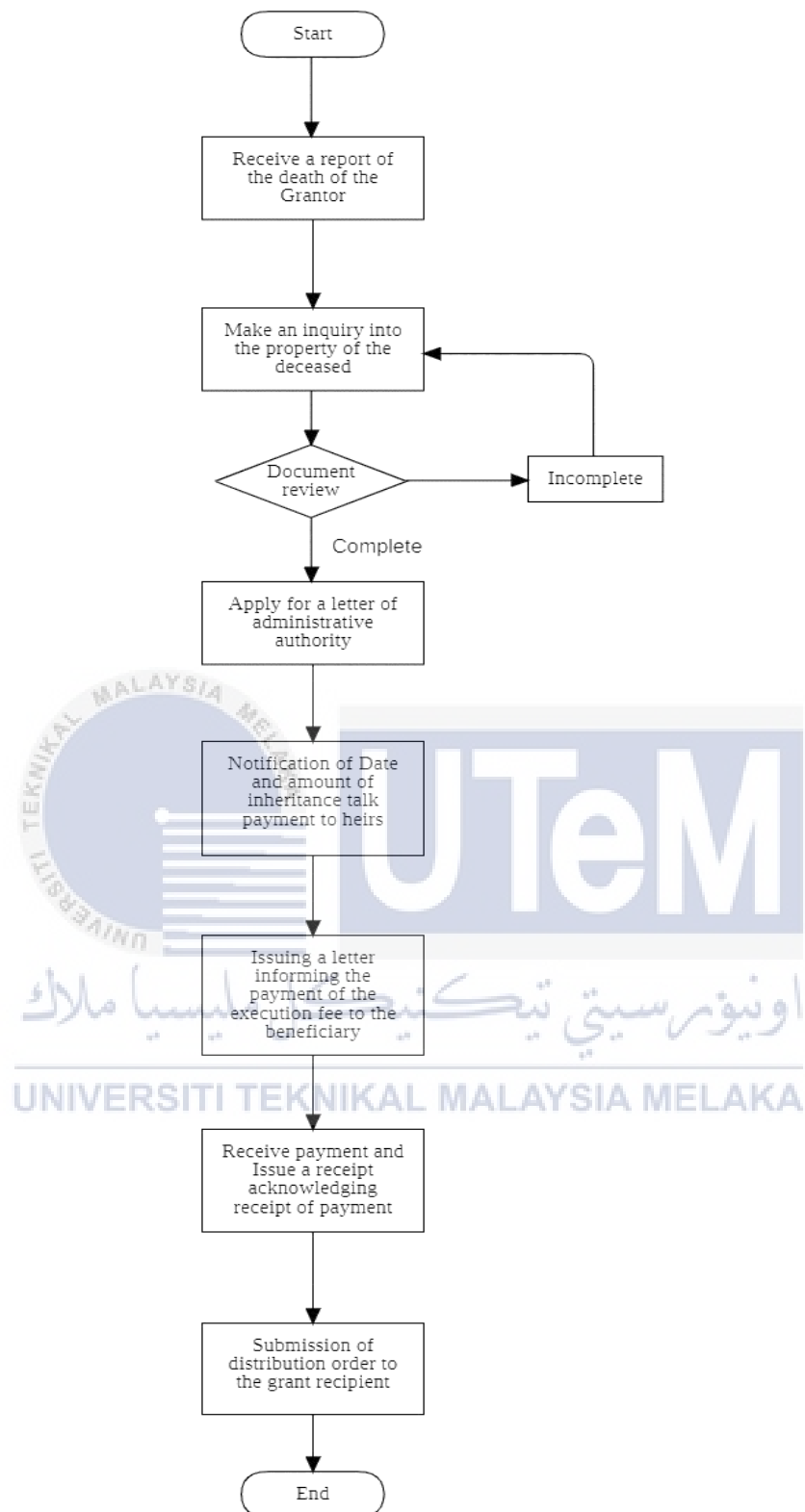
Figure 3.3: Flowchart of current system for staff

3.2.2 The current business flow for client

Figure 3.3 shows the business flow of the current system (as-is system) for client.
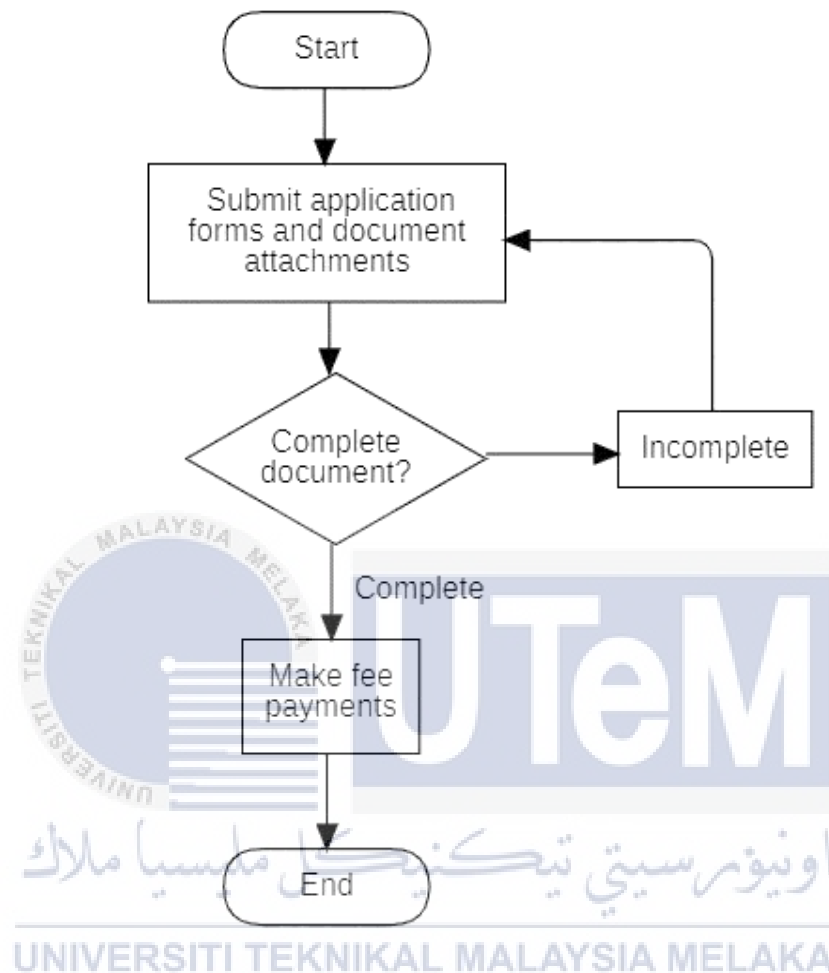
Figure 3.4: Flowchart of current system for client

3.3 Requirement analysis

This section describes the summary of function and the system operation for the Hibah Management System application. This diagram can help the to give the non-technology person that does not know about the information technology. The summary of the Hibah Management System function is as the figure below:
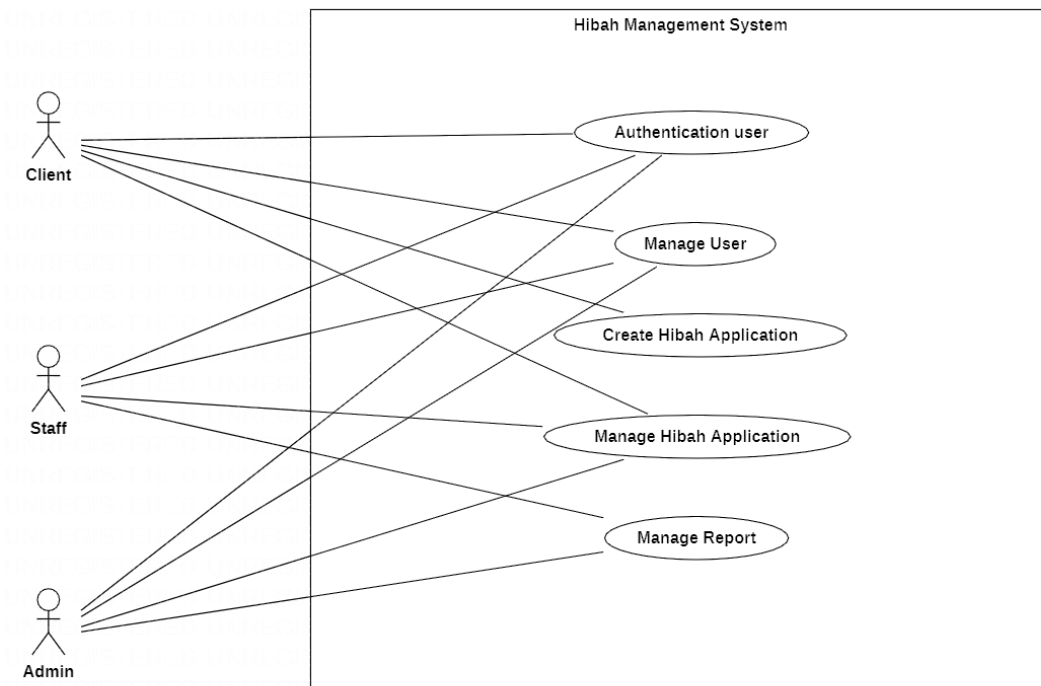
Figure 3.5: Global use case diagram

3.3.1 Data Requirement

This section will cover the data dictionary for database.

3.3.1.1 User Table

Table 3.1: User

Field Name	Data Type	Field Length	Constraint	Description
user_id	int	11	Primary key	ID, auto generated
user_name	varchar	80	Not null	Name of user
user_email	varchar	80	Not null	Email of user
user_phone_number	varchar	20	Not null	Phone number of user
keyactivation	varchar	80	Not null	Key that been used in forgot password

user_password	varchar	255	Not null	Password of user
user_pic_path	varchar	30	Default	default.png
confirmation	int	11	Not null	Status account

3.3.1.2 Giver Table

Table 3.2: Giver

Field Name	Data Type	Field Length	Constraint	Description
Giver_id	int	11	Primary key	ID, auto generated
giver_name	varchar	80	Not null	Name of giver
giver_ic	varchar	12	Not null	IC number of giver
giver_DoB	date		Not Null	Date of Birth
giver_passport	varchar	9	Not null	Passport number of giver
giver_citizenship	varchar	15	Not null	Citizenship of giver
giver_gender	char	5	Not null	Gender of giver
giver_status	varchar	15	Not null	Status of giver
giver_race	varchar	15	Not null	Race of giver
giver_religion	varchar	15	Not null	Religion of giver
giver_permanent_adress	varchar	80	Not null	Permanent Address of giver
giver_mailing_address	varchar	80	Not null	Mailing Address of giver

giver_phone_number_R	varchar	11	Not null	Regular phone number
giver_phone_number_H	varchar	11	Not null	House phone number
giver_job	varchar	20	Not Null	Job of the giver
giver_employer_name	varchar	80	Not null	Name of giver's employer
giver_employer_address	varchar	80	Not null	Address of giver's employer
giver_employer_phone	varchar	15	Not Null	Phone number of employer
giver_IC_patch	varchar	30	Not null	Name of IC document file
hibah_id	int	10	Foreign key	has

3.3.1.3 Hibah Table

Table 3.3: Hibah

Field Name	Data Type	Field Length	Constraint	Description
hibah_id	int	11	Primary key	ID, auto generated
hibah _date	date		Not null	
hibah_reference	varchar	80	Not null	
hibah_submit	date		Not null	
hibah_status	varchar	10	Not null	Current status of the hibah
hibah_code	varchar	11	Not null	
user_id	int	11	Foreign key	Make by

3.3.1.4 Recipient Table

Table 3.4: Recipient

Field Name	Data Type	Field Length	Constraint	Description
recipient_id	int	11	Primary key	ID, auto generated
recipient_name	varchar	80	Not null	Name of receipt
recipient_ic	varchar	15	Not null	IC number of receipt
recipient_phone_number	varchar	15	Not null	Phone number of receipt
recipient_address	varchar	80	Not null	Address of receipt
recipient_relationship	varchar	15	Not null	Relationship receipt with giver
recipient_ic_path	varchar	80	Not null	IC path of receipt
hibah_id	int	11	Foreign Key	has

3.3.1.5 Property Table

Table 3.5: Property

Field Name	Data Type	Field Length	Constraint	Description
property_id	int	11	Primary key	ID, auto generated
property_address	varchar	80		
property_type	varchar	15	Not null	Type of property
property_number	varchar	15	Not null	Property's number

property_PT_number	varchar	15	Not null	PT number of property
property_mukim	varchar	15	Not null	Mukim of property
property_district	varchar	15	Not null	District of property
property_width	varchar	15	Not null	Width of property
property_status	varchar	15	Not null	Status of property
property_distribution_intructions	varchar	80	Not null	Distribution instruction of property
property_path	varchar	80	Not null	path of property file name
hibah_id	int	11	Foreign key	has

3.3.1.6 Witness Table

Table 3.6: Witness

Field Name	Data Type	Field Length	Constraint	Description
witness_id	int	11	Primary key	ID, auto generated
witness_name	varchar	80	Not null	Name of witness
witness_ic	varchar	15	Not null	IC no. of witness
witness_address	varchar	80	Not null	Address of witness

witness_phone_number	varchar	30	Not null	Phone number of witness
witness_ic_path	varchar	80	Not null	Path of witness IC
hibah_id	int	11	Foreign key	has

3.3.1.7 Document Table

Table 3.7: Document

Field Name	Data Type	Field Length	Constraint	Description
document_id	int	11	Primary key	ID, auto generated
document_type	varchar	30	Not null	
document_path	varchar	80	Not null	Path of the document
document_date	date		Not null	Date of the document upload
staff_id	int	11	Foreign key	has

3.3.1.8 Update Hibah

Table 3.8: Update hibah

Field Name	Data Type	Field Length	Constraint	Description
update_id	int	11	Primary key	ID, auto generated
update_name	varchar	30	Not null	Name of the update
upadte_description	varchar	30	Not null	Description of update
update_date	date		Not null	Update date
hibah_id	int	11	Foreign key	Has

staff_id	int	11	Foreign key	Make by

3.3.1.9 Staff Table

Table 3.9: Staff

Field Name	Data Type	Field Length	Constraint	Description
staff_id	int	11	Primary key	ID, auto generated
staff_name	varchar	80	Not null	Name of staff
staff_ic_number	varchar	15	Not null	IC no. of staff
staff_id_number	varchar	15	Not null	ID no. of staff
staff_phone_number	varchar	15	Not null	Phone number of staff
staff_email	varchar	80	Not null	Email of staff
staff_type	varchar	5	Not null	Type of staff
staff_password	varchar	80	Not null	Password of staff
staff_gender	varchar	11	Not null	Gender of staff
staff_confirmation	int	11	Not null	First time login status
staff_pic_path	varchar	80	Not null	File name of the picture

3.3.1.10 User Document

Table 3.10: User document

Field Name	Data Type	Field Length	Constraint	Description
document_id	int	11	Primary key	ID, auto generated

document _date	date		Not null	Date of the document upload
document _path	varchar	80	Not null	Path of the file name
document_name	varchar	80	Not null	Document name
hibah_id	int	11	Foreign Key	Make by

3.3.1.11 Position Table

Table 3.11: Position

Field Name	Data Type	Field Length	Constraint	Description
position_id	int	11	Primary key	ID, auto generated
position_name	varchar	80	Not null	Position name
position_person	varchar	80	Not null	Name of the person

3.3.1.12 Payment Table

Table 3.12: Payment

Field Name	Data Type	Field Length	Constraint	Description
payment_id	int	11	Primary key	ID, auto generated
payment_reference	varchar	80	Not null	Reference of payment
payment_status	varchar	30	Not null	Status of the payment
payment_billcode	varchar	30	Not null	Billcode payment

payment_amount	varchar	30	Not null	Amount of the payment
payment_date	date		Not null	Date payment
hibah_id	int	11	Foreign Key	Make by

3.3.2   Functional Requirement

This section will define the high-level requirements and features of this system. It focuses on the capabilities of the system as required by the stakeholders and the target users. The table shows the breakdowns of each functional requirements for the Hibah Management System.

Table 3.13: The breakdowns of functional requirements

FR No.	Requirement	Description
Authentication user		
FR_01	Login	The system shall allow user to access to their account.
FR_02		The system shall validate a valid email format
FR_03		The system shall check the correct password.
FR_04		The system shall not allow user to leave empty field before logging in to the app.
FR_05		The system shall not allow user to login if they do not have any account.
FR_06		The system shall force staff to change the password when first time log into the system.
FR_07	Change Password	The system shall allow user to change the password
FR_08		The system shall validate the length password must more than 7 characters.
FR_09		The system shall validate the confirmation password entered must match with password entered.

Manage User		
FR_10	Register	The system allow admin to create account for staff.
FR_11		The system shall not allow staff to register their account.
FR_12		The system shall validate email format is correct.
FR_13		The system shall validate the email whether has been used or not.
FR_14		The system shall validate the password must more than 6 characters.
FR_15		The system shall validate all the data in correct format.
FR_16		The system shall validate the confirmation password must same with entered password.
FR_17		The system shall send an email to staff after admin successfully registered their account.
FR_18		The system shall allow client to register new account.
FR_19		The system shall not allow user to leave empty field when create the account
FR_20	View User Account	The system should allow user to view the account information such as name, email, number phone and etc.
FR_21	Edit User Account	The system should allow user to update their account information.
FR_22		The system shall not allow user to leave empty field when updating their account.
FR_23		The system shall validate the email whether exist or not.
FR_24		The system shall all data format is correct such phone number and email.
Create Hibah Application		

FR_25	Create Hibah	The system shall allow client to create hibah application.
FR_26		The system shall allow client to upload pdf files.
FR_27		The system shall save the data in database for each section.
FR_28		The system shall not allow client to make more than one hibah application.
FR_29		The system shall not allow user to leave empty field.
FR_30		The system shall validate all data in correct format.
		The system shall notify the staff about the new application after client make the payment
Manage Hibah Application		
FR_31	View Hibah	The system shall allow staff to view all the application.
FR_32		The system shall allow client to view their application.
FR_33		The system allow client to view their current status of the hibah application.
		The system shall allow user to download the documents.
FR_34	Update Giver, Receiver, Property, and Witness Information	The system shall allow client to update the hibah application information.
FR_35		The system shall validate all the data format is correct.
FR_36		The system shall save date and time when the event occurs.
FR_37		The system shall not allow staff to update the information.
FR_38		The system shall not allow user to leave required filed empty.
FR_39	Update Hibah Application Status	The system shall not allow client to update the status.
FR_40		The system shall allow staff to update the status

FR_41		The system shall allow staff upload new document.
FR_42		The system will send email to client after update the status
FR_43	Generate Document	The system shall allow staff to generate document such as agreement document in doc file.
FR_44	Delete Hibah Application	The system shall not allow user to delete the hibah application
FR_45		The system shall not allow staff to delete the hibah application
Manage Report		
FR_46	Generate Report	The system shall allow staff to generate monthly and yearly report.
FR_47		The system shall allow staff to print the generated report.

3.3.3   Non-functional Requirement

This section describes the general non-functional requirements for the system. They are as follows:

Table 3.14: The breakdowns of non-functional requirements

NFR No.	Requirement	Description
Authentication user		
NFR _01	Availability	Available to use. User can be able to use the system. It going to be operational 24 hours per day and 7 days a week.
NFR _02	Usability	Easy to use. User should can be able to use the application without problem or manuals
NFR _03	Response Time	Database update. The system should not take than 3 second to load the screen.

3.4   Conclusion

On this chapter, the requirement of the system has been listed down from functional requirement to the non-functional requirements. These requirements are essential to assist the developer and act as guidance to develop the system according to the proper compliance endorsed during the initial planning. Other necessary data also collected in tabulated data, such as a data dictionary, ease the developer in designing the database. They can identify data that input and output from the system.

CHAPTER 4: DESIGN

4.1 Introduction

The architecture of the system must be concerned with the design of the system. This framework is used by users using an immersive graphical user interface (GUI). The modules are aimed to provide an integrated framework that is aimed to enhance the flexibility of the forms and content of the output product using a database-centred architecture. Instead of a standard range of architectural products with a fixed format, items must be developed in a format with specifications tailored to their demands.

4.2 High-Level Design

4.2.1 System Architecture

Web Application Architecture is a framework that contains the links and interactions of application components such as middleware systems, user interfaces, and databases. This architecture's components are classified into two types: user interface components and structural components.

User interface app components are web pages displaying dashboards, logs, alerts, configuration options, and other information. They have no bearing on the application's structural development and are mainly concerned with the user interface/experience.

Business logic and data persistence are examples of structural components constructed using PHP, Python, Java, Ruby, .NET, and Node.js, among other languages. This component also includes a database that offers and maintains essential data for the programme. It may also provide the business logic and other information that the web application server manages.
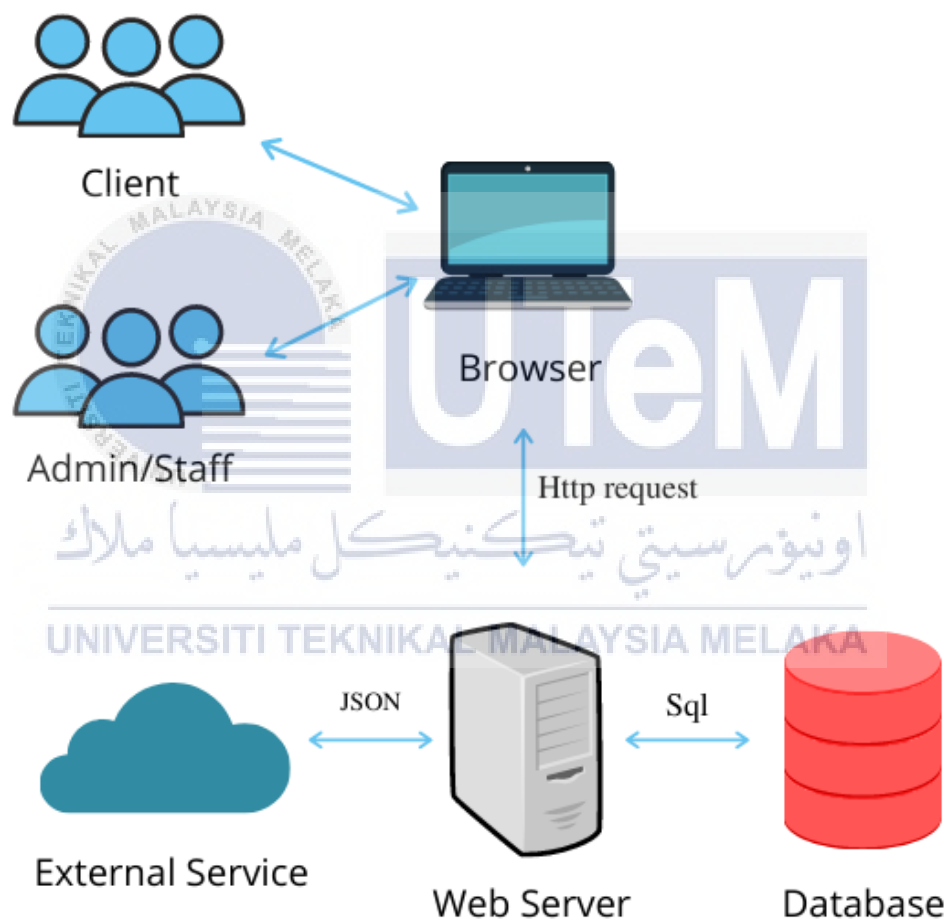
Figure 4.1: Web application architecture

4.2.2   User Interface Design

4.2.2.1 Navigation Design

a)   Vertical navigation menu
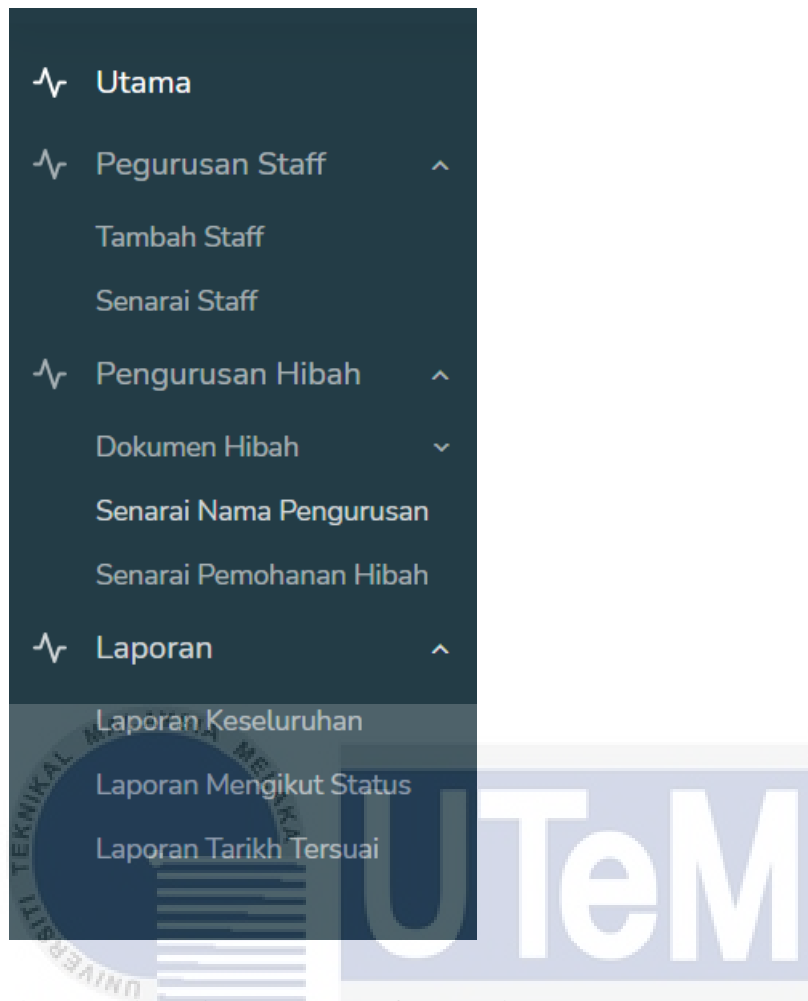
Figure 4.2: Vertical side bar menu for admin

b) Breadcrumb menu

Pengurusan Hibah > Senarai Pemohonan Hibah > Proses Pemohonan

Figure 4.3: Breadcrumb in staff and admin

c) Horizontal navigation menu

Penghibah > Penerima > Harta > Saksi > Pembayaran > Status

Figure 4.4: Horizontal menu for client

d) Hamburger navigation menu
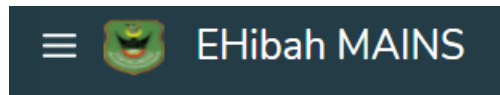
Figure 4.5: Hamburger menu to close and open vertical menu

4.2.2.2 Input Design
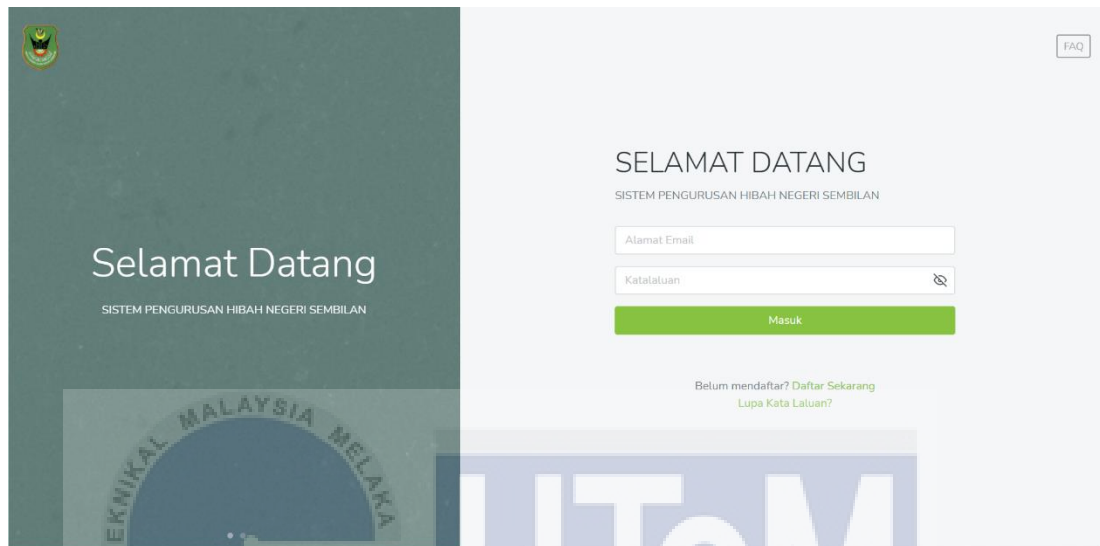
Figure 4.6: Login page

Figure 4.6 shows the login page used for all users, admin, staff, and client. It has two input field, email and password. Both input field cannot have a null value. It will display an error message to enter a value. The form will not be submitted if the email format is not correct and the password is disabled for a user to see.

Figure 4.7: Register page

Figure 4.7 show the registration page for a new client. The client needs to fill in three basic information: the full name, email, password, and re-password. This system will validate all the field. It cannot have an empty field; the email must be unique (not in the database), and the password and re-password must same before sending an email to the client for email verification. After that, a client can log in and use the system.

Figure 4.8: Forgot password page

Figure 4.8 is for a user to reset their password by email. The system will validate the email format and will display an error if the format is not correct. After that, the user

will get an email that contains a link that includes some data like email and key that will use while resetting the password.

Figure 4.9: Reset password page

Figure 4.9 show the page to reset the password after the user clicks the given link from the email. The system will make check either the link is active and can be used for this process. After the system checks the link, the user can enter a new password and re-password. When the user clicks the button, the system will validate the password with criteria that already set. The error message will display when the criteria do not achieve.

Figure 4.10: Client profile page

Figure 4.10 is for a client profile. User can update information such as profile picture, name, email and phone number. After that user click the button to update the new information.

Figure 4.11: Giver page part 1

Figure 4.12: Giver page part 2

Figure 4.11 and figure 4.12 shows a form to add information of hibah' s giver. Clients need to fill up all the information. The system will validate some data as IC format, file format, and all the fields must have value before saving data into the database.

Figure 4.13: Add property page

Figure 4.13 show the form of property hibah. The user needs to fill all the fields. The system will validate all of that and the file format. An error message will display if have empty field and wrong file format during the upload.

Figure 4.14: Add recipient page

Figure 4.14 show add recipient page. Validation that has in this page is it cannot contain an empty field, phone number and file format for the document must correct and IC number for each recipient for particular hibah application must unique.

Figure 4.15: Add witness page

Figure 4.15 show add witness page. The system will validate that each field must have data, or an error message will display. File format and phone number must correct, and IC number must be unique for a particular hibah application.

Figure 4.16: Staff profile page part 1

Figure 4.17: Staff profile part 2

Figure 4.16 and figure 4.17 shows page to update profile account for admin and staff. It can update the phone number and email only. Email and phone number must be in the correct format. Email needs to be unique. Cannot have the same email in the database.

Figure 4.18: Add staff page part 1

Figure 4.19: Add staff page part 2

Figure 4.18 and figure 4.19 shows the page to add new staff. The user needs to fill in all the fields. The system will validate the empty field and display an error message. Several fields must have the correct format and also unique such as email, id and IC number.

Figure 4.20: Add agreement document page

Figure 4.20 shows the page for the staff to upload the complete agreement document. This process occurs after the document been signed by the client, witness, and higher officer from the MAINS. The system only allows once document for each application.
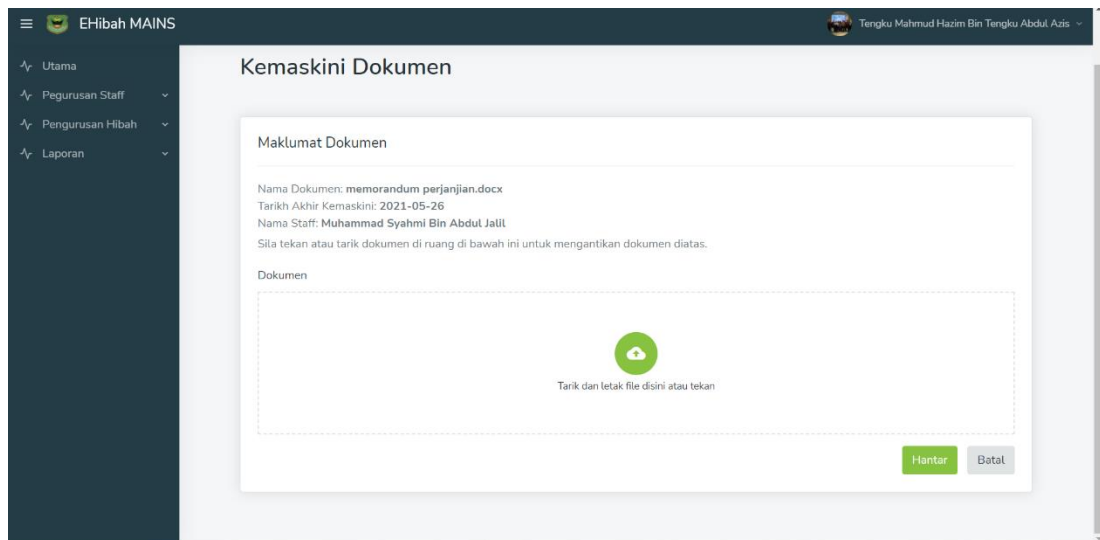
Figure 4.21: Update document page

Figure 4.21 shows the page to update the existing agreement document for particular application. The system prevents the user to add more than one document. If the user has updated document, they can do in this page. System will make the changes.

Figure 4.22: Add court instruction document page

Figure 4.22 shows the page to add court instructions document. This page only can be access after the user successful upload agreement document and court process. The system will store the document.

Figure 4.23: Add document manual page

Figure 4.23 shows the form to add a document. The system will validate the document type extension before adding it to the database. An error message will display if the file type is not correct.

Figure 4.24: Hibah status page

Figure 4.24 shows status hibah for specific hibah application. Staff or admin need to select the status of the application is accepted or rejected. If the status rejected, the system would validate the Text area field either it has a message or not. If not, the

system will display an error message. The system will email a client the application status with reasons if the application has been rejected. A client can update the data.

Figure 4.25: Update officer information

Figure 4.25 for updating the officer details used in the automatic generating document in the hibah process. The system will validate the data, it cannot be empty, or the error message will display.

4.2.2.3 Output Design

Figure 4.26: Main page client

After the client successful login into the system, they will see page as figure 4.36. In this page, the client can see list of the application with their details. The client can have maximum three application that have status 'Dalam proses'. They can 'Tambah Permohonan' to add new application. To view the details for each application, they need to click 'Lihat' link and will redirect for other page.
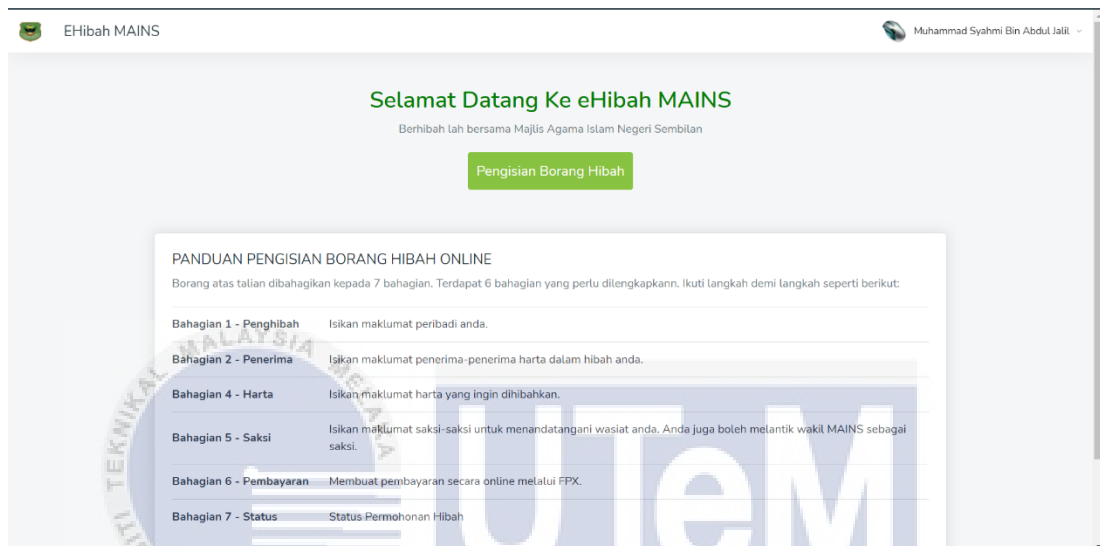
Figure 4.27: List of process involve in hibah application

Figure 4.27 shows the page after the client click 'Lihat' link from figure 4.26. This page only shows the process that need they complete. After that, they can go to details form by click 'Pengisian Barang Hibah' button.

Figure 4.28: FAQ

Figure 4.28 shows the page that contains collection of frequency asking question.
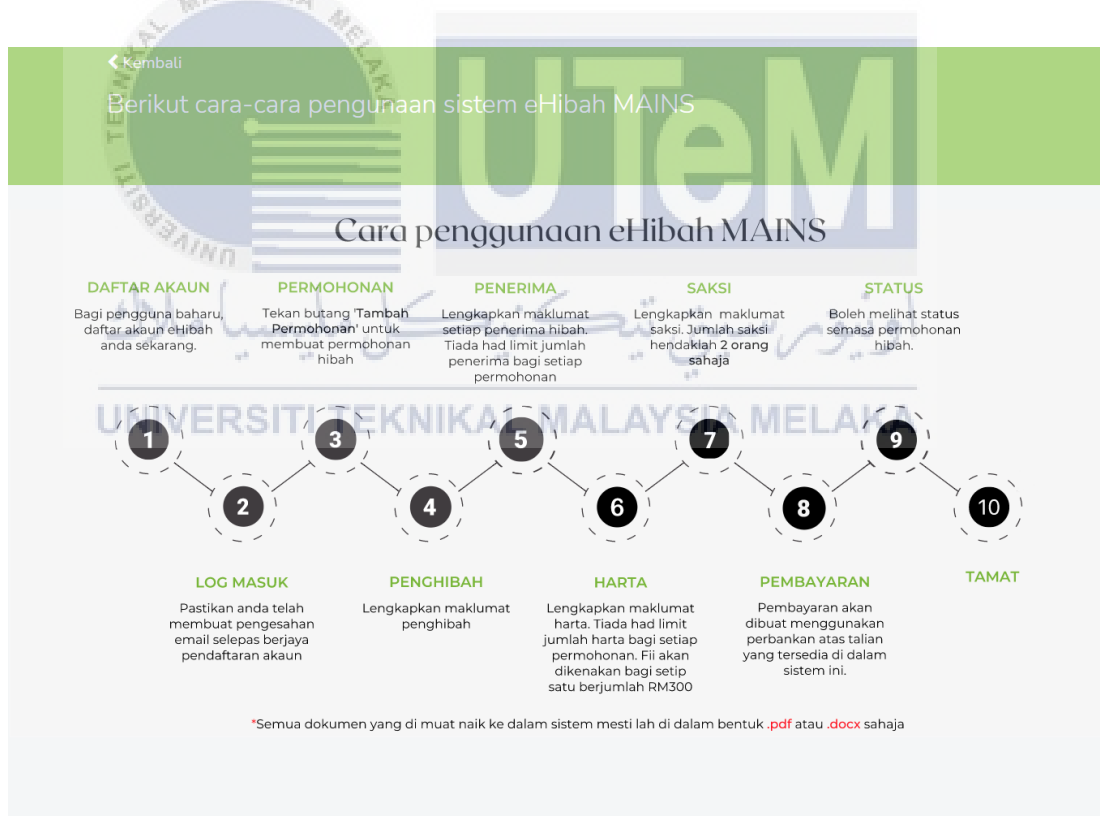
Figure 4.29: User manual

Figure 4.29 shows the manual user for the client to get the steps to make the hibah application.

Figure 4.30:Details hibah payment

Figure 4.30 shows the page for the user to make the payment via online banking.

Figure 4.31: Application status
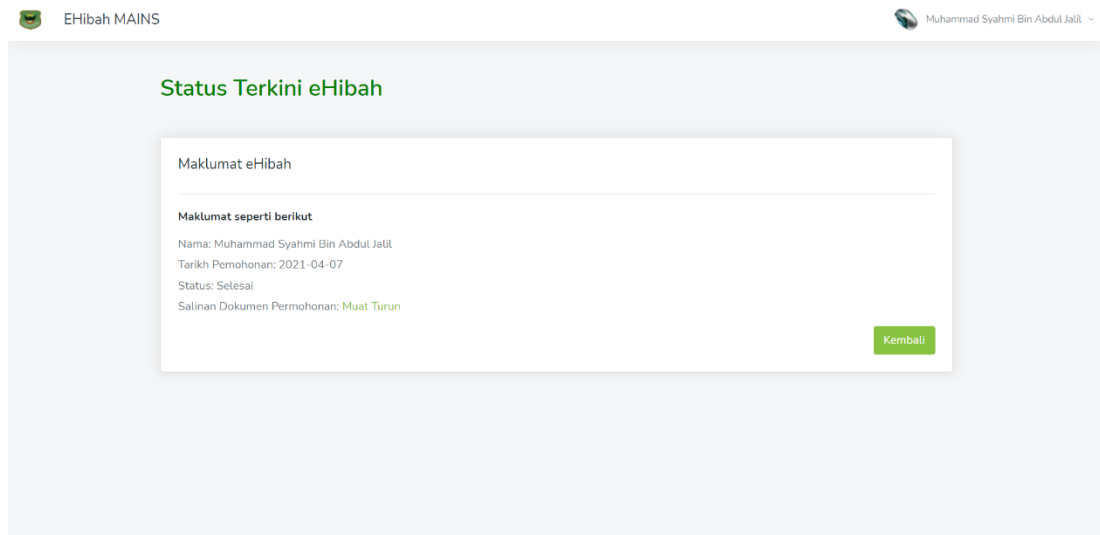
Figure 4.32: Application status part 2

After user successful finish their application. They will see the current status of their application as in figure 4.31 and figure 4.32. They can download the copy of the application by click 'Muat Turun' link.
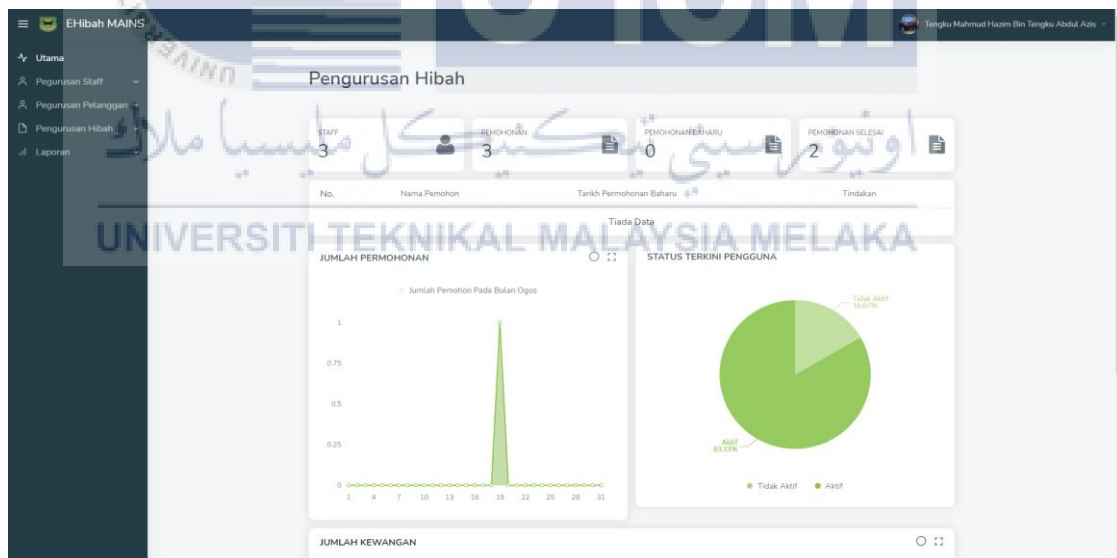
Figure 4.33: Dashboard admin

Figure 4.34: List of documents (Generate)

Figure 4.34 shows the list of template document that be use in the auto generating document. They can download the document and also update the document with strict guidelines format to ensure the template can be use succefully.

Figure 4.35: List of complete hibah application

Figure 4.35 shows the page of complete hibah application list. The user can find the current status for each application. For the detail's information, they need to click 'Seterusnya' link.
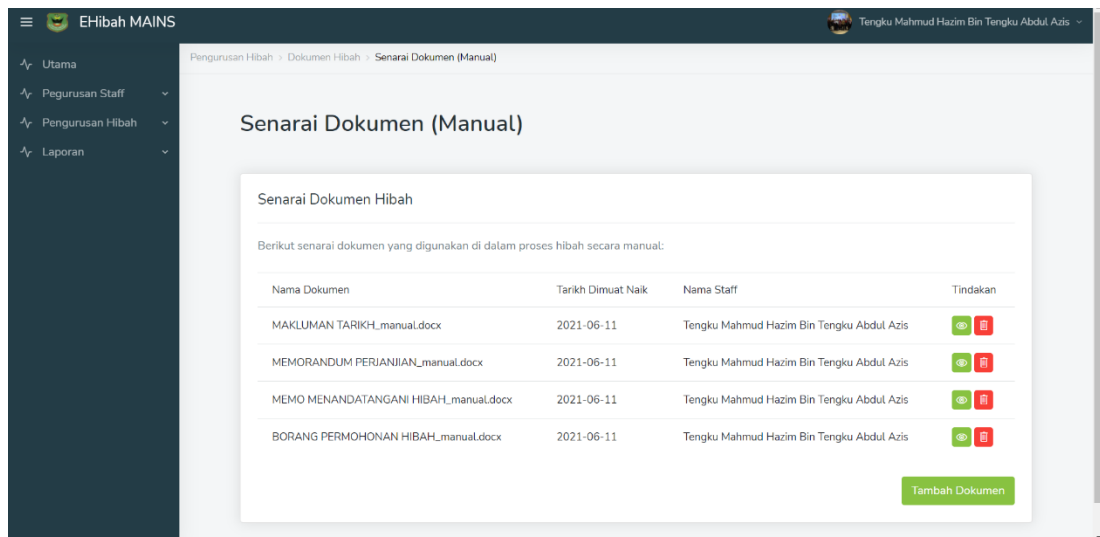
Figure 4.36: List of documents (Manual)

Figure 4.36 shows the list of documents for manual process. They can add, view and delete the document.

Figure 4.37: List of officers

Figure 4.37 shows the list of the high officers that involve in the generating document for the agreement document. They can update the information by click the 'Kemaskini' link.

Figure 4.38: List of hibah process

Figure 4.38 shows the list of the hibah application. They process only can be access if the status of the process above them 'Selesai' except for the first process. Otherwise, they can access other process.

Figure 4.39: List of staff

Figure 4.39 shows the list of the staff that registered in the system. The user can view and edit the details information of the specific user. This page can only be access by the admin.

Figure 4.40: Overall report
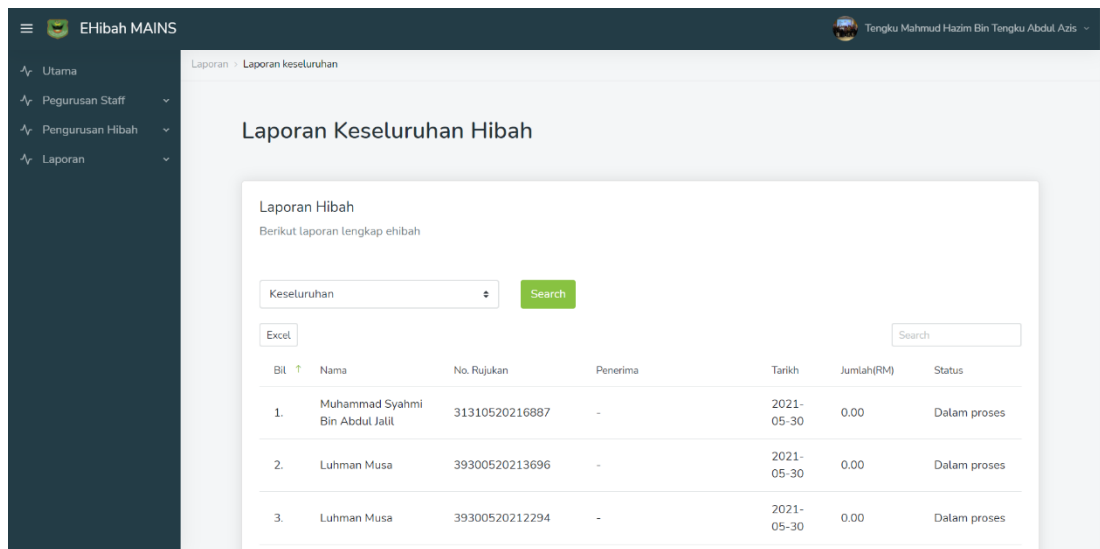
Figure 4.41: Report by date

Figure 4.42: Report by status

Figure 4.40, figure 4.41 and figure 4.42 shows the page that displays the report. It has by overall range date and status. After that user can click the Excel button to download the report in excel format.
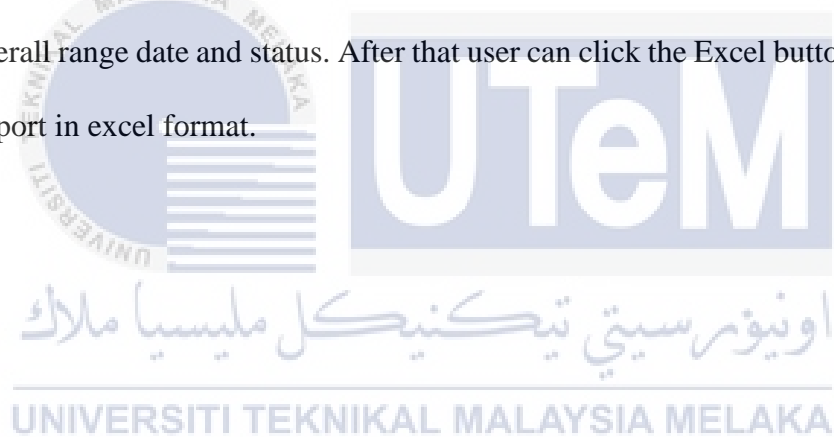
4.2.3 Database Design

4.2.3.1 Conceptual and Logical Database Design
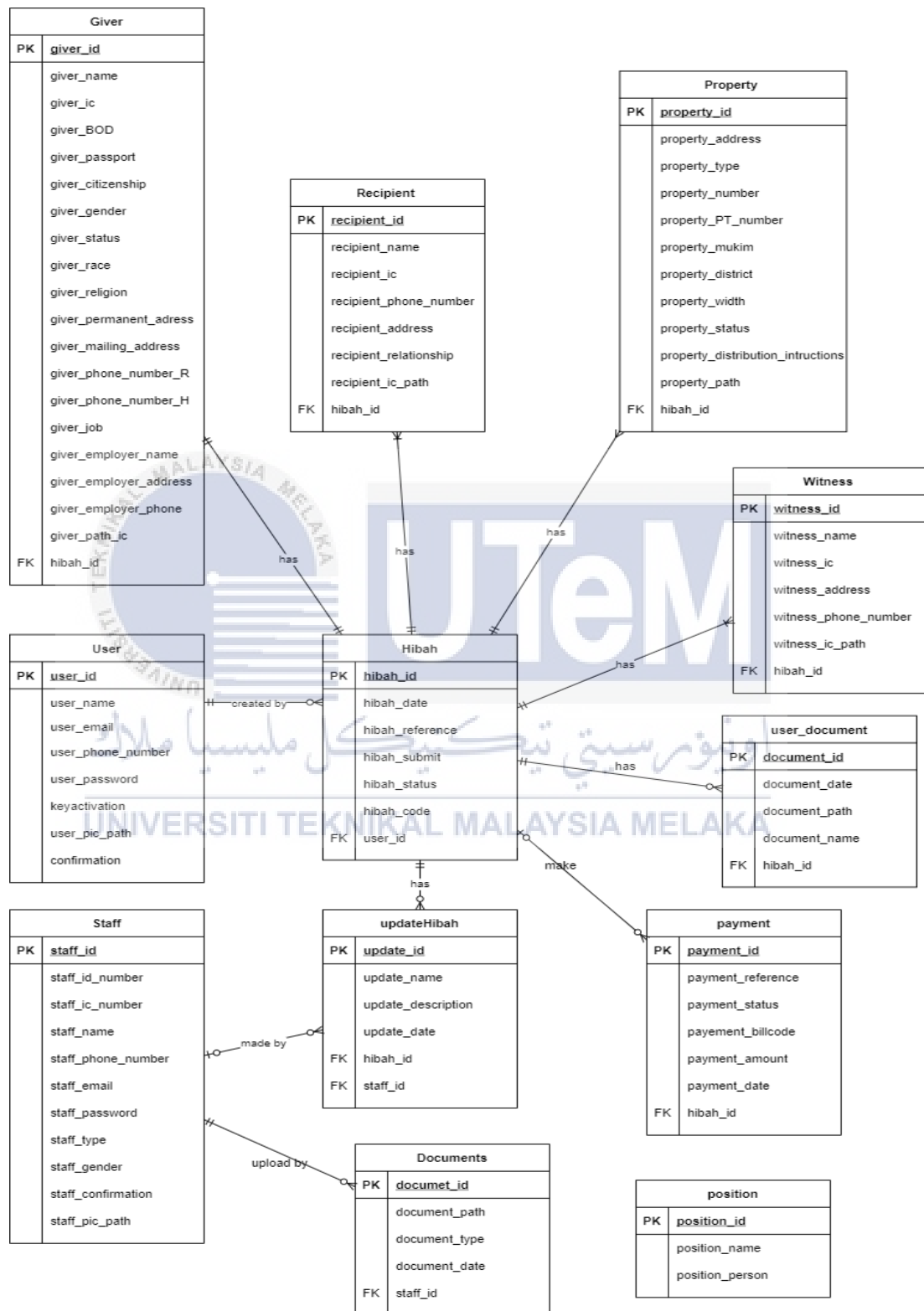
Figure 4.43: Conceptual database design
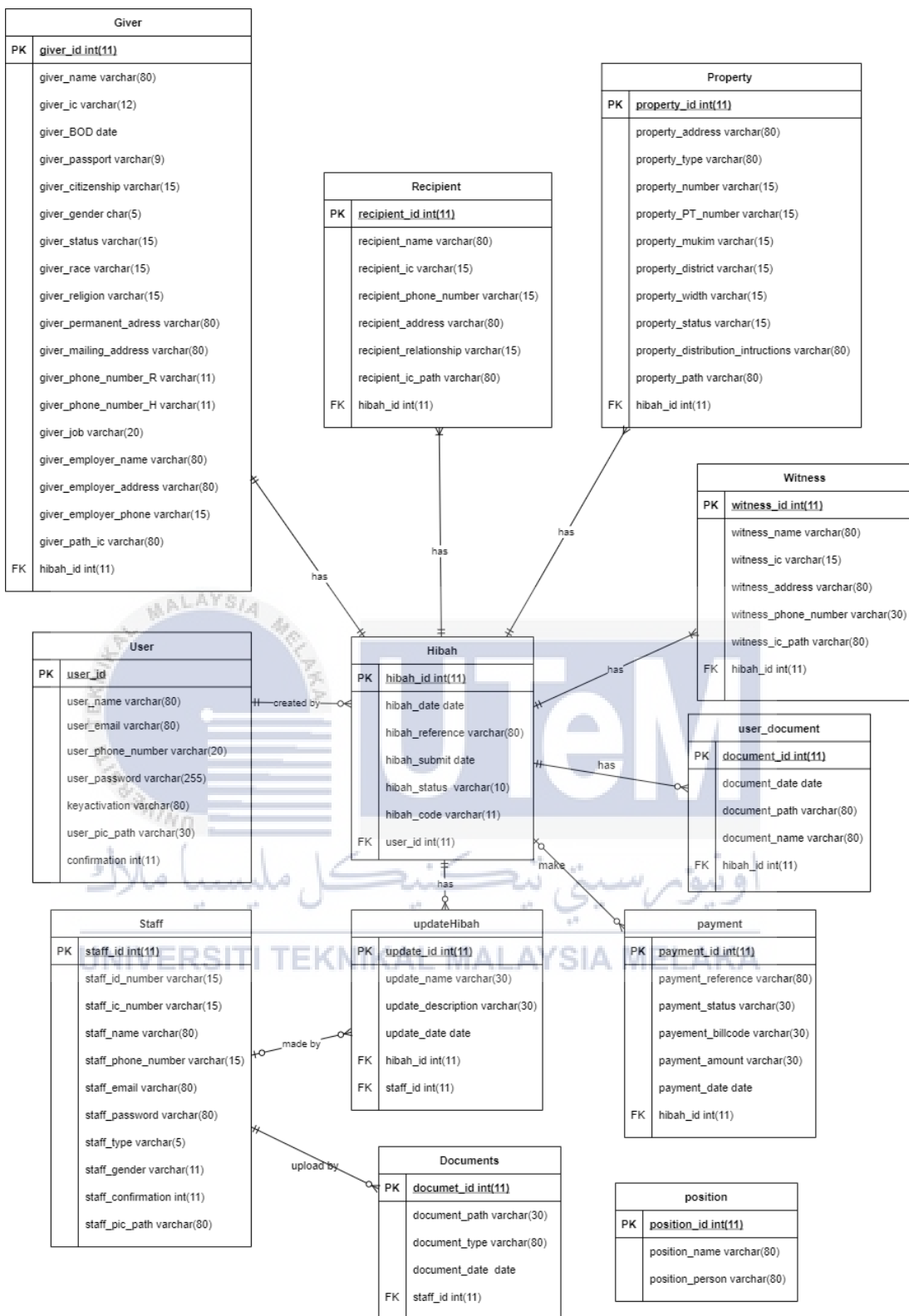
Figure 4.44: Logical database design

4.3   Detailed Design

4.3.1   Software Design
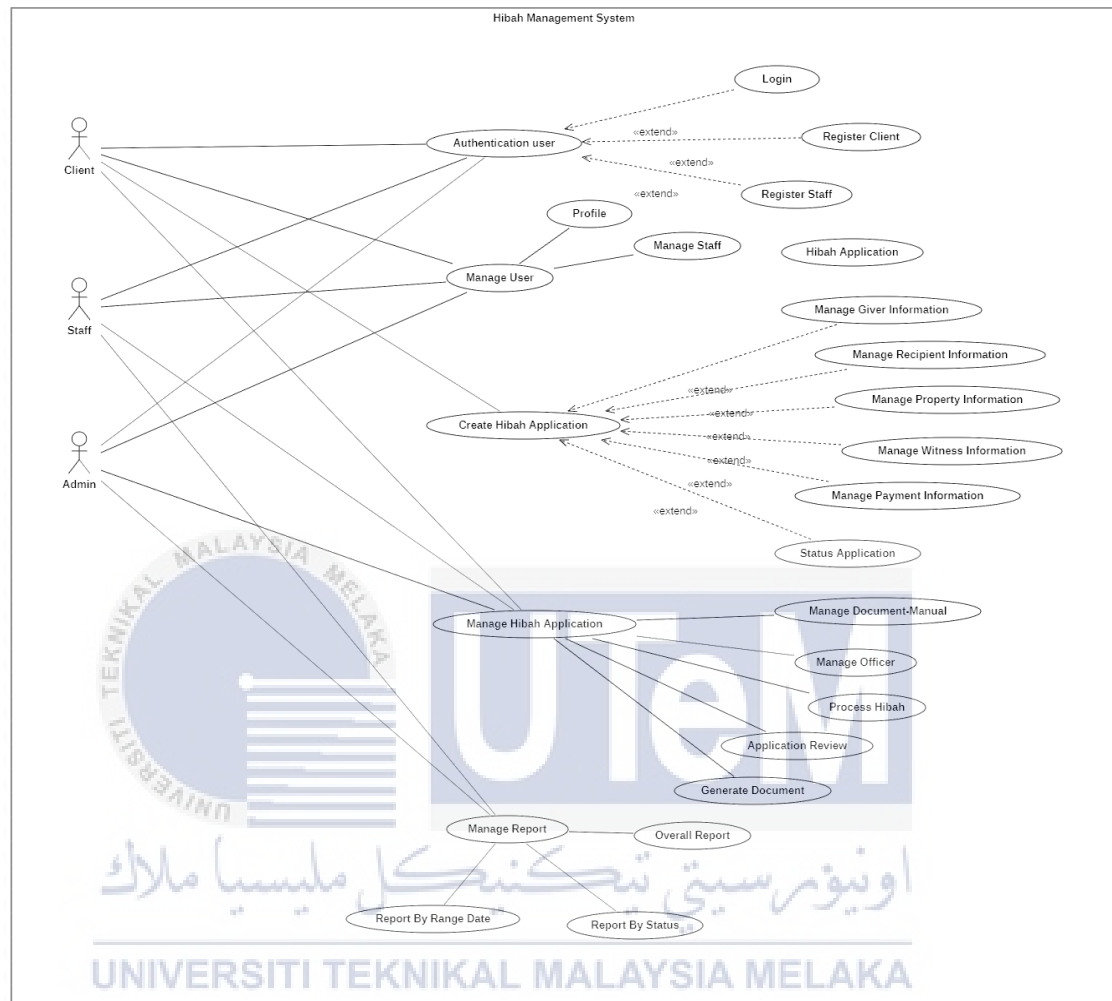
Figure 4.45: Use case diagram

4.3.1.1 Authentication user

Table 4.1: Use case login

Use Case Name	Login
Description	This use case describes process of login
Pre-Condition	Email already validates
Post-Condition	User login into the system based on type user

Flow of Events	Primary flow:
	i. The use case starts when user go to Hibah Management System
	ii. User insert username and password
	iii. Validating the data
	iv. User will redirect to dedicate page based on their type.
	v. Use case ends.
	Exception flow:
	[A]
	i. The user inserts wrong credentials for the username or password.
	ii. Error message will display.
	iii. Return to primary flow.
	[B]
	i. The user inserts correct credentials but the status of email confirmation false.
	ii. Error message will display.
	iii. Return to primary flow.

Table 4.2: Use case register client

Use Case Name	Register client

Description	This use case describes for registration of a new client.
Pre-Condition	
Post-Condition	Send email with link for email conformation.
Flow of Events	Primary flow:

Primary flow:

i. The use case starts when the client click sign up new account.

ii. The client needs to fill in all the required data.

iii. Validate email do not exist in database.

iv. Validate other information.

v. Add information into database.

vi. Use case end.

Exception flow:

i. The user inserts existing email or have empty field(s).

ii. Error message will display.

iii. Back to primary flow.

Table 4.3: use case register admin or staff

Use Case Name	Register Admin/Staff
Description	This use case describes the process of admin to add new admin or staff.
Pre-Condition	Admin login into the system.
Post-Condition	Send email contains email and temporary password.
Flow of Events	Primary flow: i. The use case starts when admin go to add staff. ii. The admin enters the staff details. iii. Validate all details. iv. New staff is recorded into database. v. Back to list of staff vi. Use case end. Exception flow: i. The admin inserts email or staff id that already exist in database. ii. Error message will display.

	iii. Back to primary flow.

4.3.1.2 Manage user
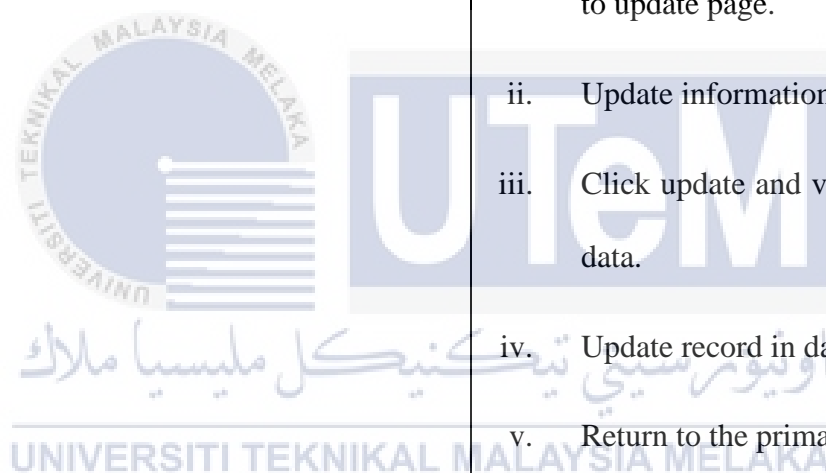
Table 4.4: Use case profile

Use Case Name	Profile
Description	This use case describes the process to view and update the basic information of the user account.
Pre-Condition	Login into the system.
Post-Condition	
Flow of Events	Primary flow: i. This use case starts when user go to profile page. ii. Users have option to update the information or back to main [A1] [A2] iii. Use case ends Alternative flow: [A1] i. Admin change the information including the profile picture.

	ii. Click update.
	iii. New information recorded.
	iv. Return to primary flow.
	[A2]
	i. Click back button.
	ii. Go to main page.

Table 4.5: Use case manage staff

Use Case Name	Manage staff
Description	This use case describes the process of admin managing the staff and admin from update and deleting.
Pre-Condition	Login into the system
Post-Condition	
Flow of Events	Primary flow: i. The use case starts when admin go to list of staff. ii. The admin have options to view, update and delete [B1][B2][B3] iii. Use case ends. Alternative flow:

	[B1]
	i. Admin click view icon.
	ii. Display the complete information of the user.
	iii. Admin click close and go back to primary flow.
	[B2]
	i. Admin click update icon and go to update page.
	ii. Update information of the user.
	iii. Click update and validate all the data.
	iv. Update record in database.
	v. Return to the primary flow.
	[B3]
	i. Admin click delete icon.
	ii. Display conformation box.
	iii. Admin click yes.
	iv. Delete record from database.
	v. Return to primary flow

4.3.1.3 Create hibah application

Table 4.6: Use case create hibah application

Use Case Name	Hibah application
Description	This use case describes the process of add new hibah application
Pre-Condition	Client login into the system
Post-Condition	Display new list of the application
Flow of Events	Primary flow: i. The use case starts when client login into the system. ii. The client has options to update and create new hibah application [C1] [C2]. iii. Use case end. Alternative flow: [C1] i. The client click button add new hibah. ii. Validate the hibah application cannot have more than three that have status "Dalam process". [[C11].
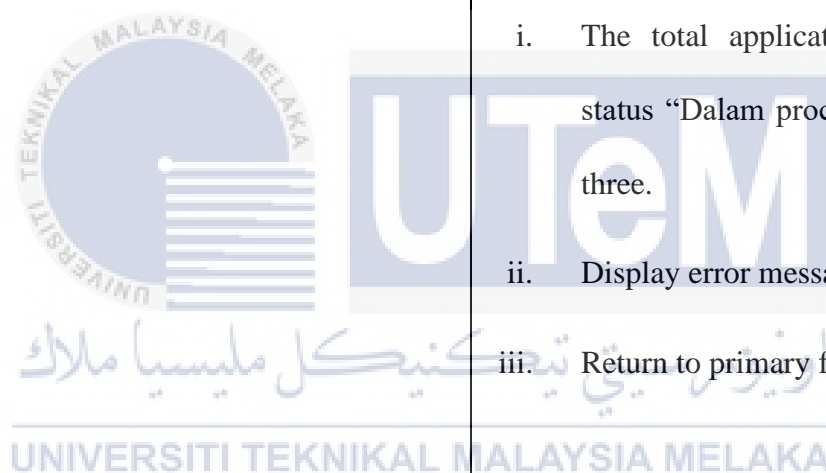
	iii. Add new hibah application in database.
	iv. Return to primary flow.
	[C2]
	i. The client clicks button update.
	ii. Go to instruction page.
	Exception flow:
	[C11]
	i. The total application that has status "Dalam process" equal to three.
	ii. Display error message.
	iii. Return to primary flow.

Table 4.7: Use case giver information
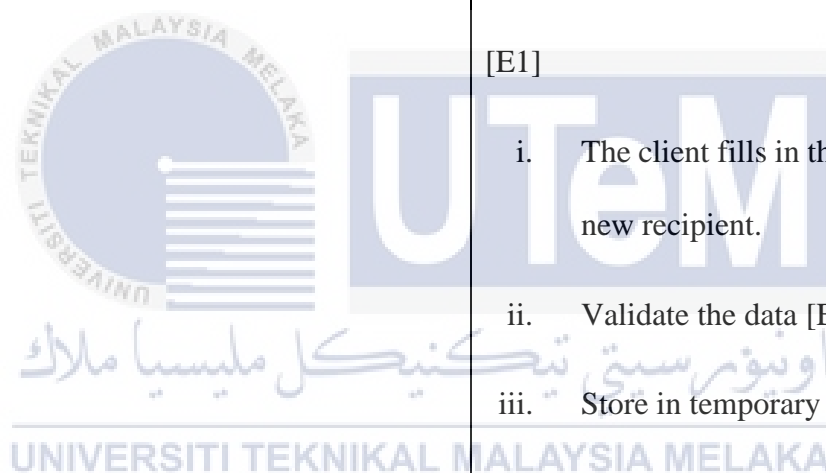
Use Case Name	Manage giver information
Description	This use case describes the process to add or update the giver information.
Pre-Condition	Login into the system and create the hibah application

Post-Condition	Go to recipient page
Flow of Events	Primary flow: i. This use case starts when user click button from instruction page. ii. The client fills in all the giver details. iii. Click button submit. iv. Validate all data. [D11] v. [D12] [D13]. vi. Use case end. Alternative flow: [D12] i. Giver data already exist in database. ii. Update new record for giver information. iii. Return to primary flow. [D13] i. Giver data do not exist in database.

	ii. Record new giver information.
	iii. Return to primary flow.
	Exception flow:
	[D11]
	i. The client enters invalid data for email, phone number and file format.
	ii. Display error message.
	iii. Return to primary flow.

Table 4.8: Use case recipient information

Use Case Name	Manage recipient information
Description	This use case describes the process to add or update the giver information.
Pre-Condition	i. Login into the system. ii. Create the hibah application. iii. Filled in giver information.
Post-Condition	Update recipient data
Flow of Events	Primary flow:

	i. The use case starts when the client click button from menu or from giver page.
	ii. The clients have options to add, view, update and delete the recipient [E1] [E2] [E3] [E4] [E5].
	iii. Use case end.
	Alternative flow:
	[E1]
	i. The client fills in the form to add new recipient.
	ii. Validate the data [E11].
	iii. Store in temporary list.
	iv. Return to primary flow.
	[E2]
	i. The client clicks reset button.
	ii. Delete list of recipients from temporary list.
	iii. Return to primary flow.
	[E3]
	i. The client click view icon.
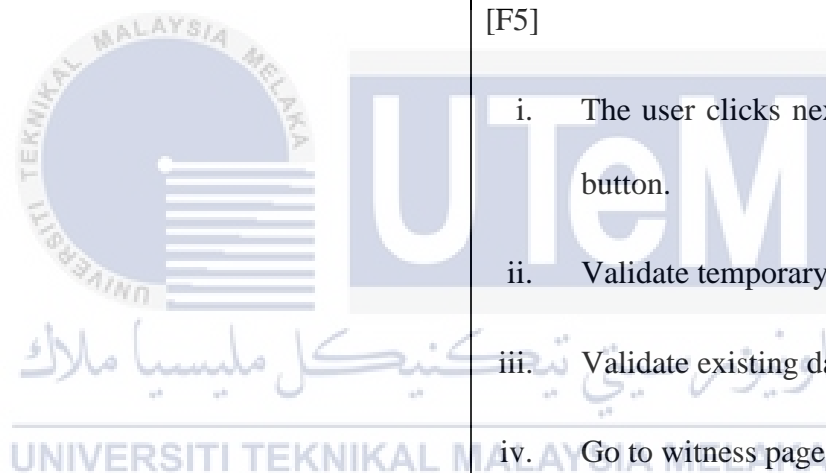
	ii. Display details information for the recipient.
	iii. Return to primary flow.
	[E4]
	i. The user click delete icon.
	ii. Delete the recipient information from database.
	iii. Return to primary flow.
	[E5]
	i. The user clicks next and submit button.
	ii. Validate temporary list [E12].
	iii. Validate existing data [E13].
	iv. Go to property page.
	Alternative flow:
	[E12]
	i. Have data in temporary list.
	ii. Add all data from the temporary list into the database.
	Exception flow:
	[E11]

	i. The user enters wrong or empty field.
	ii. Display error message.
	iii. Return to primary flow.
	[E13]
	i. Do not have data in temporary list and also table.
	ii. Display error message.
	iii. Back to primary flow.

Table 4.9: Use case property information

Use Case Name	Manage Property information
Description	This use case describes the process of add, update, view and delete the property information.
Pre-Condition	i. Login into the system
	ii. Filled up giver and recipient information.
Post-Condition	Update property information
Flow of Events	Primary flow:
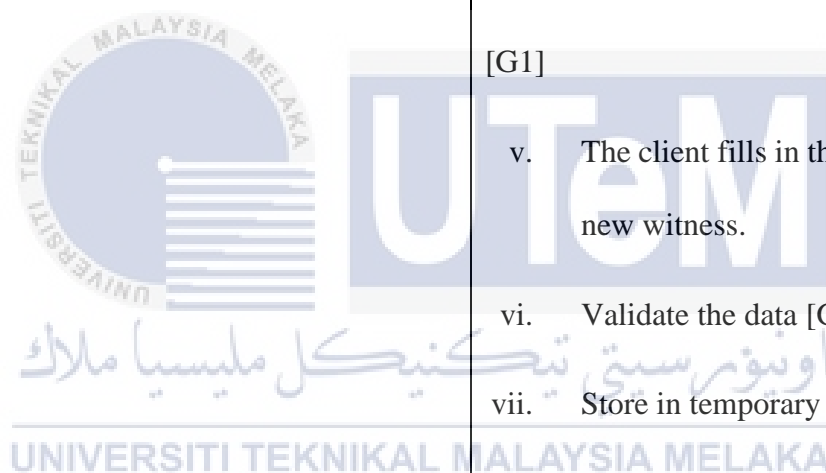
	i. The use case starts when the client click button from menu or from recipient page.
	ii. The clients have options to add, view, update and delete the property [F1] [F2] [F3] [F4] [F5].
	iii. Use case end.
	Alternative flow:
	[F1]
	i. The client fills in the form to add new property.
	ii. Validate the data [F11].
	iii. Store in temporary list.
	iv. Return to primary flow.
	[F2]
	i. The client clicks reset button.
	ii. Delete list of recipients from temporary list.
	iii. Return to primary flow.
	[F3]
	i. The client click view icon.
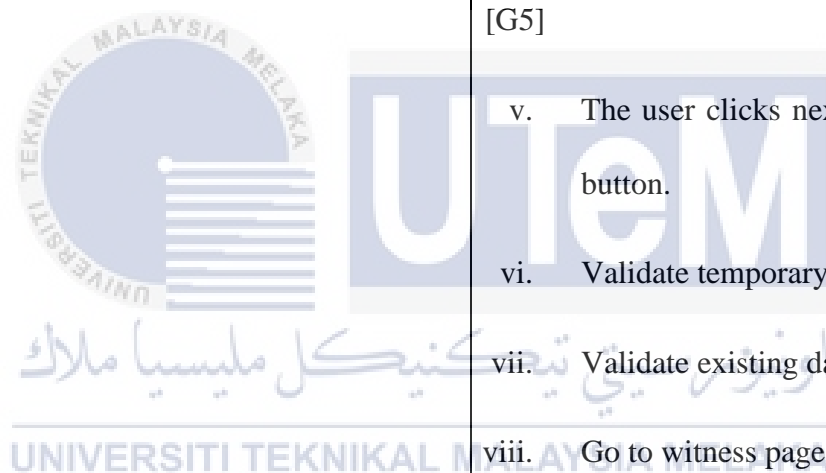
	ii. Display details information for the property.
	iii. Return to primary flow.
	[F4]
	i. The user click delete icon.
	ii. Delete the property information from database.
	iii. Return to primary flow.
	[F5]
	i. The user clicks next and submit button.
	ii. Validate temporary list [F12].
	iii. Validate existing data [F13].
	iv. Go to witness page.
	Alternative flow:
	[F12]
	i. Have data in temporary list.
	ii. Add all data from the temporary list into the database.
	Exception flow:
	[F11]

	i. The user enters wrong or empty field.
	ii. Display error message.
	iii. Return to primary flow.
	[F13]
	i. Do not have data in temporary list and also table.
	ii. Display error message.
	iii. Back to primary flow.

Table 4.10: Use case witness information

Use Case Name	Manage Witness information
Description	This use case describes the process of add, update, view and delete the witness information.
Pre-Condition	iii. Login into the system iv. Filled up giver, recipient and property information.
Post-Condition	Update witness information
Flow of Events	Primary flow:

	iv. The use case starts when the client click button from menu or from property page.
	v. The clients have options to add, view, update and delete the witness [G1] [G2] [G3] [G4] [G5].
	vi. Use case end.
	Alternative flow:
	[G1]
	v. The client fills in the form to add new witness.
	vi. Validate the data [G11].
	vii. Store in temporary list.
	viii. Return to primary flow.
	[G2]
	iv. The client clicks reset button.
	v. Delete list of recipients from temporary list.
	vi. Return to primary flow.
	[G3]
	iv. The client click view icon.

	v. Display details information for the witness.
	vi. Return to primary flow.
	[G4]
	iv. The user click delete icon.
	v. Delete the witness information from database.
	vi. Return to primary flow.
	[G5]
	v. The user clicks next and submit button.
	vi. Validate temporary list [G12].
	vii. Validate existing data [G13].
	viii. Go to witness page.
	Alternative flow:
	[G12]
	iii. Have data in temporary list.
	iv. Add all data from the temporary list into the database.
	Exception flow:
	[G11]

	iv. The user enters wrong, empty field or total witness equal to two.
	v. Display error message.
	vi. Return to primary flow.
	[G13]
	iv. Do not have data in temporary list and also table.
	v. Display error message.
	vi. Back to primary flow.

Table 4.11: Use case payment information

Use Case Name	Manage Payment information
Description	This use case describes the process of the payment.
Pre-Condition	i. Login into the system
	ii. Filled up giver, recipient, property and witness information.
Post-Condition	

Flow of Events	Primary flow:
	i. The use case starts when the client goes to payment page.
	ii. The client has options to make payment and go to status page. [J1] [J2].
	iii. Use case end.
	Alternative flow:
	[J1]
	i. Validate if the client does not have success payment or do not make one [J11].
	ii. Click button pay.
	iii. Go to payment gateway toyyib pay.
	iv. Get return URL value.
	v. Record the value in database.
	vi. Return to primary flow.
	[J2]
	i. The client click button next and submit.

	ii. Validate if the payment already made and success [J12].
	iii. Go to status page.
	Exception flow:
	[J11]
	i. Already made payment.
	ii. Return to primary flow.
	[J12]
	i. Do not have payment or the payment not success.
	ii. Return to primary flow.

Table 4.12: Use case status application
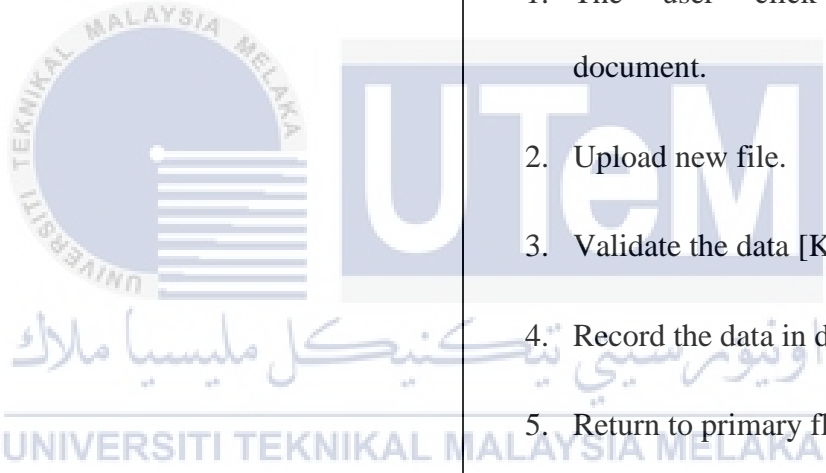
Use Case Name	Status application
Description	This use case describes the process of generate the hibah application document.
Pre-Condition	i. Login into the system ii. Filled up giver, recipient, property, witness and payment information.

Post-Condition	
Flow of Events	Primary flow: i. The use case starts when user click status, or the application already been approved by the MAINS. ii. The client can download the hibah application document and see status of the application. iii. Use case end.

4.3.1.4 Manage hibah application

Table 4.13: Use case manage document (manual)

Use Case Name	Manage document (manual)
Description	This use case describes the process to add, view, update and delete document (manual).
Pre-Condition	Login as admin into the system.
Post-Condition	
Flow of Events	Primary flow:

	1. The use case starts when the user goes to manual document page.
	2. The user has options to add, view and delete the document [K1] [K2] [K3].
	3. Use case end.
	Alternative flow:
	[K1]
	1. The user click add new document.
	2. Upload new file.
	3. Validate the data [K12].
	4. Record the data in database.
	5. Return to primary flow.
	[K2]
	1. The user clicks the view icon and go to view details of the document.
	2. The user has option to update the document [K4].
	3. Return to primary flow.

	[K3]
	1. The user click delete icon.
	2. Confirmation message display.
	3. Delete record from the database.
	4. Return to primary flow.
	[K4]
	1. The user upload new document.
	2. Validate the file [K11].
	3. Update the document information in database.
	4. Return to primary flow.
	Exception flow:
	[K11]
	1. The user upload wrong format for the file.
	2. Display error message.
	3. Return to [K4].
	[K12]
	1. The user upload wrong format for the file.

| | 2. Display error message. |
| | 3. Return to primary flow. |

Table 4.14: Use case manage officer

Use Case Name	Manage officer
Description	This use case describes the process to update the officer's information that be used in generating the document.
Pre-Condition	Login as admin into the system.
Post-Condition	Go to list of officer page
Flow of Events	Primary flow: 1. The use case starts when the user goes to update page. 2. The user has option to cancel and update the officer information [L1] [L2]. 3. Use case end. Alternative flow: [L1]
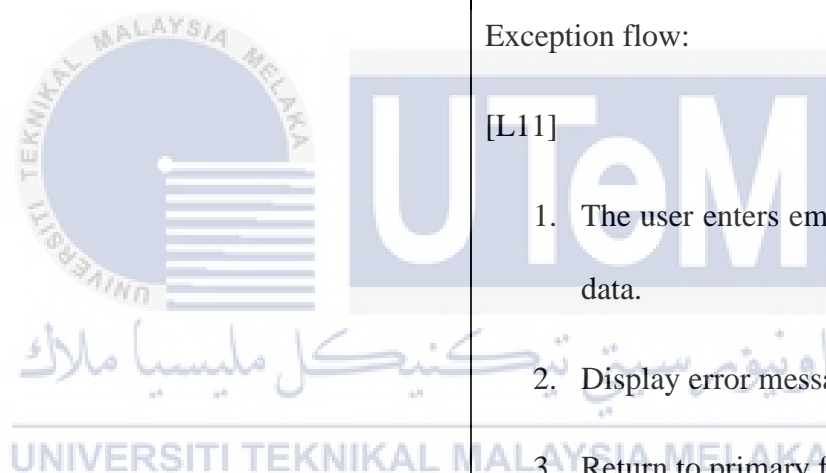
	1. The user inserts new information for the officer.
	2. Validate data [L11].
	3. Update the record in database.
	4. Return to primary flow.
	[L2]
	1. The user click cancel button.
	2. Go to list of officers.
	Exception flow:
	[L11]
	1. The user enters empty or invalid data.
	2. Display error message.
	3. Return to primary flow.

Table 4.15: Use case process hibah

Use Case Name	Process hibah
Description	This use case describes for process that involve during the process of hibah application

Pre-Condition	i. Staff or admin login into the system. ii. Client made the payment and success.
Post-Condition	
Flow of Events	Primary flow: i. The use case starts when the user clicks "Seterusnya" button in list of hibah application. ii. Validate the current status of the application. iii. The user only can go to the specific process step by step or the process that already done. iv. Use case end.

Table 4.16: Use case application review

Use Case Name	Application review
Description	This use case describes the process of the admin or staff review the hibah application before to accept or reject.
Pre-Condition	1. Login into the system

	2. Have data in URL.
Post-Condition	
Flow of Events	Primary flow: i. The use case starts when the user click link from process hibah and go to review page. ii. After reviewing all the information and documents, the user has options to accept or reject the application for user to correct the mistakes [M1] [M2]. iii. The user can click back button redirect to list of process page. iv. Use case end. Alternative flow: [M1] i. The user chooses to accept the application.
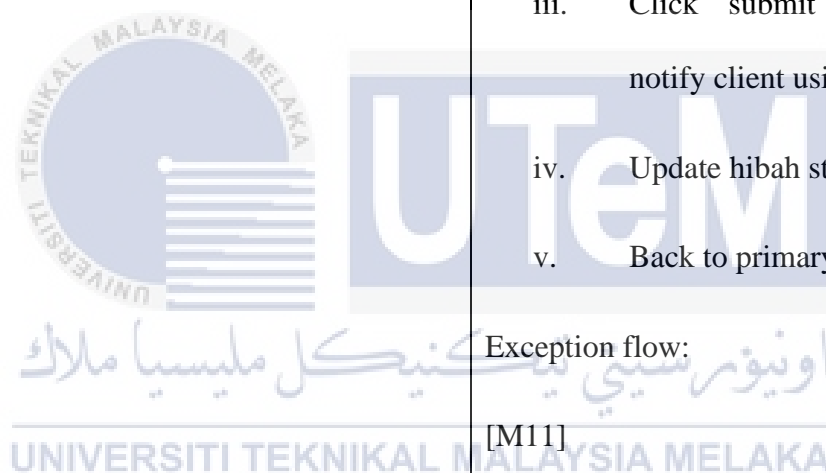
	ii. Click submit button and notify client using email.
	iii. Back to primary flow.
	[M2]
	i. The user chooses to reject the application and enter message for the user.
	ii. Validate the data [M11].
	iii. Click submit button and notify client using email.
	iv. Update hibah status.
	v. Back to primary flow.
	Exception flow:
	[M11]
	i. The user leaves empty field for the message.
	ii. Display error message.
	iii. Back to primary flow.

Table 4.17: Use case generate agreement

Use Case Name	Generate agreement

Description	This use case describes the process of generating the agreement document.
Pre-Condition	i. Login into the system. ii. The application already been accepted by the MAINS. iii. Have data in URL.
Post-Condition	Update hibah status
Flow of Events	Primary flow: i. The use case starts when the user click link to the generate agreement page from the process page. ii. System will automate generating the agreement document and store the data in folder and record the file name in database. iii. User can download the document by click the link. iv. Use case end.
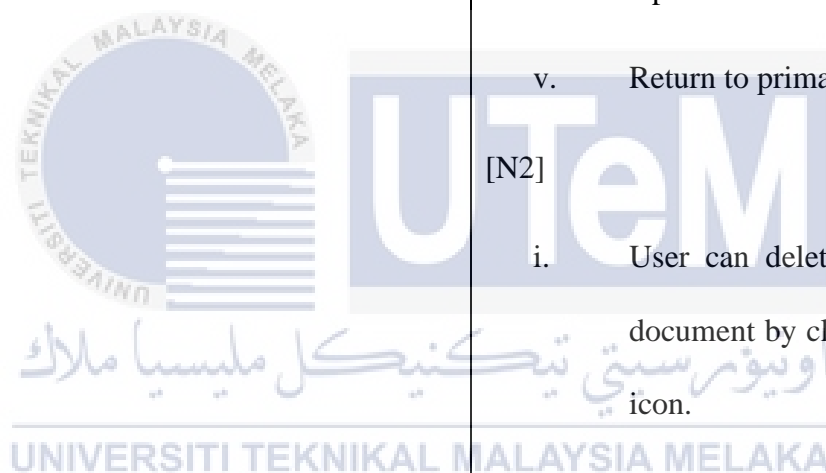
Table 4.18: Use case generate requesting signature letter

Use Case Name	Generate requesting signature letter
Description	This use case describes the process of generating the requesting signature letter.
Pre-Condition	i. Login into the system. ii. Success generating the agreement document process. iii. Have data in URL.
Post-Condition	Update hibah status
Flow of Events	Primary flow: i. The use case starts when the user click link to the generate the document page from the process page. ii. System will automate generating the requesting signature letter document and store the data in folder and record the file name in database.

| | iii. | User can download the document by click the link. |
| | iv. | Use case end. |

Table 4.19: Use case upload agreement document

Use Case Name	Upload agreement document
Description	This use case describes the process of uploading the agreement document.
Pre-Condition	i. Login into the system. ii. Success generating the requesting signature letter document process. iii. Have data in URL.
Post-Condition	
Flow of Events	Primary flow: i. The use case starts when the user clicks the link from the process page. ii. The user has option to add new document or delete the current document [N1] [N2]. iii. Use case end.

	Alternative flow:
	i. User can add new document by click the add new document button and will go to add document page [N11].
	ii. Upload the document and validate[N12].
	iii. Record the data in database.
	iv. Update hibah status.
	v. Return to primary flow.
	[N2]
	i. User can delete the current document by click the delete icon.
	ii. Display the conformation message.
	iii. Delete record from database and internal folder.
	iv. Return to primary flow.
	Exception flow:
	[N11]

	i. Documents exist in database and internal folder.
	ii. Display error message cannot add new document.
	iii. Return to primary flow.
	[N22]
	i. The user uploads the wrong file format.
	ii. Display error message.
	iii. Return to primary flow.

Table 4.20: Use case generate court date letter
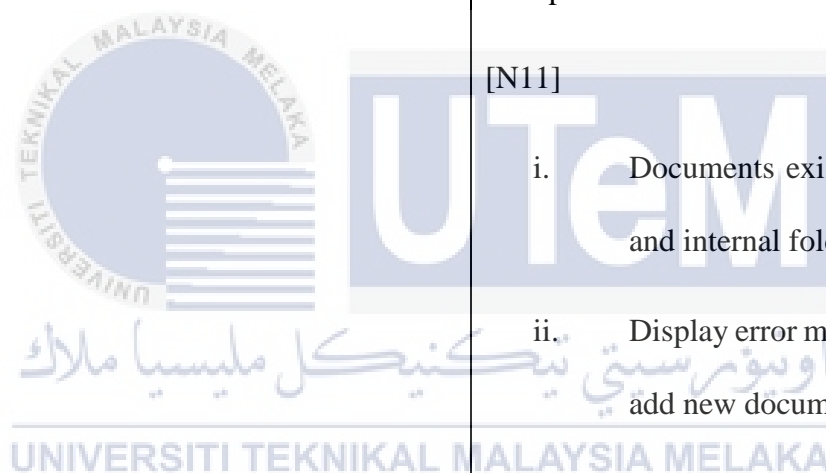
Use Case Name	Generate court date letter
Description	This use case describes the process of generating the court letter.
Pre-Condition	i. Login into the system.
	ii. The user has uploaded the agreement document.
	iii. Have data in URL.
Post-Condition	Update the hibah status
Flow of Events	Primary flow:

	i.	The use case starts when the user click link to the generate the document page from the process page.
	ii.	System will automate generating the court date letter document and store the data in folder and record the file name in database.
	iii.	User can download the document by click the link.
	iv.	Use case end.

Table 4.21: Use case upload court instruction

Use Case Name	Upload court instruction
Description	This use case describes the process of uploading the document of court instruction.
Pre-Condition	i. Login into the system. ii. The user has generated the court document. iii. Have data in URL.

Post-Condition	

Flow of Events	Primary flow:
	
	    iv. The use case starts when the user clicks the link from the process page.
	
	    v. The user has option to add new document or delete the current document [O1] [O2].
	
	    vi. Use case end.
	
	Alternative flow:
	
	[O1]
	
	    i. User can add new document by click the add new document button and will go to add document page [O11].
	
	    ii. Upload the document and validate[O12].
	
	    iii. Record the data in database.
	
	    iv. Update hibah status.
	
	    v. Return to primary flow.
	
	[O2]

	i. User can delete the current document by click the delete icon.
	ii. Display the conformation message.
	iii. Delete record from database and internal folder.
	iv. Return to primary flow.
	Exception flow:
	[N11]
	i. Documents exist in database and internal folder.
	ii. Display error message cannot add new document.
	iii. Return to primary flow.
	[N22]
	i. The user uploads the wrong file format.
	ii. Display error message.
	iii. Return to primary flow.

4.3.1.5 Manage report

Table 4.22: Use case overall report

Use Case Name	Overall report
Description	This use case describes the process of generating the report and convert the report into excel format.
Pre-Condition	Login into the system.
Post-Condition	
Flow of Events	Primary flow: 1. The use case starts when the user goes to overall report page. 2. Display list of the report. 3. The user has an option to convert the report into the excel format. Automatic download after click excel button. 4. Use case end

Table 4.23: Use case report by status

Use Case Name	Report by status

Description	This use case describes the process of generating the report and convert the report into excel format.
Pre-Condition	Login into the system.
Post-Condition	
Flow of Events	Primary flow: 1. The use case starts when the user goes to overall report page. 2. Display list of the report. 3. The user can select the option based on status and click the search button. 4. Update selected data in table. 5. The user has an option to convert the report into the excel format. Automatic download after click excel button. 6. Use case end

Table 4.24: Use case report by range date

Use Case Name	Report by range date

Description	This use case describes the process of generating the report and convert the report into excel format.
Pre-Condition	Login into the system.
Post-Condition	
Flow of Events	Primary flow: 1. The use case starts when the user goes to overall report page. 2. Display list of the report. 3. The user can select start date and end date and click submit button. 4. Update selected data in table. 5. The user has an option to convert the report into the excel format. Automatic download after click excel button. 6. Use case end

4.3.2   Physical Database Design

Table 4.25: Data Definition Language

CREATE TABLE `documents` (

```
 `document_id` int(11) NOT NULL,

 `document_type` varchar(30) NOT NULL,

 `document_path` varchar(80) NOT NULL,

 `document_date` date NOT NULL,

 `staff_id` int(11) NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--

-- Table structure for table `giver`

--

CREATE TABLE `giver` (

 `giver_id` int(11) NOT NULL,

 `giver_name` varchar(80) NOT NULL,

 `giver_ic` varchar(15) NOT NULL,

 `giver_passport` varchar(15) NOT NULL,

 `giver_citezenship` varchar(15) NOT NULL,

 `giver_gender` varchar(15) NOT NULL,

 `giver_status` varchar(15) NOT NULL,

 `giver_race` varchar(15) NOT NULL,

 `giver_religion` varchar(15) NOT NULL,

 `giver_permanent_address` varchar(80) NOT NULL,
```

```
 `giver_mailing_address` varchar(80) NOT NULL,

 `giver_phone_number_r` varchar(15) NOT NULL,

 `giver_phone_number_h` varchar(15) NOT NULL,

 `giver_employer_name` varchar(80) NOT NULL,

 `giver_employer_address` varchar(80) NOT NULL,

 `giver_path_ic` varchar(30) NOT NULL,

 `hibah_id` int(11) NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--

-- Table structure for table `hibah`

--

CREATE TABLE `hibah` (

 `hibah_id` int(11) NOT NULL,

 `hibah_reference` varchar(80) DEFAULT NULL,

 `hibah_date` date NOT NULL,

 `hibah_submit` date DEFAULT NULL,

 `hibah_status` varchar(80) NOT NULL,

 `hibah_code` int(11) NOT NULL DEFAULT '0',

 `user_id` int(11) NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
--

-- Table structure for table `payment`

--

CREATE TABLE `payment` (

  `payment_id` int(11) NOT NULL,

  `payment_reference` varchar(80) DEFAULT NULL,

  `payment_status` varchar(30) DEFAULT NULL,

  `payment_billcode` varchar(30) DEFAULT NULL,

  `payment_amount` varchar(30) DEFAULT NULL,

  `payment_date` varchar(20) DEFAULT NULL,

  `hibah_id` int(11) NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--

-- Table structure for table `position`

--

CREATE TABLE `position` (

  `position_id` int(11) NOT NULL,

  `position_name` varchar(80) NOT NULL,

  `position_person` varchar(80) NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
--

-- Table structure for table `property`

--

CREATE TABLE `property` (

 `property_id` int(11) NOT NULL,

 `property_type` varchar(15) NOT NULL,

 `property_number` varchar(15) NOT NULL,

 `property_pt_number` varchar(15) NOT NULL,

 `property_mukim` varchar(15) NOT NULL,

 `property_district` varchar(15) NOT NULL,

 `property_width` varchar(15) NOT NULL,

 `property_status` varchar(15) NOT NULL,

 `property_instruction` varchar(80) NOT NULL,

 `property_path` varchar(80) NOT NULL,

 `hibah_id` int(11) NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--

-- Table structure for table `recipient`

--

CREATE TABLE `recipient` (
```

```
`recipient_id` int(11) NOT NULL,

`recipient_name` varchar(80) NOT NULL,

`recipient_ic` varchar(15) NOT NULL,

`recipient_phone_number` varchar(15) NOT NULL,

`recipient_address` varchar(80) NOT NULL,

`recipient_relationship` varchar(15) NOT NULL,

`recipient_ic_path` varchar(80) NOT NULL,

`hibah_id` int(11) NOT NULL

--

-- Table structure for table `staff`

--

CREATE TABLE `staff` (

`staff_id` int(11) NOT NULL,

`staff_name` varchar(80) NOT NULL,

`staff_id_number` varchar(15) NOT NULL,

`staff_ic_number` varchar(15) NOT NULL,

`staff_phone_number` varchar(15) NOT NULL,

`staff_email` varchar(80) NOT NULL,

`staff_type` varchar(5) NOT NULL,

`staff_password` varchar(80) NOT NULL,
```

```
 `staff_gender` varchar(11) NOT NULL,

 `staff_confirmation` int(11) DEFAULT '0',

 `staff_pic_path` varchar(30) DEFAULT 'default.png'

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--

-- Table structure for table `updatehibah`

--

CREATE TABLE `updatehibah` (

 `update_id` int(11) NOT NULL,

 `update_name` varchar(30) NOT NULL,

 `update_description` varchar(30) NOT NULL,

 `update_date` date NOT NULL,

 `hibah_id` int(11) NOT NULL,

 `staff_id` int(11) NOT NULL

--

-- Table structure for table `user`

--

CREATE TABLE `user` (

 `user_id` int(11) NOT NULL,

 `user_name` varchar(80) NOT NULL,
```

```
 `user_email` varchar(80) NOT NULL,

 `user_phone_number` varchar(20) NOT NULL,

 `user_password` varchar(255) NOT NULL,

 `keyactivation` varchar(80) NOT NULL,

 `user_pic_path` varchar(30) DEFAULT 'default.png',

 `confirmation` int(11) NOT NULL DEFAULT '0'

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--

-- Table structure for table `user_document`

--

CREATE TABLE `user_document` (

 `document_id` int(11) NOT NULL,

 `document_date` date NOT NULL,

 `document_path` varchar(80) NOT NULL,

 `hibah_id` int(11) NOT NULL,

 `document_name` varchar(80) NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--

-- Table structure for table `witness`

--
```

```
CREATE TABLE `witness` (

 `witness_id` int(11) NOT NULL,

 `witness_name` varchar(80) NOT NULL,

 `witness_ic` varchar(15) NOT NULL,

 `witness_address` varchar(80) NOT NULL,

 `witness_phone_number` varchar(30) NOT NULL,

 `witness_ic_path` varchar(80) NOT NULL,

 `hibah_id` int(11) NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--

-- Indexes for table `documents`

--

ALTER TABLE `documents`

 ADD PRIMARY KEY (`document_id`),

 ADD KEY `FK_Staff_id` (`staff_id`);

--

-- Indexes for table `giver`

--

ALTER TABLE `giver`

 ADD PRIMARY KEY (`giver_id`),
```

```
 ADD KEY `hibah_id` (`hibah_id`);

--

-- Indexes for table `hibah`

--

ALTER TABLE `hibah`

 ADD PRIMARY KEY (`hibah_id`),

 ADD KEY `user_id` (`user_id`);

--

-- Indexes for table `payment`

--

ALTER TABLE `payment`

 ADD PRIMARY KEY (`payment_id`),

 ADD KEY `FK_Payment_ID` (`hibah_id`);

--

-- Indexes for table `position`

--

ALTER TABLE `position`

 ADD PRIMARY KEY (`position_id`);



--
```

```
-- Indexes for table `property`

--

ALTER TABLE `property`

  ADD PRIMARY KEY (`property_id`),

  ADD KEY `hibah_id` (`hibah_id`);

--

-- Indexes for table `recipient`

--

ALTER TABLE `recipient`

  ADD PRIMARY KEY (`recipient_id`),

  ADD KEY `hibah_id` (`hibah_id`);

--

-- Indexes for table `staff`

--

ALTER TABLE `staff`

  ADD PRIMARY KEY (`staff_id`),

  ADD UNIQUE KEY `staff_id_number` (`staff_id_number`);

--

-- Indexes for table `updatehibah`

--
```

```
ALTER TABLE `updatehibah`

 ADD PRIMARY KEY (`update_id`),

 ADD KEY `staff_id` (`staff_id`),

 ADD KEY `hibah_id` (`hibah_id`);

--

-- Indexes for table `user`

--

ALTER TABLE `user`

 ADD PRIMARY KEY (`user_id`);

--

-- Indexes for table `user_document`

--

ALTER TABLE `user_document`

 ADD PRIMARY KEY (`document_id`),

 ADD KEY `hibah_id` (`hibah_id`);

--

-- Indexes for table `witness`

--

ALTER TABLE `witness`

 ADD PRIMARY KEY (`witness_id`),
```

```
  ADD KEY `hibah_id` (`hibah_id`);

--

-- AUTO_INCREMENT for dumped tables

--

--

-- AUTO_INCREMENT for table `documents`

--

ALTER TABLE `documents`

  MODIFY `document_id` int(11) NOT NULL AUTO_INCREMENT,
AUTO_INCREMENT=15;

--

-- AUTO_INCREMENT for table `giver`

--

ALTER TABLE `giver`

  MODIFY `giver_id` int(11) NOT NULL AUTO_INCREMENT,
AUTO_INCREMENT=17;

--

-- AUTO_INCREMENT for table `hibah`

--

ALTER TABLE `hibah`
```

```
  MODIFY    `hibah_id`    int(11)    NOT    NULL    AUTO_INCREMENT,
AUTO_INCREMENT=523;

--

-- AUTO_INCREMENT for table `payment`

--

ALTER TABLE `payment`

 MODIFY    `payment_id`    int(11)    NOT    NULL    AUTO_INCREMENT,
AUTO_INCREMENT=7;

--

-- AUTO_INCREMENT for table `position`

--

ALTER TABLE `position`

 MODIFY    `position_id`    int(11)    NOT    NULL    AUTO_INCREMENT,
AUTO_INCREMENT=4;

--

-- AUTO_INCREMENT for table `property`

--

ALTER TABLE `property`

 MODIFY    `property_id`    int(11)    NOT    NULL    AUTO_INCREMENT,
AUTO_INCREMENT=28;
```

```
--

-- AUTO_INCREMENT for table `recipient`

--

ALTER TABLE `recipient`

 MODIFY `recipient_id` int(11) NOT NULL AUTO_INCREMENT,
AUTO_INCREMENT=15;

--

-- AUTO_INCREMENT for table `staff`

--

ALTER TABLE `staff`

 MODIFY `staff_id` int(11) NOT NULL AUTO_INCREMENT,
AUTO_INCREMENT=16;

--

-- AUTO_INCREMENT for table `updatehibah`

--

ALTER TABLE `updatehibah`

 MODIFY `update_id` int(11) NOT NULL AUTO_INCREMENT,
AUTO_INCREMENT=12;

--

-- AUTO_INCREMENT for table `user`

--
```

```
ALTER TABLE `user`

 MODIFY `user_id` int(11) NOT NULL AUTO_INCREMENT,
AUTO_INCREMENT=40;

--

-- AUTO_INCREMENT for table `user_document`

--

ALTER TABLE `user_document`

 MODIFY `document_id` int(11) NOT NULL AUTO_INCREMENT,
AUTO_INCREMENT=13;

--

-- AUTO_INCREMENT for table `witness`

--

ALTER TABLE `witness`

 MODIFY `witness_id` int(11) NOT NULL AUTO_INCREMENT,
AUTO_INCREMENT=15;

--

-- Constraints for dumped tables

--

--

-- Constraints for table `documents`

--
```

```sql
ALTER TABLE `documents`

  ADD CONSTRAINT `FK_Staff_id` FOREIGN KEY (`staff_id`) REFERENCES

`staff` (`staff_id`);

--

-- Constraints for table `giver`

--

ALTER TABLE `giver`

  ADD CONSTRAINT `giver_ibfk_1` FOREIGN KEY (`hibah_id`)

REFERENCES `hibah` (`hibah_id`);

--

-- Constraints for table `hibah`

--

ALTER TABLE `hibah`

  ADD CONSTRAINT `hibah_ibfk_1` FOREIGN KEY (`user_id`) REFERENCES

`user` (`user_id`);

--

-- Constraints for table `payment`

--

ALTER TABLE `payment`

  ADD CONSTRAINT `FK_Payment_ID` FOREIGN KEY (`hibah_id`)

REFERENCES `hibah` (`hibah_id`);
```

```
--

-- Constraints for table `property`

--

ALTER TABLE `property`

  ADD CONSTRAINT `property_ibfk_1` FOREIGN KEY (`hibah_id`)
REFERENCES `hibah` (`hibah_id`);

--

-- Constraints for table `recipient`

--

ALTER TABLE `recipient`

  ADD CONSTRAINT `recipient_ibfk_1` FOREIGN KEY (`hibah_id`)
REFERENCES `hibah` (`hibah_id`);

--

-- Constraints for table `updatehibah`

--

ALTER TABLE `updatehibah`

  ADD CONSTRAINT `updatehibah_ibfk_1` FOREIGN KEY (`staff_id`)
REFERENCES `staff` (`staff_id`),

  ADD CONSTRAINT `updatehibah_ibfk_2` FOREIGN KEY (`hibah_id`)
REFERENCES `hibah` (`hibah_id`);
```

```
--

-- Constraints for table `user_document`

--

ALTER TABLE `user_document`

  ADD CONSTRAINT `user_document_ibfk_1` FOREIGN KEY (`hibah_id`)
REFERENCES `hibah` (`hibah_id`);

--

-- Constraints for table `witness`

--

ALTER TABLE `witness`

  ADD CONSTRAINT `witness_ibfk_1` FOREIGN KEY (`hibah_id`)
REFERENCES `hibah` (`hibah_id`);

COMMIT;
```

4.4   Conclusion

As a result, this section focuses primarily on web server architecture. The diagrams
are created to showcase the system's architecture description. The database
architecture section includes an entity-relationship diagram and a data dictionary to
contain information about the relationship between the tables that have been produced
to protect the data. The Data Dictionary is a collection of data types, content, format,
and characteristics used to store information and descriptions.

CHAPTER 5: IMPLEMENTATION

5.1   Introduction

This chapter deals with the system's implementation. System implementation is deciding how the system will be built and maintaining its operation and usage. It is critical to guarantee that the system meets quality requirements. The objective is to deploy the system to a select number of users and then incorporate the technology within the company for continuing support and maintenance.

The five modules of the Hibah Management System are authentication, manage staff, manage user, manage hibah, and manage report. Physical software and hardware must be constructed in logical architecture since each of these components performs a defined purpose.

5.2   Software Development Environment setup

During the execution of this stage, the software that is fully utilized is Laragon. This software or application are strictly required to get access to phpMyAdmin. Laragon is an open-sources cross-platform web server that is essentially required to get accessibility on the database. This system was developed using the same platform, Visual Studio Code, and used localhost access through the Laragon application.
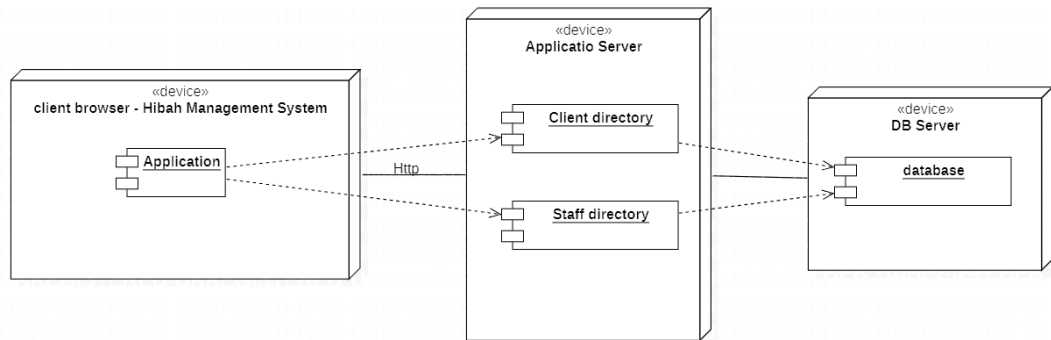
Figure 5.1: Deployment diagram

5.3   Software Configuration Management

5.3.1   Configuration environment setup

Table 5.1: Configuration environment setup

Working directory	Directory purpose
C:\laragon\www\hibah-management-system	The data source for the system is the folder in the Laragon folder called "hibah-management system." The system's data in the system, including system data, files, and pictures, will be retrieved from this folder. This folder contains source code, interface code, graphic, and other system-related material that will be saved and stored in this folder. This folder will also verify that the correct code displays in the web browser and that the system functions properly.
C:\Users\Syafiqah\AppData\Roaming\Microsoft\Windows\Start	This directory is Integrated Development Environment (IDE) that used to develop the system, It can manage all the codes that used in

Menu\Programs\Visual Studio Code	the system development such as HTML, CSS, JavaScript, and php. Visual Studio Code have many features that useful in the development process.

5.3.2 Version Control Procedure

Version control is a critical method in which every development and progress can track and record the system's development is correct, and the iteration is proceeding as planned. Version control allows to maintain track of projects and readily investigate the changes made, whether the data, code scripts or notes. Version control is considerably smoother and easier to apply using version control software like Git. This project uses Github as an online platform to store files, as figure5.1, which implies that the data are backed up online. Github also has tools that allow viewing the history of any file, making it simple to see how it has changed over time. It also allows to keep track of work and effortlessly browse between the many versions of files have written, all while keeping an online backup.
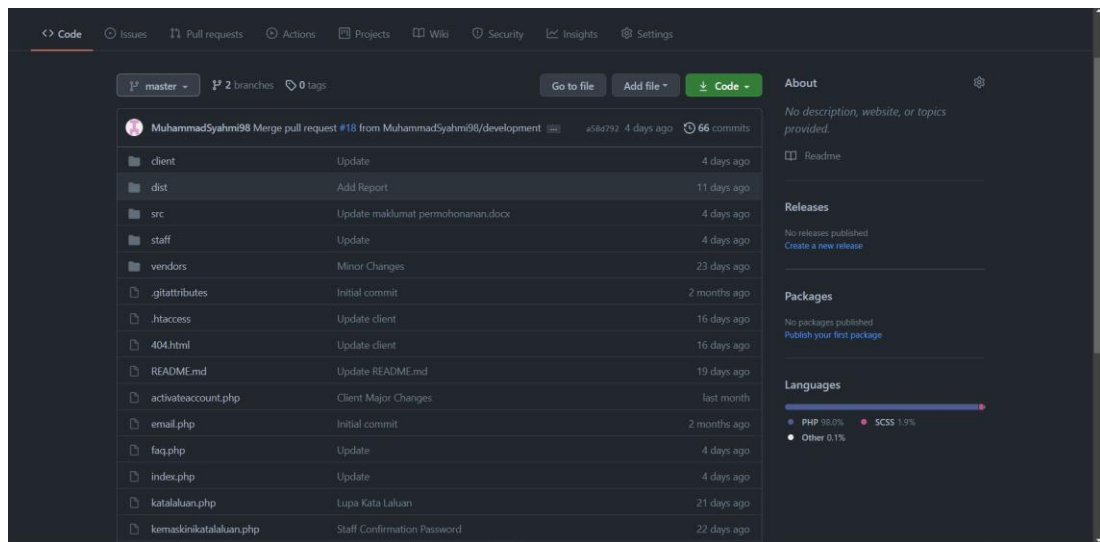
Figure 5.2: Github interface

5.4   Implementation Status

The creation of the Hibah Management System followed the capstone timeframe perfectly, and the implementation status demonstrates that the development is proceeding according to the initial plan.

Module	Description	Duration	Date Complete	Size
Authentication	This module consists of login page, register page for client, forgot password page and reset password page.	5 days	14 Planning and Analysis April 2021	
Create hibah application	This module consists of dashboard, instructions, giver, recipient, property witness, payment gateway and status page. With lot of pages that	21 days	5 May 2021	

	need to develop. This module requires more time to be completed.			
Manage user	This module is for add, view and update user data. Admin can add new staff, manage their data, and also view the full information. Staff only can view profile and update some information only. Client can view profile and update the information.	5 days	12 May 2021	
Manage hibah application	This module uses for staff and admin to manage the application that been completed by the client. Users can manage the document that been used in manual and automatic generated document. They can process hibah application and generated related document. During the process, they need to follow step by step process. This	18 days	30 May 2021	

	module require a lot time to completed because have many sub components to develop.			
Manage report	This module only provide report by overall, status and range date. Staff and admin can generate report from the system. It can view at system or they can download it to excel format.	3 days	3 June 2021	

5.5 Conclusion

The implementation stage or phase is necessary and critical for the effective development of the Hibah Management System. The systematic approach for building the system environment setup and versioning control ensures that the system is built precisely under the requirements, mainly the time range and outcome after the development. Because of the multitude of potential risks that might divert the attention and concentration of the development process, the implementation stage is supposed to be recognized as the most crucial operation.

Correct installation and setup procedures assist the project to prevent technical difficulties with the software program, which can slow down development. It can reduce the amount of time spent redoing and reinstalling software if a technical issue arises that cannot be solved.

CHAPTER 6: TESTING

6.1   Introduction

Any project's development will require a testing phase to evaluate the completed project to ensure that it adheres to the initial plan and particular compliances. The testing method will divide into stages, identifying any flaws in the system source code, such as bugs and errors. If an error discovers during the testing step, the process will continue with the fixing process.

Software testing is a type of testing used in information technology, specifically for system development. The procedure to verify that the project can complete with the minor errors possible in functional and non-functional criteria. This method will meet all possible error occurrences in order to limit the probability of software error.

This project employs three different types of testing: test plan, test strategy, and test design. All test results and verdicts are kept on record. The recorded result will be used as a guide to correct any errors that may have occurred.

6.2   Test Plan

A test plan is a thorough document that specifies the aim, resources, and processes involved in testing a given hardware or software product. A test plan is a

piece of paper used to collect data on the testing results during the testing process. The test organization, test environment, and test schedule are all involved in this procedure.

6.2.1 Test Organization

The organization will highlight the role of each individual involved in the Hibah Management System testing phase. Integration testing, system testing, and user testing are the four essential testing methods employed during the testing. To avoid bias evaluation, the testing, as mentioned earlier, will be carried out by a different person. Because of the Covid-19 epidemic, some of the testing will be done manually by the developer. The individual listed in table 6.1 will be in charge of the testing operation.

Table 6.1: Person in Charge for Each Testing

Tester ID	Responsible Person	Type of Testing
TID001	Muhammad Syahmi Bin Abdul Jalil	Unit testing
		Integration testing
		System testing
TID002	Officer MAINS	User acceptance testing

6.2.2 Test Environment

The technique for evaluating and testing the system environment is known as the test environment. The testing field encompasses equipment, software, and the system's network. Environment testing is necessary for the developer to comprehend the real-world circumstances of the client's working environment and the system environment. The test was running on a server that supports the application. In this situation, It is

using cPanel hosting to make this app online. The tester can access the system by clicking on the HMS link. The server configuration shown in table 6.2 is the one I used during the environmental testing.

Table 6.2: Server Configuration

Server Configuration	Specification
cPanel version	96.0 (build 15)
Apache version	2.4.48
Php version	7.2.34
MySQL version	10.3.30-MariaDB-cll-lve
Architecture	x86_64
Operating system	Linux
Kernel Version	4.18.0-147.8.1.el7h.lve.1.x86_64

6.2.3 Test Schedule

The process flow of the testing procedure will be defined by the test schedule below. Module, test cycle, duration, start and end dates of the test are among the details.

Table 6.3: Test Schedule

Module	Test Cycle	Duration	Start Date	End Date
Login module	4	10 minutes	22 August 2020	22 August 2020

Registration	4	10 minutes	22 August 2020	22 August 2020
Create hibah application	4	15 minutes	22 August 2020	22 August 2020
Manage profile	4	5 minutes	22 August 2020	22 August 2020
Manage staff	4	10 minutes	22 August 2020	22 August 2020
Manage client	4	10 minutes	22 August 2020	22 August 2020
Manage hibah application	4	15 minutes	22 August 2020	22 August 2020
Manage report	4	10 minutes	22 August 2020	22 August 2020

6.3 Test Strategy

The test strategy, or how the test will be carried out, is critical for reducing the test's complexity and difficulty. In order to get a favorable and effective outcome, the proper and appropriate approach or plan must be chosen. The plan must also be feasible and justifiable.

The following is an example of a high-potential-for-success approach. The use of the following method increased the likelihood of accurate and exact testing.

I. White box testing

The information on the code is used to do white-box testing. The developer can view the program code to test its components. This type of testing is beneficial in assisting the tester and developer identify any errors in the code.

II.   Black box testing

Black box testing is a technique for determining user acceptability and the functionality of the system. The Hibah Management System is tested as a black box in this proposal. The information about the internal structure is not used consistently. Validation testing at this step entails assessing the plan or portion during or after the execution stage to determine whether it satisfied the specified requirements.

6.3.1   Classes of tests

The number of testing procedures accessible across the world is incalculable. However, the developer's and expertise's advanced technology led them to build and implement a new testing procedure. A tester can use a variety of testing techniques. The tester and developer's testing approach aids them in executing and performing unit testing, integration testing, and system testing.

I.   Unit testing

This particular testing activity related to the project was evaluated and reviewed by the tester to ensure the software properly run the system accordingly to the requirements required.

II.   Integration testing

This particular testing activity was first carried out in order to verify the system's functionality across all of the project's applications.

III.   System testing

The system testing is primarily carried out to evaluate each project module's operation and to establish if the system is functioning effectively and, as a result, whether the system can process needed data appropriately.

IV.   Acceptance testing

The most frequent testers perform to assess the system's acceptability and adaptability is user acceptance testing. Testing guarantees that the development meets the customer's needs and adheres to the regulations.

6.4   Test Design

6.4.1   Test Description

I.   Test case register user

Test ID	Test Case	Expected Result
TC_01_01	Register by leaving the name field empty	Failed with error message
TC_01_02	Register by leaving the email field empty	Failed with error message
TC_01_03	Register by leaving the phone number field empty	Failed with error message
TC_01_04	Register by password the name field empty	Failed with error message
TC_01_05	Register with different password for password and re-password field	Failed with error message
TC_01_06	Register by wrong phone number format	Failed with error message

TC_01_07	Register with valid data	User should get email verification

II.   Test case login user

Test ID	Test Case	Expected Result
TC_02_01	Login with email and password field empty	Failed with error message
TC_02_02	Login with valid email and invalid password	Failed with error message
TC_02_03	Login with invalid email and valid password	Failed with error message
TC_02_04	Login with valid email and password	User should login into the system

III.   Test cases add new hibah application

Test ID	Test Case	Expected Result
TC_03_01	Not have any application and click add application button	Success and system will create new hibah application
TC_03_02	Have 3 application with status 'Dalam permohonan' and click add application button	Failed with error message
TC_03_03	Click 'Lihat'	Success and go to other page.

IV.   Test case client profile

Test ID	Test Case	Expected Result

TC_04_01	Update profile with document that does not have photo extensions and valid data for name, email and phone number field. Exp photo extensions: jpeg, png, jpg	Failed with error message
TC_04_02	Update profile with empty field.	Failed with error message
TC_04_03	Update profile with no changes of the data	Success
TC_04_04	Update profile with changes of the data	Success

V.   Test case giver

Test ID	Test Case	Expected Result
TC_05_01	Add giver with empty fields	Failed with error message
TC_05_02	Add giver with wrong format of identification card document.	Failed with error message
TC_05_03	Add giver with valid data and empty field for passport, name and address of the employer	Success and go to receiver page.
TC_05_04	Add giver with wrong format for phone number	Failed with error message

VI.   Test case receiver

Test ID	Test Case	Expected Result
TC_05_01	Click 'Simpan Dan Seterusnya' button without add receiver information	Failed with error message
TC_05_02	Add receiver with all the fields empty	Failed with error message
TC_05_03	Add receiver with some of the field empty	Failed with error message
TC_05_04	Add receiver with valid data	Success. Display the data in table receiver
TC_05_05	Add receiver with not valid data for document in 'Dokumen Salinan kad pengenalan' field. Exp: png, jpeg	Failed with error message
TC_05_06	Click 'Reset' button	Success and delete all the new receiver data from the table

VII. Test case property

Test ID	Test Case	Expected Result
TC_06_01	Click 'Simpan Dan Seterusnya' button to go to witness page without have any data display in the property table information	Failed with error message
TC_06_02	Add property with all the input fields empty	Failed with error message

TC_06_03	Add property with some of the input fields empty	Failed with error message
TC_06_04	Add property with valid data for all input fields except 'Dokumen Salinan Harta' field has invalid data. Exp: jpg, jpeg, zip	Failed with error message
TC_06_05	Add property with valid data for all input fields	Success and display the information in the table
TC_06_06	Click 'Reset' button	Success and delete all the new property data from the table

VIII.   Test case witness

Test ID	Test Case	Expected Result
TC_07_01	Click 'Simpan Dan Seterusnya' button without have any data in the table	Failed with error message
TC_07_02	Add witness with have empty input fields	Failed with error message
TC_07_03	Add witness with valid data for all input fields except 'Dokumen Salinan Kad Pengenalan' field has invalid data. Exp: jpg, jpeg, zip	Failed with error message
TC_07_04	Add witness with valid data	Success and display the information in the table

TC_07_05	Add witness with total witness in table have 2 persons	Failed with error message
TC_07_06	Click 'Reset' button	Success and delete all the new property data from the table

IX.   Test case payment

Test ID	Test Case	Expected Result
TC_08_01	Status payment: Belum Selesai Click 'Bayar Disini' button and finish the operation with valid data	Success and will redirect to payment gateway
TC_08_02	Status payment: Belum Selesai Click 'Bayar Disini' button and finish the operation with invalid data	Failed with success message
TC_08_03	Status payment: Belum Selesai/Tidak Berjaya Click 'Seterusnya' button	Failed with error message
TC_08_04	Status payment: Bayaran Diterima Click 'Seterusnya' button	Success and go to status page
TC_08_05	Status payment: Bayaran Diterima	Failed. Can't click button

	Click 'Bayar Disini' button	

X.   Test case client status

Test ID	Test Case	Expected Result
TC_09_01	Download 'Salinan Dokumen Permohonan'	Can download and have information about the hibah application

XI.   Test case staff profile

Test ID	Test Case	Expected Result
TC_10_01	Update profle by empty phone number or email input fields	Failed with error message
TC_10_02	Update profile with image document format zip, doc	Failed with error message
TC_10_03	Update profile with image document format jpeg, png or jpg	Success
TC_10_04	Update profile with invalid data for phone number or email	Failed with error message
TC_10_05	Update profile with valid data	Success

XII.   Test cases add staff

Test ID	Test Case	Expected Result

TC_11_01	Add staff with empty input fields	Failed with error message
TC_11_02	Add staff with some of the input fields empty	Failed with error message
TC_11_03	Add staff with valid data except email	Failed with error message
TC_11_04	Add staff with IC number that already registered	Failed with error message
TC_11_05	Add staff with staff ID that already registered	Failed with error message
TC_11_06	Click 'Batal' button and to cancel the registration	Success and all data in input field will be remove
TC_11_07	Add staff with valid data	Success and will redirect to list of staff

XIII. Test cases edit staff information

Test ID	Test Case	Expected Result
TC_12_01	Update staff with empty fields	Failed with error message
TC_12_02	Update staff with some of the input fields empty	Failed with error message
TC_12_03	Update staff with valid data except email	Failed with error message
TC_12_04	Update staff with IC number that already registered for other staff	Failed with error message
TC_12_05	Update staff with staff ID that already	Failed with error message

	registered for other staff	
TC_12_06	Update 'Batal' button and to cancel the registration	Success and redirect to list of staff page
TC_12_07	Update staff with valid data	Success and will redirect to list of staff

XIV.   Test cases manage client account

Test ID	Test Case	Expected Result
TC_13_01	Click 'activate' button that have status active user	Failed with error message
TC_13_02	Click 'activate' button that have status inactive user	Success and the status will change to active

XV.   Test cases manage document hibah(manual)

Test ID	Test Case	Expected Result
TC_14_01	Add document with format extension jpg, jpeg, zip or png	Fail with error message
TC_14_02	Add document with format extension docx or pdf	Success and redirect to list of documents (manual)
TC_14_03	Add document in the input field and click 'Batal' button to cancel it.	Success and the document in the input field will be remove

TC_14_04	Click eye icon to go to edit page	Success and redirect to the edit page
TC_14_05	Update document with the document that already exist in the system	Failed with error message
TC_14_06	Update document with empty field	Failed with error message
TC_14_07	Click dustbin icon to remove the document	Success and remove the data

XVI. Test cases manage officer information

Test ID	Test Case	Expected Result
TC_15_01	Update officer with empty name field	Failed with error message
TC_15_02	Update officer with valid data	Success
TC_15_03	Update officer name with name that have registered	Failed with error message

XVII. Test case hibah application

Test ID	Test Case	Expected Result
Review section		
TC_16_01	Click 'Simpan' button with empty fields	Failed
TC_16_02	Tick 'Ditolak' for 'Status Permohonan' option with empty	Failed with error message

	'Alasan Penolakan' text field and submit	
TC_16_03	Tick 'Ditolak' for 'Status Permohonan' option with valid data 'Alasan Penolakan' text field and submit	Success and system will send email to the user the reason
TC_16_04	Tick 'Diterima for 'Status Permohonan' option with valid data 'Alasan Penolakan' text field and submit	Success and send email to the user
TC_16_05	Tick 'Diterima for 'Status Permohonan' option with empty 'Alasan Penolakan' text field and submit	Success and send email to the user
Generate agreement hibah document section		
TC_16_06	Click 'Muat Turun' link to download the document	Success
TC_16_07	Click 'Kembali' link to back list of process hibah	Success
Generate memo document section		
TC_16_08	Click 'Muat Turun' link to download the document	Success
TC_16_09	Click 'Kembali' link to back list of process hibah	Success
Upload agreement document section		

TC_16_10	Click 'Tambah Dokumen' when no document in the list table	Success and redirect to add page
TC_16_11	Click 'Tambah Dokumen' when have document in the list table	Failed with error message
TC_16_12	Add document with empty field	Failed with error message
TC_16_13	Add document with invalid data	Failed with error message
TC_16_14	Add document with valid data	Success and redirect to table document
TC_16_15	Click 'Muat Turun' link to download document	Success
TC_16_16	Click icon delete to delete the document	Success
Generate court document section		
TC_16_17	Click 'Muat Turun' link to download the document	Success
Upload court document section		
TC_16_18	Click 'Tambah Dokumen' when no document in the list table	Success and redirect to add page
TC_16_19	Click 'Tambah Dokumen' when have document in the list table	Failed with error message
TC_16_20	Add document with empty field	Failed with error message

TC_16_21	Add document with invalid data	Failed with error message
TC_16_22	Add document with valid data	Success and redirect to table document
TC_16_23	Click 'Muat Turun' link to download document	Success
TC_16_24	Click icon delete to delete the document	Success

6.4.2   Test Data

I.   Test data register user

Test ID	Test Data
TC_01_01	Name: Muhammad Syahmi Bin Abdul Jalil Email: Phone No.: Password: Re-password:
TC_01_02	Nama: Muhammad Syahmi Bin Abdul Jalil Email: Phone No.: 019-6399925 Paaword: Qwerty@12 Re-password:Qwerty@12
TC_01_03	Name: Muhammad Syahmi Bin Abdul Jalil Email: syahmijalil12@gmail.com Phone No.: 019-6399925 Passowrd: Qwerty@12 Re-password: Qwerty@12
TC_01_04	Nama: Muhammad Syahmi Bin Abdul Jalil Email: syahmijalil12@gmail.com Phone No.: 019-6399925 Password: Qwerty@12

	Re-password: Qwerty@12
TC_01_05	Nama: Muhammad Syahmi Bin Abdul Jalil
	Email: syahmijalil12@gmail.com
	Phone No.:: 019-6399925
	Password: Qwerty@12
	Re-password: Qwerty@12

II.   Test data login user

Test ID	Test Data
TC_02_01	Email: Kata Laluan:
TC_02_02	Email: syahmijalil12@gmail.com Kata Laluan: Zxcvbn@12
TC_02_03	Email: ahmad@gmail.com Kata Laluan: Qwerty@12
TC_02_04	Email: syahmijalil12@gmail.com Kata Laluan: Qwerty@12

III.   Test data client profile

Test ID	Test Data
TC_04_01	Document: example.docx
TC_04_02	No data
TC_04_03	Original data from database
TC_04_04	Name: Muhammad Syafiq Bin Abdul Jalil Email: syafiq@gmail.com

IV.   Test data giver

Test ID	Test Data

TC_05_01	No data
TC_05_02	Document: example.jpg
TC_05_03	Name: Fatimah Az-zahrah
	IC No.: 600728-06-2323
	Document: IC.pdf
	Citizenship: Malaysia
	Gender: Woman
	Status: Single
	Race: Malay
	Religion: Islam
	Address: Tl 60 Jalan masjid
	Mailing address: Tl 60 Jalan masjid
	Phone no.: 019-7622234
	Phone no.: 06-4532345
TC_05_04	Name: Fatimah Az-zahrah
	IC No.: 600728-06-2323
	Document: IC.pdf
	Citizenship: Malaysia
	Gender: Woman
	Status: Single
	Race: Malay
	Religion: Islam
	Address: Tl 60 Jalan masjid
	Mailing address: Tl 60 Jalan masjid
	Phone no.: 019-7622234456
	Phone no.: test

V.   Test data receiver

Test ID	Test Data
TC_05_01	No data
TC_05_02	No data
TC_05_03	Name: Muhammad Asraf Bin Zul

	IC no.:
	Phone no.:
	Address:
	Relationship:
	Document:
TC_05_04	Name: Muhammad Asraf Bin Zul
	IC no.: 800902-02-5095
	Phone no.: 011-1111234
	Address: Tl 60 Jalan masjid
	Relationship: Anak
	Document: IC.docx
TC_05_05	Name: Muhammad Asraf Bin Zul
	IC no.: 800902-02-5095
	Phone no.: 011-1111234
	Address: Tl 60 Jalan masjid
	Relationship: Anak
	Document: IC.png
TC_05_06	No data

VI.   Test data property

Test ID	Test Data
TC_06_01	No data
TC_06_02	No data
TC_06_03	Property no.: ASDJN8990
	PT no.:
	Mukim:
	District:
	Area: 2 hectare
	Status: Private
	Property instruction: 1/3
	Document: geran.pdf
TC_06_04	Property no.: ASDJN8990

	PT no.: 98756
	Mukim: Serom
	District: Tangkak
	Area: 2 hectare
	Status: Private
	Property instruction: 1/3
	Document: geran.png
TC_06_05	Property no.: ASDJN8990
	PT no.: 98756
	Mukim: Serom
	District: Tangkak
	Area: 2 hectare
	Status: Private
	Property instruction: 1/3
	Document: geran.pdf
TC_06_06	No data

VII. Test data witness

Test ID	Test Data
TC_07_01	No data
TC_07_02	No data
TC_07_03	Name: Abdul Razak Bin Mihat
	IC no.: 700223-05-7635
	Phone no.: 019-8765478
	Address: Tl 32 Jalan Tengah
	Document: ic.png
TC_07_04	Name: Abdul Razak Bin Mihat
	IC no.: 700223-05-7635
	Phone no.: 019-8765478
	Address: Tl 32 Jalan Tengah
	Document: ic.pdf
TC_07_05	No data

TC_07_06	No data

VIII.   Test data payment

Test ID	Test Data
TC_08_01	No data
TC_08_02	No data
TC_08_03	No data
TC_08_04	No data
TC_08_05	No data

IX.   Test data client status

Test ID	Test Data
TC_09_01	No data

X.   Test data staff profile

Test ID	Test Data
TC_10_01	No data
TC_10_02	Pic.docx
TC_10_03	Pic.png
TC_10_04	019-98767899786
TC_10_05	Document: pic2.jpeg Phone no.: 019-7689876 Email: syahmijalil13@gmail.com

XI.   Test data add staff

Test ID	Test Data
TC_11_01	No data
TC_11_02	Name:

	Email: Zulkarnain@gmail.com
	IC no.: 890112-07-7665
	Staff ID:
	Phone no.: 012-6546754
	Gender: Male
	Level: Admin
TC_11_03	Name: Zulkarnain Bin Ahmad
	Email: syahmi.com
	IC no.: 890112-07-7665
	Staff ID: SA02
	Phone no.: 012-6546754
	Gender: Male
	Level: Admin
TC_11_04	Name: Zulkarnain Bin Ahmad
	Email: Zulkarnain@gmail.com
	IC no.: 900802-02-2323
	Staff ID: SA02
	Phone no.: 012-6546754
	Gender: Male
	Level: Admin
TC_11_05	Name: Zulkarnain Bin Ahmad
	Email: Zulkarnain@gmail.com
	IC no.: 890112-07-7665
	Staff ID: SA01
	Phone no.: 012-6546754
	Gender: Male
	Level: Admin
TC_11_06	No data
TC_11_07	Name: Zulkarnain Bin Ahmad
	Email: Zulkarnain@gmail.com
	IC no.: 890112-07-7665
	Staff ID: SA02
	Phone no.: 012-6546754

	Gender: Male
	Level: Admin

XII.   Test data edit staff information

Test ID	Test Data
TC_12_01	No data
TC_12_02	Name: Email: Zulkarnain@gmail.com IC no.: 890112-07-7665 Staff ID: SA02 Phone no.: 012-6546754 Gender: Male Level: Admin
TC_12_03	Name: Zulkarnain Bin Ahmad Email: syahmi.com IC no.: 890112-07-7665 Staff ID: SA02 Phone no.: 012-6546754 Gender: Male Level: Admin
TC_12_04	Name: Zulkarnain Bin Ahmad Email: Zulkarnain@gmail.com IC no.: 890112-07-7665 Staff ID: SA02 Phone no.: 012-6546754 Gender: Male Level: Admin
TC_12_05	Name: Zulkarnain Bin Ahmad Email: Zulkarnain@gmail.com IC no.: 890112-07-7665 Staff ID: SA01 Phone no.: 012-6546754

	Gender: Male
	Level: Admin
TC_12_06	No data
TC_12_07	Name: Zulkarnain Bin Ahmad
	Email: Zulkarnain@gmail.com
	IC no.: 890112-07-7665
	Staff ID: SA02
	Phone no.: 012-6546754
	Gender: Male
	Level: Admin

XIII.   Test data manage client account

Test ID	Test Data
TC_13_01	No data
TC_13_02	No data

XIV.   Test data manage document hibah(manual)

Test ID	Test Data
TC_14_01	Document.png
TC_14_02	Document.pdf
TC_14_03	Document.pngd
TC_14_04	No data
TC_14_05	Document.pdf
TC_14_06	No data
TC_14_07	No data

XV.   Test data manage officer information

Test ID	Test Data
TC_15_01	No data

TC_15_02	Name: Siti Fatimah Binti Abdul Rahman
TC_15_03	Name: Siti Fatimah Binti Abdul Rahman

XVI.   Test data hibah application

Test ID	Test Data
Review section	
TC_16_01	No data
TC_16_02	No data
TC_16_03	Reason: Kesilapan pada bahagian penerima.
TC_16_04	Reason: Diterima.
TC_16_05	No data
Generate agreement hibah document section	
TC_16_06	No data
TC_16_07	No data
Generate memo document section	
TC_16_08	No data
TC_16_09	No data
Upload agreement document section	
TC_16_10	No data
TC_16_11	Document.pdf
TC_16_12	No data
TC_16_13	Document.png
TC_16_14	Document.pdf
TC_16_15	No data
TC_16_16	No data
Generate court document section	
TC_16_17	No data
Upload court document section	

TC_16_18	No data
TC_16_19	No data
TC_16_20	No data
TC_16_21	Document.png
TC_16_22	Document.pdf
TC_16_23	No data
TC_16_24	No data

6.5 Test Results and Analysis

I. Test result register user

Test ID	Tester ID	Test result	Comment
TC_01_01	TID001/TD002	Pass	
TC_01_02	TID001/TD002	Pass	
TC_01_03	TID001/TD002	Pass	
TC_01_04	TID001/TD002	Pass	
TC_01_05	TID001/TD002	Pass	
TC_01_06	TID001/TD002	Pass	
TC_01_07	TID001/TD002	Pass	

II. Test result login user

Test ID	Tester ID	Test result	Comment
TC_02_01	TID001/TD002	Pass	
TC_02_02	TID001/TD002	Pass	
TC_02_03	TID001/TD002	Pass	
TC_02_04	TID001/TD002	Pass	

III. Test results add new hibah application

Test ID	Tester ID	Test result	Comment

TC_03_01	TID001/TD002	Pass	
TC_03_02	TID001/TD002	Pass	
TC_03_03	TID001/TD002	Pass	

IV.   Test result client profile

Test ID	Tester ID	Test result	Comment
TC_04_01	TID001/TD002	Pass	
TC_04_02	TID001/TD002	Pass	
TC_04_03	TID001/TD002	Pass	
TC_04_04	TID001/TD002	Pass	

V.   Test result giver

Test ID	Tester ID	Test result	Comment
TC_05_01	TID001/TD002	Pass	
TC_05_02	TID001/TD002	Pass	
TC_05_03	TID001/TD002	Pass	
TC_05_04	TID001/TD002	Pass	

VI.   Test result receiver

Test ID	Tester ID	Test result	Comment
TC_05_01	TID001/TD002	Pass	
TC_05_02	TID001/TD002	Pass	
TC_05_03	TID001/TD002	Pass	
TC_05_04	TID001/TD002	Pass	
TC_05_05	TID001/TD002	Pass	
TC_05_06	TID001/TD002	Pass	

VII.   Test result property

Test ID	Tester ID	Test result	Comment
TC_06_01	TID001/TD002	Pass	
TC_06_02	TID001/TD002	Pass	
TC_06_03	TID001/TD002	Pass	
TC_06_04	TID001/TD002	Pass	
TC_06_05	TID001/TD002	Pass	
TC_06_06	TID001/TD002	Pass	

VIII.   Test result witness

Test ID	Tester ID	Test result	Comment
TC_07_01	TID001/TD002	Pass	
TC_07_02	TID001/TD002	Pass	
TC_07_03	TID001/TD002	Pass	
TC_07_04	TID001/TD002	Pass	
TC_07_05	TID001/TD002	Pass	
TC_07_06	TID001/TD002	Pass	

IX.   Test result payment

Test ID	Tester ID	Test result	Comment
TC_08_01	TID001/TD002	Pass	
TC_08_02	TID001/TD002	Pass	
TC_08_03	TID001/TD002	Pass	
TC_08_04	TID001/TD002	Pass	
TC_08_05	TID001/TD002	Pass	

X.   Test result client status

Test ID	Tester ID	Test result	Comment
TC_09_01	TID001/TD002	Pass	

XI.   Test result staff profile

Test ID	Tester ID	Test result	Comment
TC_10_01	TID001/TD002	Pass	
TC_10_02	TID001/TD002	Pass	
TC_10_03	TID001/TD002	Pass	
TC_10_04	TID001/TD002	Pass	
TC_10_05	TID001/TD002	Pass	

XII.   Test result add staff

Test ID	Tester ID	Test result	Comment
TC_11_01	TID001/TD002	Pass	
TC_11_02	TID001/TD002	Pass	
TC_11_03	TID001/TD002	Pass	
TC_11_04	TID001/TD002	Pass	
TC_11_05	TID001/TD002	Pass	
TC_11_06	TID001/TD002	Pass	
TC_11_07	TID001/TD002	Pass	

XIII.   Test result edit staff information

Test ID	Tester ID	Test result	Comment
TC_12_01	TID001/TD002	Pass	
TC_12_02	TID001/TD002	Pass	
TC_12_03	TID001/TD002	Pass	
TC_12_04	TID001/TD002	Pass	
TC_12_05	TID001/TD002	Pass	
TC_12_06	TID001/TD002	Pass	
TC_12_07	TID001/TD002	Pass	

XIV.   Test result manage client account

Test ID	Tester ID	Test result	Comment
TC_13_01	TID001/TD002	Pass	
TC_13_02	TID001/TD002	Pass	

XV. Test result manage document hibah(manual)

Test ID	Tester ID	Test result	Comment
TC_14_01	TID001/TD002	Pass	
TC_14_02	TID001/TD002	Pass	
TC_14_03	TID001/TD002	Pass	
TC_14_04	TID001/TD002	Pass	
TC_14_05	TID001/TD002	Pass	
TC_14_06	TID001/TD002	Pass	
TC_14_07	TID001/TD002	Pass	

XVI. Test result manage officer information

Test ID	Tester ID	Test result	Comment
TC_15_01	TID001/TD002	Pass	
TC_15_02	TID001/TD002	Pass	
TC_15_03	TID001/TD002	Pass	

XVII. Test result hibah application

Test ID	Tester ID	Test result	Comment
Review section			
TC_16_01	TID001/TD002	Pass	
TC_16_02	TID001/TD002	Pass	
TC_16_03	TID001/TD002	Pass	
TC_16_04	TID001/TD002	Pass	
TC_16_05	TID001/TD002	Pass	
Generate agreement hibah document section			

TC_16_06	TID001/TD002	Pass	
TC_16_07	TID001/TD002	Pass	
Generate memo document section			
TC_16_08	TID001/TD002	Pass	
TC_16_09	TID001/TD002	Pass	
Upload agreement document section			
TC_16_10	TID001/TD002	Pass	
TC_16_11	TID001/TD002	Pass	
TC_16_12	TID001/TD002	Pass	
TC_16_13	TID001/TD002	Pass	
TC_16_14	TID001/TD002	Pass	
TC_16_15	TID001/TD002	Pass	
TC_16_16	TID001/TD002	Pass	
Generate court document section			
TC_16_17	TID001/TD002	Pass	
Upload court document section			
TC_16_18	TID001/TD002	Pass	
TC_16_19	TID001/TD002	Pass	
TC_16_20	TID001/TD002	Pass	
TC_16_21	TID001/TD002	Pass	
TC_16_22	TID001/TD002	Pass	
TC_16_23	TID001/TD002	Pass	
TC_16_24	TID001/TD002	Pass	

6.6   Conclusion

The testing technique must be carried out to evaluate the system's performance and the development's success. Before handing over the product to the client, this phase determines the project's potency and ensures that the system works correctly and without any flaws. Before a genuine user uses the system in their business, it must be in the best possible shape.

Before the system was implemented, the testing process was carried out carefully and in detail, following a protocol and method to ensure that the system worked without flaws. As a result, the performance is measured by the primary demand as well as specific compliances.

All of the results of the tests were meticulously recorded in a document for future reference. The results will also aid in the correction of deficiencies and errors in the system, allowing it to perform better.

CHAPTER 7: PROJECT CONCLUSION

7.1   Observation on Weaknesses and Strengths

The Hibah Management System has concluded its development cycle. The system's output has been documented during the performance, testing results, and other information throughout the development process and phase. Before the handover procedure with the customer, the developer examined this system to evaluate and identify the system's strengths and weaknesses.

The system's security may be revealed as a vulnerability throughout the observation phase. The system lacks security to prevent unauthorized persons from accessing sensitive documents such as a copy of an identity card directly from the source. The general public cannot utilize the system as a result of this issue. When it comes to sensitive data, the system's security is critical, as previously said.

The second flaw that the system occurred might note is a poor user experience. The present design may make it difficult for elderly users to understand how to utilize the system. They will require more time to comprehend how to use the system properly.

The system's main flaw, in my opinion, is all of the above-listed spots. This fault, for some reason, does not affect the system's performance, and it will continue to function normally despite the flaws. However, based on demand, there is plenty of

opportunity for development in the future. The entire product has been developed effectively under the specifications.

7.2 Propositions for Improvement

Making the system more secure is one of the suggestions for system enhancement. I may accomplish the issue by employing a more specific Laravel framework to decrease security risks. It is due to Laravel's various security capabilities, which allow developers to create applications with a high level of security. For example, cross-site request forgery (CSRF) vulnerabilities are reduced, as is protection against Cross-Site Scripting (XSS), SQL injection, file protection, and other helpful features. As a result, developers may create applications in a short amount of time of excellent quality.

Furthermore, I may improve the user experience of the system. It requires additional in-depth user experience study. The system's attractive user interface does not ensure a positive user experience. The problem may accomplish it by focusing on a specific user persona. It will take up time, but the business owner will benefit.

7.3 Conclusion

The Hibah Management System may be a practical application for MAINS, clients, and employees. Even though this system still has problems and weaknesses, I hope to address the issues. I'm also hoping that everyone who uses this system will be happy and satisfied.

REFERENCES

Sacolick, I. (2020, February 25). *What is agile methodology? Modern software development explained*. InfoWorld. https://www.infoworld.com/article/3237508/what-is-agile-methodology-modern-software-development-explained.html.

User, S. (n.d.). *HIBAH*. Majlis Agama Islam Negeri Sembilan - HIBAH. https://www.mains.gov.my/v1/wakaf/perkhidmatan-wakaf/hibah.

Hibah. Majlis Agama Islam Selangor. (n.d.). https://www.mais.gov.my/hibah/.

Ahmad, K., Majid, R. A., Abdullah, Z., Minhad, S. F., & Shafinar, I. (2017). Acceptance of hibah as an alternative mechanism in Muslims Asset Management. *SHS Web of Conferences*, *36*, 00030. https://doi.org/10.1051/shsconf/20173600030

What is Agile Software Development? Agile Alliance |. (2021, February 9). https://www.agilealliance.org/agile101/.

APPENDICES

Appendix A: Report