FloraHub: Smart Plant Hydration System

SHAUFY YANA BINTI MOHD EZANI

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

FLORAHUB: SMART PLANT HYDRATION SYSTEM

SHAUFY YANA BINTI MOHD EZANI

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

2024

DECLARATION

I hereby declare that this project report entitled

FLORAHUB: SMART PLANT HYDRATION SYSTEM

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

STUDENT : ______________________ Date : 1 JUNE 2024
(SHAUFY YANA BINTI MOHD EZANI)

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 : ______________________ Date : 05 / 09 / 2024
(DR. INTAN ERMAHANI A. JALIL)

DEDICATION

This project is dedicated to Dr. Intan Ermahani. Your guidance and expertise have been invaluable, and I am deeply grateful for your mentorship. Thank you for being an inspiring force throughout this journey. I also dedicate this work to my parents, whose unwavering support and encouragement have been my guiding light. Your faith in my abilities and your constant motivation have been instrumental in helping me achieve my goals. Lastly, I dedicate this project to my friends, whose companionship and encouragement have been a source of strength throughout this journey. Your belief in me has been a powerful motivator. Thank you all for being a part of this journey and for helping me turn this project into a reality.

ACKNOWLEDGEMENTS

I would like to sincerely thank Dr. Intan Ermahani for her helpful advice and help, which have been crucial to the successful completion of this project. Her knowledge and assistance have been invaluable during this process.

I would like to express my sincere gratitude to my parents for their constant inspiration, support, and motivation. Their confidence in my skills has been a continual source of support, motivating me to aim for perfection in each task related to this project.

I also want to express my gratitude to my friends who helped make this project a reality, whether directly or indirectly. Your support and encouragement are much appreciated.

Thank you all for your invaluable contributions to this project.

ABSTRACT

The "FloraHub: Smart Plant Hydration System" project tackles the shortcomings of traditional plant watering methods by proposing an innovative IoT-based solution. Conventional timed watering systems often result in water wastage and inadequate hydration, adversely affecting plant health and increasing expenses. To address these challenges, the project introduces an automated watering system integrated with soil moisture sensors and water flow sensors. These sensors continuously monitor soil moisture levels in real-time, triggering watering only when necessary, while also providing users the option to utilize a timer or tap a button on the mobile app for manual watering. Additionally, the incorporation of Grafana analytics enables comprehensive data analysis, offering insights into soil moisture trends, watering patterns, and water usage. By leveraging technology and data-driven solutions, the project aims to enhance operational efficiency, minimize water consumption, and promote environmentally responsible practices in plant care. The FloraHub system represents a significant advancement in plant management, providing users with a user-friendly and sustainable approach to ensure optimal plant growth and health.

ABSTRAK

Projek "FloraHub: Smart Plant Hydration System" menangani kekurangan kaedah penyiraman tanaman tradisional dengan mencadangkan penyelesaian inovatif berasaskan IoT. Sistem penyiraman berjadual konvensional sering kali menyebabkan pembaziran air dan kekurangan hidrasi yang memberi kesan buruk kepada kesihatan tanaman serta meningkatkan kos. Untuk mengatasi cabaran ini, projek ini memperkenalkan sistem penyiraman automatik yang digabungkan dengan sensor kelembapan tanah dan sensor aliran air. Sensor ini memantau tahap kelembapan tanah secara masa nyata, hanya mengaktifkan penyiraman apabila diperlukan, di samping memberikan pengguna pilihan untuk menggunakan pemasa atau menekan butang pada aplikasi mudah alih untuk penyiraman manual. Selain itu, integrasi analitik Grafana membolehkan analisis data yang komprehensif, menawarkan pandangan mengenai corak kelembapan tanah, corak penyiraman, dan penggunaan air. Dengan memanfaatkan teknologi dan penyelesaian berasaskan data, projek ini bertujuan untuk meningkatkan kecekapan operasi, meminimumkan penggunaan air, dan mempromosikan amalan mesra alam dalam penjagaan tanaman. Sistem FloraHub mewakili kemajuan ketara dalam pengurusan tanaman, menyediakan pendekatan yang mesra pengguna dan mampan untuk memastikan pertumbuhan dan kesihatan tanaman yang optimum.

TABLE OF CONTENTS

PAGE

LIST OF TABLES

PAGE

LIST OF FIGURES

PAGE

LIST OF ABBREVIATIONS

FYP	**-**	**Final Year Project**
IOT	**-**	**Internet Of Thing**
API	**-**	**Application Programming Interface**
PDF	**-**	**Portable Document Format**

CHAPTER 1: INTRODUCTION

1.1 Introduction

Taking care of plants has become challenging in a world where weather patterns are unpredictable and water costs are constantly rising. The traditional method of watering them at predetermined intervals wastes water and increases our expenses. This is where the "FloraHub: Smart Plant Hydration System" becomes useful; it functions similarly to a smart plant assistant.

The regular watering routine is not as effective when the weather is uncooperative. It uses more water than is necessary and is bad for the plants. With the help of unique sensors, FloraHub acts as a smart companion for your plants, ensuring that they get water only when needed. In addition, it monitors water consumption to ensure that nothing is wasted.

1.2 Problem Statement(s)

The main difficulties faced are:

1. The common use of timers for watering plants isn't great. It doesn't consider what the plants and soil need, leading to either too much water or not enough. This can seriously affect the health of the plants.

2. Understanding and improving plant care behaviors is difficult because the current method is unable to offer comprehensive insights into watering patterns.

1.3 Objective

The following are the project's objectives:

1. Replace the current manual timer-based method with an IoT-based automatic watering system.

2. Implement water flow sensors and real-time soil moisture data to enable precise autonomous watering of plants at the right times. This goal calculates the amount of water used to improve plant hydration, minimize water waste, and guarantee sustainable water use.

3. Integrate Grafana analytics to make thorough data analysis easier and offer insightful information.

1.4 Scope

a. Target User

Home Gardeners and Commercial Nurseries

b. Modules to be developed

i. Authentication Module

- **Login:** Users will have a secure login process to access the FloraHub mobile application. This involves entering their registered email and password.
- **Registration:** New users will go through a registration process, providing essential information such as their name, email, and preferred login credentials. This step establishes a personalized account within the application.
- **User Profile:** After logging in, users can access and manage their user profiles. This section allows them to update personal information, adjust notification preferences, and customize their overall experience within the application.

ii. Notification Module

- **Alerts:** To notify users about their plants' status and relevant activities.

iii. **Soil Moisture Module**

- **Sensor Integration:** Soil moisture sensors are embedded in the plant's environment, continuously monitoring moisture levels. These sensors provide real-time data on the soil's hydration status.
- **Threshold-Based Watering:** The system uses real-time soil moisture data to determine when watering is necessary. If moisture levels fall below a set threshold, the automatic watering system is triggered, ensuring plants receive water precisely when needed.
- **Manual Control:** Users have the flexibility to control the watering manually through the mobile app. This includes options for setting timer controls or simply tapping a button in the app to enable watering, providing users with direct and convenient control over their plant hydration.

iv. **Water Flow Module**

- **Sensor Integration:** Water flow sensors are integrated into the irrigation system, measuring the volume of water used during each watering session.
- **Volume Calculation:** Using information from water flow sensor readings, the system intelligently determines the volume of water utilized. It also includes a tracking feature with a cost calculation function. Users can gain a practical understanding of the financial aspects related to their plant care practices by analyzing the volume data and obtaining insights into the corresponding water cost.

v. **Analytics Module**

- **Data Analysis with Grafana:** To make thorough data analysis easier, the system interfaces with Grafana, a potent analytics tool. Grafana allows users to explore soil moisture trends, watering patterns, and water usage through interactive dashboards.
- **Usage Reporting**: In addition to providing insights into real-time data, the Analytics Module offers comprehensive reporting on water usage. Users can access daily, monthly, and yearly reports based on volume calculation, including a cost calculation function. This reporting feature enhances users'

understanding of their plant care practices over time and supports informed decision-making for optimized plant management.

1.5 Project Significance

Several advantages are provided by the "FloraHub: Smart Plant Hydration System" project to various stakeholders, including home gardeners, commercial nurseries, and the environment.

The automation provided by FloraHub technology offers significant benefits to both home gardeners and commercial nurseries. For home gardeners, the system ensures plants receive water precisely when needed, without requiring manual intervention. This guarantees ideal plant health while saving time and effort. Moreover, the incorporation of water flow sensors and real-time soil moisture data minimizes water waste, aligning with water conservation efforts.

Similarly, commercial nurseries can optimize their operations through automation. By replacing manual watering techniques with an IoT-based solution, nurseries can boost productivity and efficiently allocate resources. The system's accurate watering and water usage analysis enables nurseries to reduce water consumption and associated expenses, ultimately enhancing profitability. Additionally, by maintaining healthier plant inventories through timely and precise watering, commercial nurseries can increase customer satisfaction and retention. Thus, FloraHub technology revolutionizes plant care practices for both individual gardeners and commercial enterprises, promoting sustainability and efficiency in plant management.

1.6 Expected Output

The expected outcomes of the project are designed to simplify and improve plant care for users while promoting environmental responsibility.
These are the expected outputs that the user can expect:

1. **Automated Watering:**
   - The system automatically waters plants when they need it, using sensors to detect soil moisture levels. This means no more guesswork or manual watering schedules.
   - By switching to automated watering, users can save water and reduce waste. The system calculates how much water is used, helping users understand their water usage patterns and conserve resources.

2. **User-Friendly Controls:**
   - Users have easy access to the system through a mobile app. They can set watering schedules or manually water their plants with just a few taps on their phone.

3. **Real-time Monitoring and Alerts:**
   - The system continuously monitors soil moisture and sends alerts to users when plants need water. This helps users stay informed about their plants' health and take action promptly.

4. **Data Analysis and Reporting:**
   - The system provides users with insights into soil moisture trends, watering patterns, and water usage through simple and interactive reports. This helps users make informed decisions about their plant care practices.

5. **Environmental Sustainability:**
   - By reducing water waste and promoting efficient watering practices, the system contributes to environmental sustainability. Users can feel good about conserving water and minimizing their environmental impact.

1.7 Conclusion

In conclusion, the "FloraHub: Smart Plant Hydration System" project addresses the challenges faced by home gardeners and commercial nurseries in traditional plant care practices. By automating watering processes and integrating advanced technologies like IoT sensors and data analytics, FloraHub revolutionizes plant care, offering precise watering, operational efficiency, and environmental sustainability.

Home gardeners benefit from the system's ability to ensure plants receive water only when needed, saving time and effort while promoting optimal plant health. Meanwhile, commercial nurseries optimize their operations, reduce water consumption, and enhance customer satisfaction.

The expected outputs of the project simplify plant care for users, providing automated watering, user-friendly controls, real-time monitoring, data analysis, and environmental sustainability. Ultimately, FloraHub empowers users to make informed decisions about their plant care practices while contributing to water conservation efforts and promoting environmental responsibility.

CHAPTER 2: LITERATURE REVIEW AND PROJECT METHODOLOGY

2.1 Introduction

This chapter explores the development process and necessary specifications for the FloraHub: Smart Plant Hydration System. This section offers details on the project methodology that was used, emphasizing the Scrum framework in conjunction with an Agile approach. It also lists the hardware and software prerequisites that must be met for the system to be implemented successfully. These include sensors and Raspberry Pi Pico hardware, as well as development tools like Visual Studio Code and the Eclipse IDE. This chapter lays the groundwork for understanding the methodical approach and resources used in developing the FloraHub system by looking at these important factors.

2.2 Facts and Findings

2.2.1 2.2.1 Existing System

The existing literature provides comprehensive insights into the development and implementation of automated watering systems for plant care, showcasing innovative solutions that leverage IoT technology and micro-controller boards. One notable research study, "**Automatic Watering System for Plants with IoT Monitoring and Notification**" presents a sophisticated prototype designed to optimize plant hydration through real-time monitoring and automated watering processes. This system integrates soil moisture sensors to continuously measure the moisture content in the soil. When the moisture level falls below a predetermined threshold, indicating the need for watering, the system triggers the watering process by activating a solenoid valve to allow water to flow through the irrigation system.

This automated process ensures that plants receive water precisely when needed, thereby promoting optimal growth and health.

Moreover, the integration of IoT platforms such as ThingSpeak enhances the functionality of the watering system by enabling real-time data monitoring and visualization. ThingSpeak serves as a centralized platform for recording and displaying soil moisture data in graphical form, allowing users to track moisture levels remotely. Additionally, the incorporation of Blynk apps facilitates instant notifications to users' smartphones, informing them about the watering process's status. This real-time notification feature ensures that users stay informed and can take timely action to address any plant care needs.

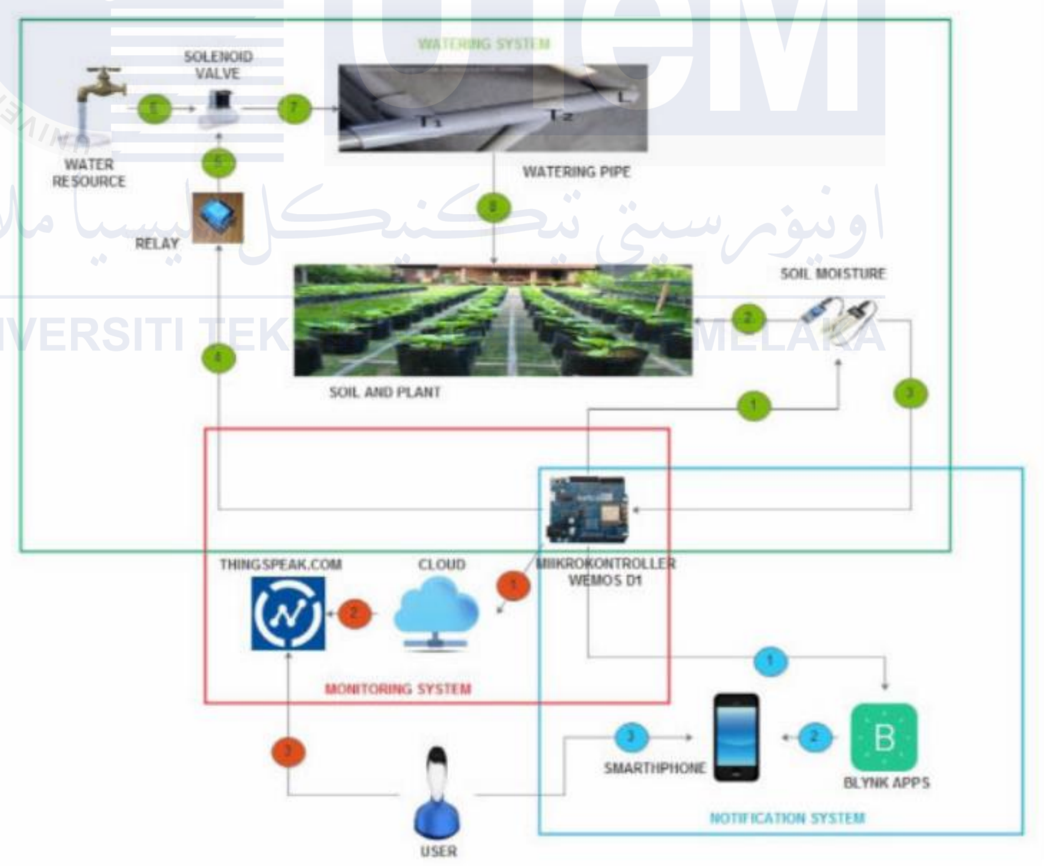


Figure 2.1: The design of the Automatic Watering System

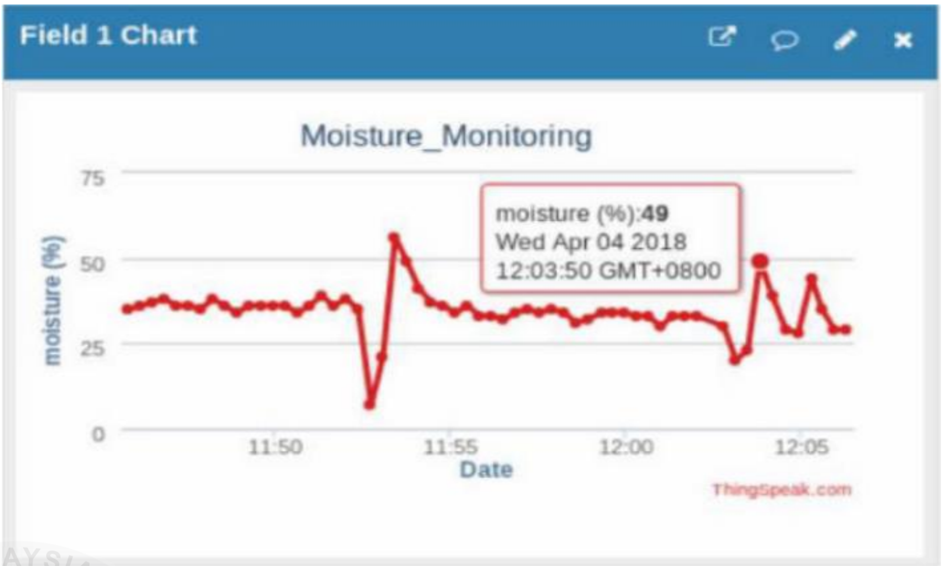


Figure 2.2: ThingSpeak graph monitoring the soil moisture

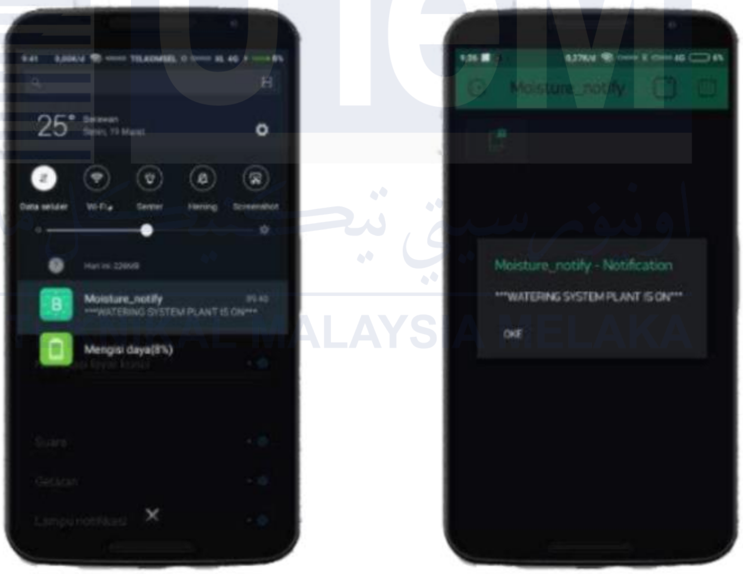


Figure 2.3: Blynk notification when the watering function is on

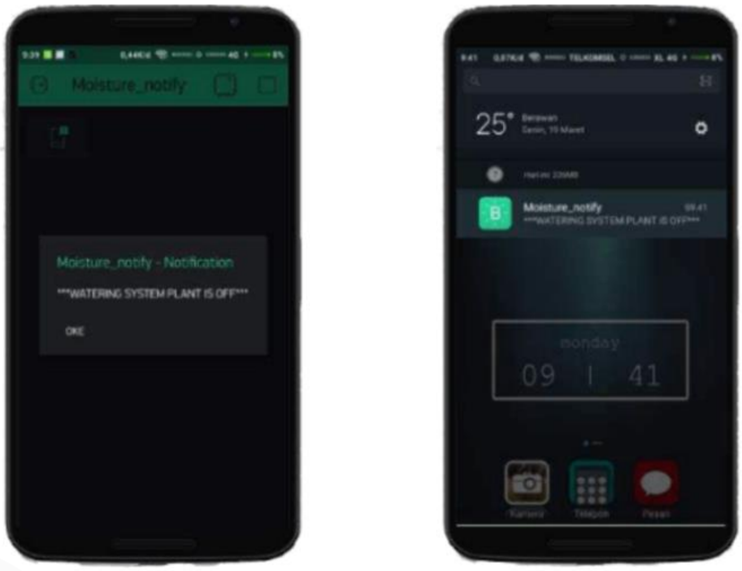


Figure 2.4: Blynk notification when the watering device is Off

Comparison between FloraHub and Existing Automatic Watering System are as follows:

Table 2.1: Comparison between the existing system and the proposed system

No.	Comparison	Existing System	FloraHub
1	Automation	Uses soil moisture sensors, Wemos D1 microcontroller, relay channel, and solenoid valve to automate watering.	Uses Raspberry Pi Pico for automation with soil moisture sensors and integrates threshold-based and manual control.
2	Functionality	Provides real-time soil moisture monitoring and notifications via Blynk and ThingSpeak.	Includes water flow sensors for measuring usage and cost, with advanced data analysis via Grafana.
3	Interaction	Users receive soil moisture notifications via the Blynk app, but have limited manual control and data interaction.	Provides comprehensive control through a FloraHub mobile app, including manual watering and setting schedules.
4	Data Analysis	Limited to basic real-time monitoring and notifications.	Uses Grafana for in-depth analysis of soil moisture trends, watering patterns, and water usage.

5	Reporting	Lacks comprehensive reporting features.	Generates detailed daily, monthly, and yearly water usage reports.
6	Target Users	Aimed at tech-savvy users comfortable with IoT setups.	Targets home gardeners and commercial nurseries, with features for both individual and commercial use.

2.3 Project Methodology

For the project methodology, the FloraHub: Smart Plant Hydration System will be employing the Agile approach, specifically the Scrum framework, which aligns seamlessly with its development objectives. Agile methodology emphasizes iterative development and cross-functional collaboration, even within a single-person project setup. This approach allows for continuous refinement and enhancement of the smart plant hydration system through short, focused iterations.

By adopting the Scrum framework, the project ensures structured development, facilitates regular communication and feedback, and supports efficient adaptation to evolving requirements. These elements are crucial for delivering a solution that optimally addresses the needs of home gardeners and commercial nurseries while promoting efficient plant care practices and water conservation.

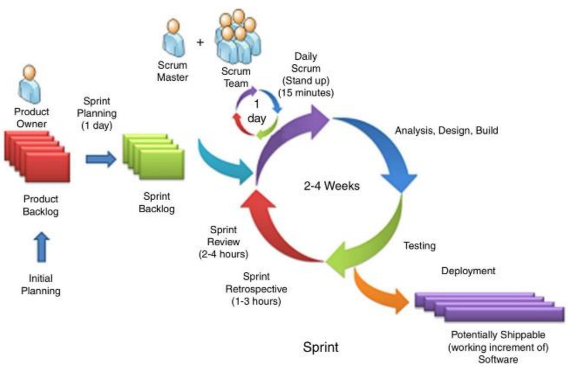


Figure 2.5: Macro Vision of Scrum Project Development

Based on Figure 2.5, in the development process of the FloraHub system, consisting of the developer collaborates closely with a supervisor who provides guidance, sets priorities, and offers feedback throughout the project. At the end of the evaluation phase, the system will undergo an assessment by a designated evaluator.

Each development cycle, known as a sprint, involves collaborative planning sessions with the supervisor to define sprint goals and select tasks from the backlog. Frequent stand-up meetings are conducted to review progress, discuss challenges, and adjust plans accordingly. After each sprint, the completed work is presented to the supervisor for review and feedback. Additionally, retrospective meetings are held to reflect on the sprint's outcomes and identify areas for improvement.

Throughout the development process, a prioritized backlog of tasks and features guides our work, with regular refinement and adjustment in collaboration with the supervisor. Furthermore, the system's final product will undergo evaluation by the designated evaluator based on predefined criteria, informing future iterations and improvements. This structured approach ensures efficient development, alignment with project objectives, and effective communication and collaboration within the team.

2.4 Project Requirements

2.4.1 Software Requirement

- Visual Studio Code

Figure 2.6: Visual Studio Code

This versatile source code editor, Visual Studio Code, provides a streamlined environment for the front-end development of mobile applications using Flutter and Dart language. Its robust features and extensions support coding in multiple languages and frameworks, ensuring efficient and effective development. Additionally, Visual Studio Code is utilized for coding the

Raspberry Pi Pico in Python language, offering a comprehensive development experience for both front-end mobile applications and IoT device programming.

- PhpMyAdmin and MySQL

Figure 2.7: phpMyAdmin

For the database management aspect of the FloraHub system, we used phpMyAdmin, a web-based administration tool bundled with XAMPP, a widely used open-source software stack. XAMPP provides a local server environment for PHP development, and phpMyAdmin serves as a user-friendly interface for managing MySQL databases. This combination offers a convenient and efficient solution for creating, querying, and managing the MySQL database used by the FloraHub system, ensuring smooth data operations and facilitating seamless integration with the system's backend components.

- Eclipse IDE with Spring Boot

Figure 2.8: Eclipse

Eclipse IDE, coupled with the Spring Boot framework, serves as the development environment for building the backend components of the FloraHub system. Eclipse's robust features and Spring Boot's ease of development facilitate the creation of scalable, enterprise-grade backend services in Java, ensuring reliability and performance.

2.4.2 Internet of Things (IoT) Requirement

- Raspberry Pi Pico Microcontroller Board

Figure 2.9: Raspberry Pi Pico W

This figure shows a Raspberry Pi Pico W microcontroller board, which is a core component for the Internet of Things (IoT) set up in the FloraHub system. The Raspberry Pi Pico W is a low-cost, high-performance microcontroller board with flexible digital interfaces.

- Soil moisture sensor

Figure 2.10: Soil Moisture Sensor

This figure illustrates a Soil Moisture Sensor, an essential component for monitoring the moisture levels in the soil of plants. It is used in the FloraHub system to measure the soil's moisture content, providing real-time data that helps in deciding when and how much to water the plants. This sensor ensures that plants receive the appropriate amount of water, optimizing their growth and health.

- Water flow sensor

Figure 2.11: Water Flow Sensor

This figure shows a Water Flow Sensor, which is integral to measuring the flow rate of water in the FloraHub irrigation system. The sensor tracks the amount of water delivered to the plants, ensuring precise control and monitoring of water usage. By providing accurate water flow data, it helps in maintaining efficient water usage and supports the system in generating detailed water consumption reports.

- 2 AA Battery Holder

Figure 2.12: 2 AA Battery Holder

This figure depicts a 2 AA Battery Holder, which is used to power the FloraHub Smart Hydration System. It holds two AA batteries. The battery holder ensures that the system remains portable and can function independently of a direct power source, making it convenient for outdoor and remote plant care.

- Water pump

Figure 2.13: Water Pump

This figure shows a Water Pump, a crucial component of the FloraHub irrigation system. It ensures the plants receive adequate water by pumping it through the irrigation system.

2.5 Project Schedule and Milestone

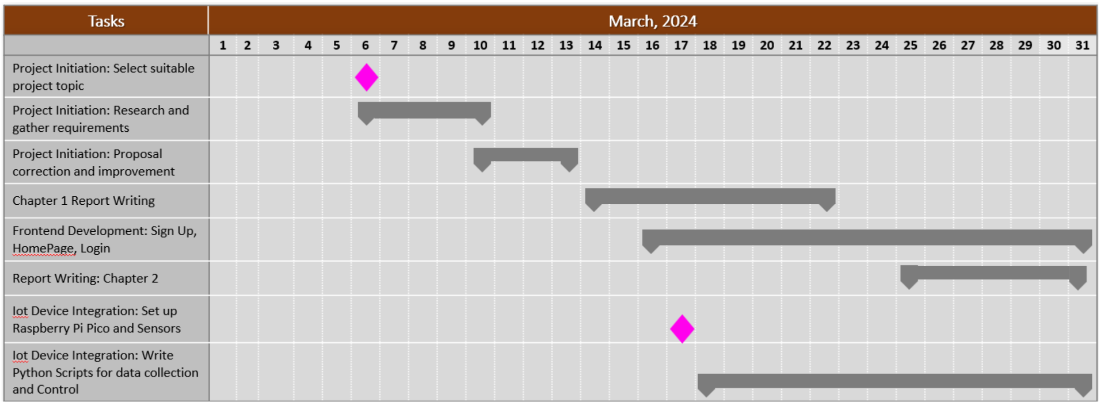


Figure 2.14: Project schedule and milestone for March 2024

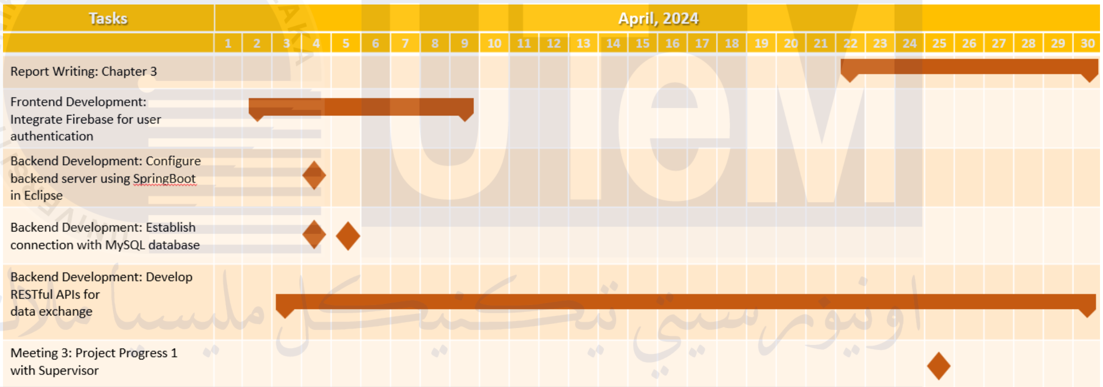


Figure 2.15: Project schedule and milestone for April 2024

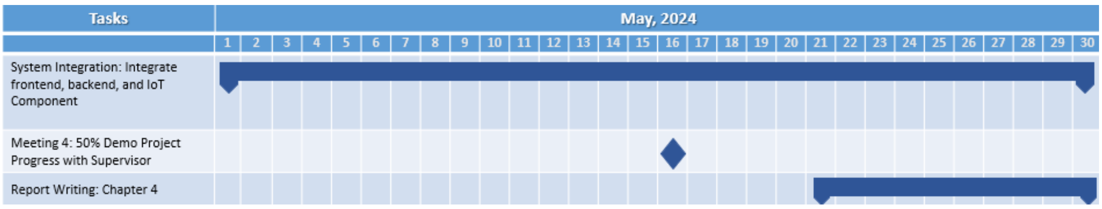


Figure 2.16: Project schedule and milestone for May 2024

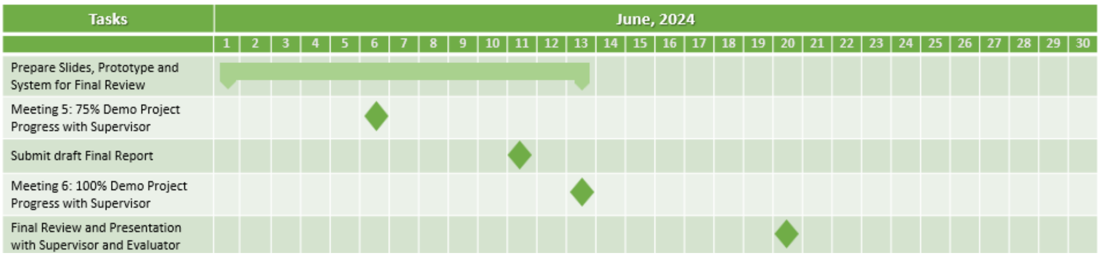


Figure 2.17: Project schedule and milestone for June 2024

2.6 Conclusion

This chapter of the report explores the development process and essential specifications for the FloraHub: Smart Plant Hydration System. It begins by reviewing existing literature and highlighting research on automated watering systems for plant care. Notable studies showcase the effectiveness of IoT technology and micro-controller boards in optimizing plant hydration and addressing water scarcity challenges.

The chapter then discusses the project methodology, emphasizing the Agile approach, particularly the Scrum framework. It outlines the integration of hardware components like soil moisture sensors, Raspberry Pi Pico microcontroller boards, water flow sensors, and water pumps, along with software tools such as Visual Studio Code, Eclipse IDE, and Firebase. Furthermore, the chapter details software and hardware prerequisites essential for successful system implementation, ensuring efficient development and alignment with project objectives.

CHAPTER 3: ANALYSIS

3.1 Introduction

This section will cover the analysis of the system, it will explore the proposed FloraHub system, detailing its architecture, workflow, and processes.

3.2 Problem Analysis

3.2.1 Use Case Diagram

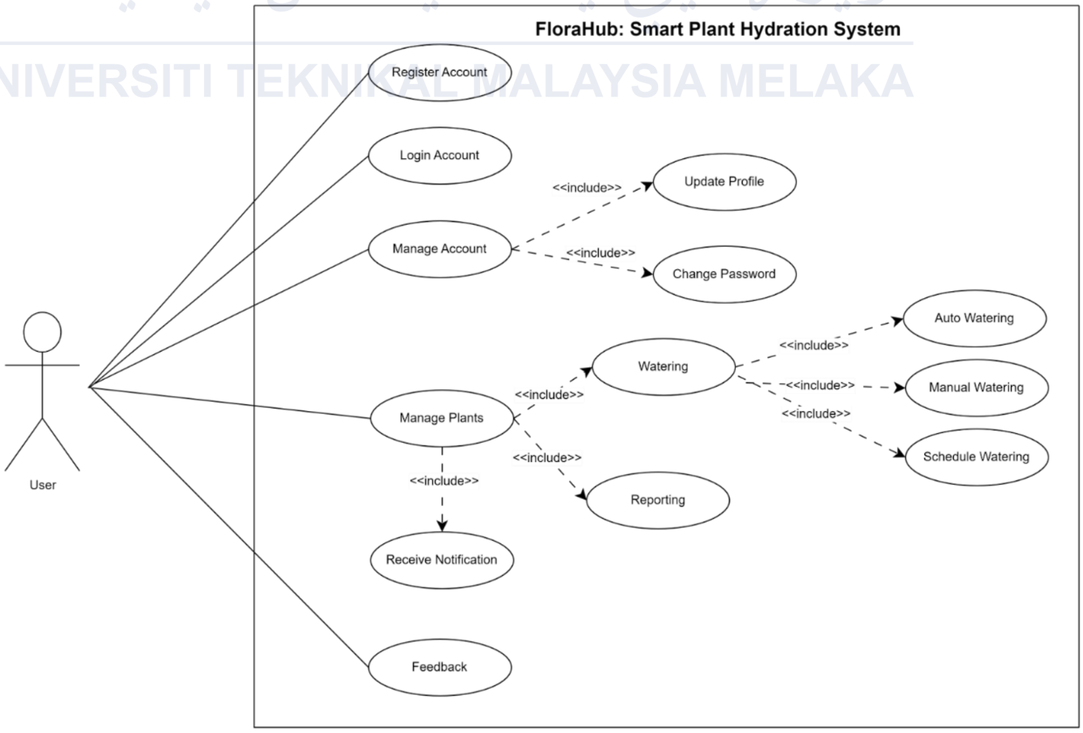


Figure 3.1: Use Case Diagram

Figure 3.1 illustrate the use case diagram for FloraHub outlines the primary functionalities and interactions between the user and the smart plant hydration system. Users can create and manage their accounts, add and monitor their plants, and utilize various watering methods, including auto-watering, manual watering, and scheduled watering. The system also provides notifications for plant watering updates and allows users to provide feedback on their experiences. Additionally, the system generates reports on plant watering patterns. This comprehensive overview of the system's features highlights its user-friendly interface and its ability to effectively assist users in caring for their plants.

3.2.2 Activity Diagram

- Sign Up Account

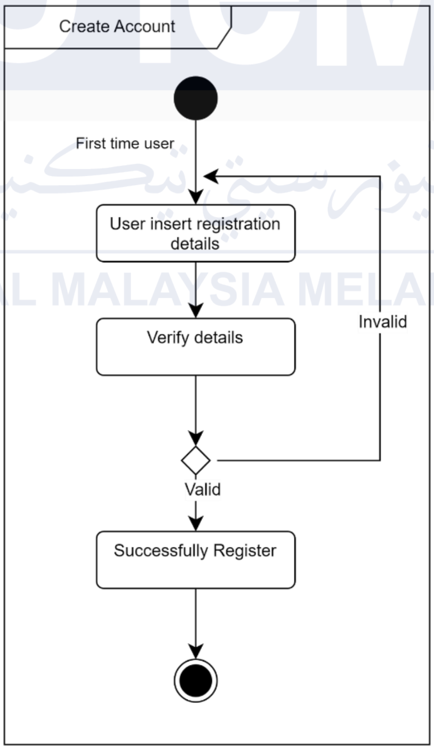


Figure 3.2: Sign Up Account

Figure 3.2 illustrates the sign-up process for a new user. The flowchart starts with a first-time user initiating the account creation process by inserting their registration details. The details are then verified. If the details are invalid, the user is prompted to re-enter their information. Once the details are validated, the user is successfully registered, completing the sign-up process.

- Sign In Account

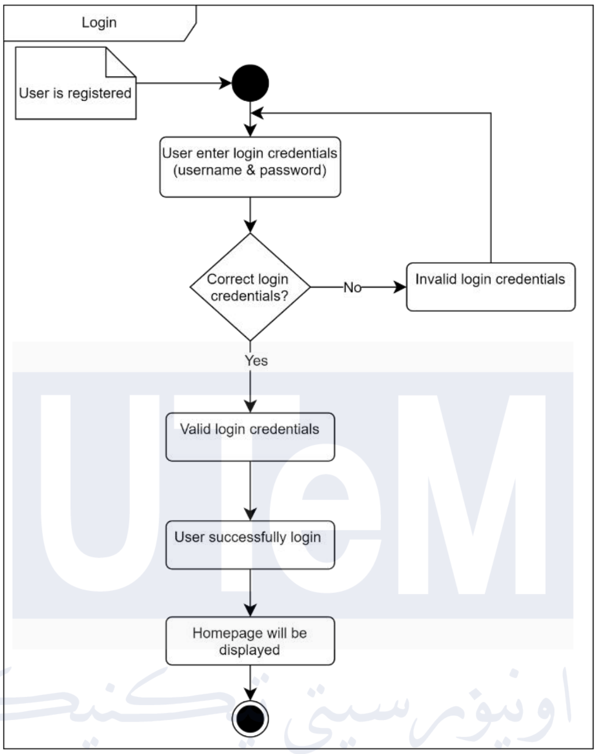


Figure 3.3: Sign In Account

Figure 3.3 illustrates the sign-in process for users that already been registered. The user will enter the login credentials (email and password), if the information is correct the user will be redirected to the homepage, otherwise, the user has to re-enter the login credentials.

- Forgot Password

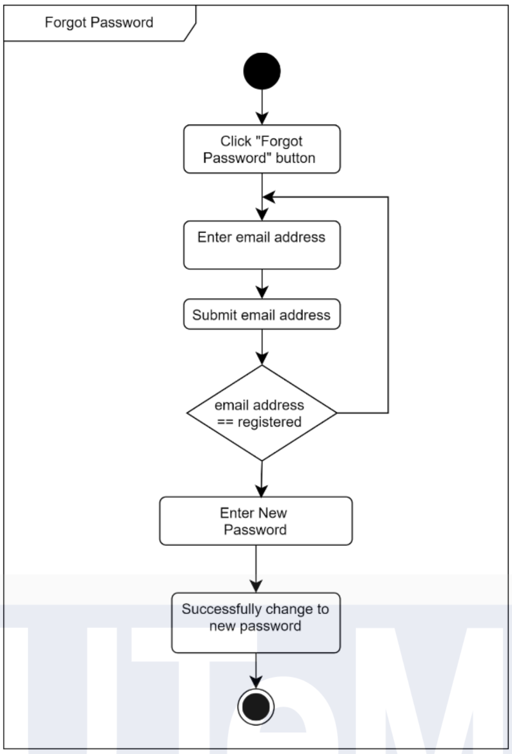


Figure 3.4: Forgot Password

Figure 3.4 outlines the process for users to recover their forgotten passwords. To initiate the password recovery process, users should click the "Forgot Password" button. This action will prompt them to enter their registered email address. Upon submitting the email address, the system will verify its validity. If the email address is registered, the user will be guided through the process of creating a new password. Once the new password is successfully entered, the system will confirm the password change.

- Change Password

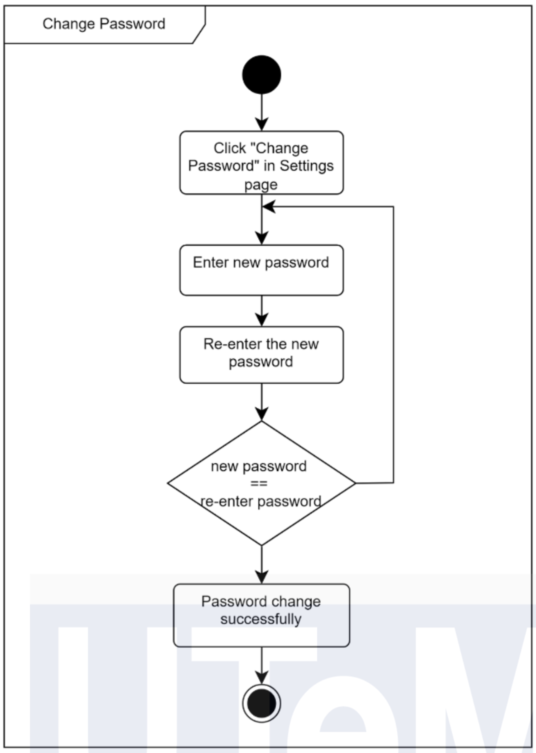


Figure 3.5: Change Password

Figure 3.5 illustrates the change of password process for users. The user will enter their new desired password and be required to re-enter the same password, if both are the same, will pop out a notification indicating that the password has successfully changed.

- User Profile

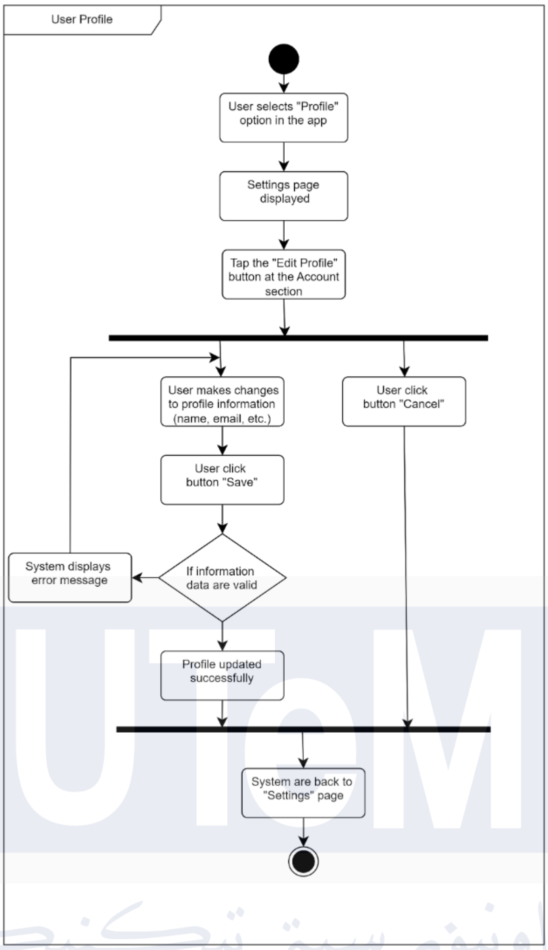


Figure 3.6: Update User Profile

Figure 3.6 illustrates the update user profile. User can make changes to their profile information such as username, email, and profile picture. Once the user clicks the save button, the updated information will stored and displayed an in-app message indicating the profile has been updated successfully.

- Watering Plants

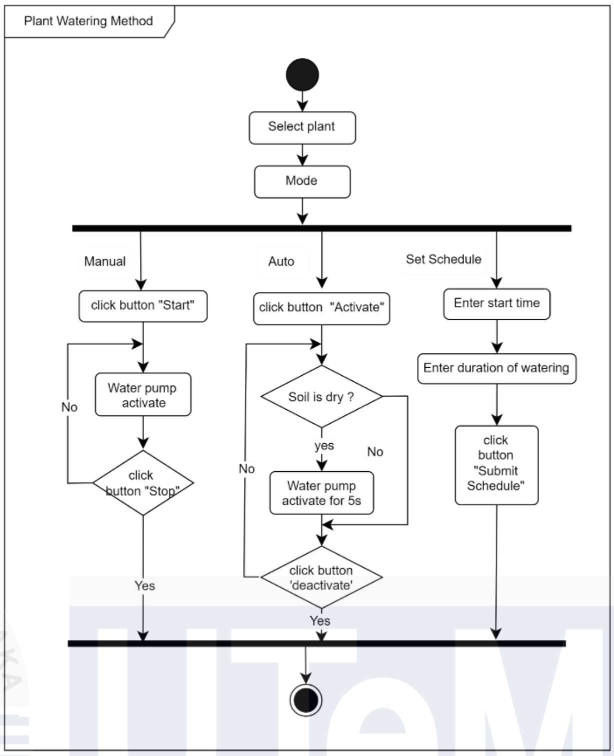


Figure 3.7: Plant Watering Method

This Figure 3.7 illustrates the plant watering methods which are manual, auto, and set the schedule. The user can choose their desired watering method by simply tap an activate button.

- Feedback

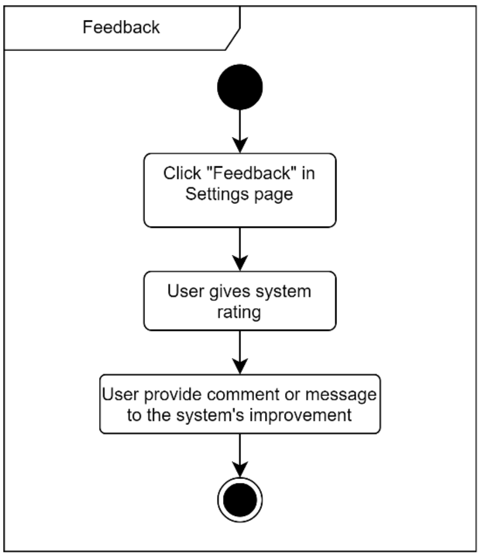


Figure 3.8: Feedback

Based on the Figure 3.8 outlines the process of user feedback on the system. To provide feedback, users can navigate to the Settings page and click the "Feedback" option. Once in the feedback section, users are prompted to rate the system's overall performance. They also have the opportunity to provide additional comments or suggestions for improvement, which can be valuable for identifying areas for enhancement or addressing specific issues. This feedback mechanism allows users to actively contribute to the system's development and ensures that their experiences and preferences are taken into account.

- Reporting

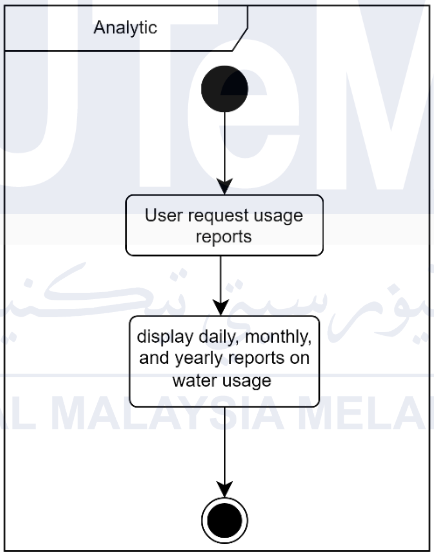


Figure 3.9: Analytic Report

Figure 3.9 illustrates the analytic report. The analytics provide daily, monthly, and yearly reports on water usage and the overall cost.

3.2.3 Internet-of-Things Design

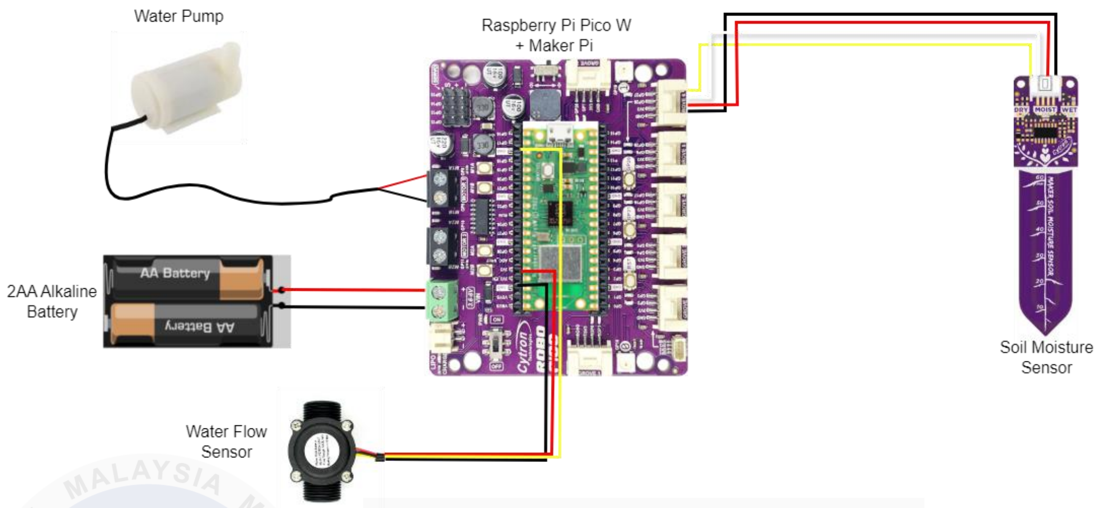


Figure 3.10: The design of the Internet-of-Things for FloraHub

Figure 3.10 shows the components of the FloraHub Smart Hydration System, including a water pump, AA batteries, a water flow sensor, a Raspberry Pi Pico W with a Maker Pi, and a soil moisture sensor, all connected to automate plant watering based on soil moisture levels.

3.3 Requirement Analysis

3.3.1 Data Requirement

Table 3.1: User Data

USER					
ATTRIBUTE NAME	DATA TYPE AND SIZE	REQUIRED	UNIQUE	PK OR FK	FK REFERENCED TABLE
Id	Int(100)	YES	YES	PK	
Username	Varchar(100)	YES			
Email	Varchar(50)	YES			
Password	Varchar(255)	YES			
State	Varchar(100	YES			

Table 3.2: User plant data

USER_PLANT					
ATTRIBUTE NAME	**DATA TYPE AND SIZE**	**REQUIRED**	**UNIQUE**	**PK OR FK**	**FK REFERENCED TABLE**
Id	Int(11)	YES	YES	PK	
Name	Varchar(100)	YES			
Description	Varchar(1000)	YES			
Type	Varchar(100)	YES			
Schedule Time	Time	YES			
UserId	Int(255)	YES		FK	User

Table 3.3: Data for Soil Moisture Sensor that are associated with plant

DATA_PLANT					
ATTRIBUTE NAME	**DATA TYPE AND SIZE**	**REQUIRED**	**UNIQUE**	**PK OR FK**	**FK REFERENCED TABLE**
Id	Int(255)	YES	YES	PK	
Date	DATE	YES			
Reading	Int(255)	YES			
Percent	Int(255)	YES			
PlantId	Int(11)	YES		FK	Plant

Table 3.4: Data for Water Flow Sensor

DATA_WATER					
ATTRIBUTE NAME	**DATA TYPE AND SIZE**	**REQUIRED**	**UNIQUE**	**PK OR FK**	**FK REFERENCED TABLE**
Id	Int(255)	YES	YES	PK	
Date	DATE	YES			
Flow_rate	Double	NULL			
Volume	Double	NULL			

Table 3.5: Data for scheduled watering based on a specific plant

SCHEDULE_WATERING					
ATTRIBUTE NAME	**DATA TYPE AND SIZE**	**REQUIRED**	**UNIQUE**	**PK OR FK**	**FK REFERENCED TABLE**
Id	Int(11)	YES	YES	PK	
Duration	Varchar(100)	YES			
StartTime	Varchar(1000)	YES			
PlantId	Int(11)	YES		FK	Plant

Table 3.6: Data for user image

USER_IMAGE					
ATTRIBUTE NAME	**DATA TYPE AND SIZE**	**REQUIRED**	**UNIQUE**	**PK OR FK**	**FK REFERENCED TABLE**
img_no	Int(11)	YES	YES	PK	
img_id	Varchar(100)	YES			
img_name	Varchar(1000)	YES			
img_type	Varchar(100)	YES			
img_size	Int(14)	YES			
img_data	longblob	YES			
userId	Int(11)	YES		FK	User

Table 3.7: Data for plant image

PLANT_IMAGE					
ATTRIBUTE NAME	**DATA TYPE AND SIZE**	**REQUIRED**	**UNIQUE**	**PK OR FK**	**FK REFERENCED TABLE**
img_no	Int(255)	YES	YES	PK	
img_id	Varchar(100)	YES			
img_name	Varchar(1000)	YES			

img_type	Varchar(100)	YES			
img_size	Int(14)	YES			
img_data	longblob	YES			
plantId	Int(255)	YES		FK	User

Table 3.8: Data for user send feedback

FEEDBACK					
ATTRIBUTE NAME	**DATA TYPE AND SIZE**	**REQUIRED**	**UNIQUE**	**PK OR FK**	**FK REFERENCED TABLE**
Id	Int(11)	YES	YES	PK	
DateTime	DATE TIME	YES			
Rate	Varchar(100)	YES			
Message	Varchar(1000)	YES			
UserId	Int(255)	YES		FK	User

3.3.2 Functional Requirement

The functional requirements of the FloraHub system are as below:

Table 3.9: Functional Requirement of FloraHub

No	Functional Requirements	Description
FR1	User Registration and Secure Login	Users should be able to register an account and log in securely using authentication mechanisms like encrypted passwords.
FR2	User Profile Management	Users should have functionalities to update personal information such as username details, and update profile image.
FR3	Alerts and Notifications	The system should send alerts to users regarding their plants' status and relevant activities, including watering activated, watering deactivated, and schedule watering.
FR4	Soil Moisture Sensor Integration	Integration of soil moisture sensors to continuously monitor moisture levels in real-time, ensuring accurate and up-to-date soil moisture data.

FR5	Threshold-Based Watering	Implementation of threshold-based watering, where the system automatically triggers watering when moisture levels fall below a user-defined threshold, optimizing water usage.
FR6	Manual Watering Control	Provision of manual watering control through the mobile app, allowing users to initiate watering manually for convenience and flexibility.
FR7	Water Usage and Cost Calculation	Calculation of water volume used and cost associated with watering based on sensor readings, providing users with detailed insights into their water consumption.
FR8	Grafana Integration	Integration with Grafana for comprehensive data analysis, enabling users to visualize and interpret soil moisture and watering data effectively.
FR9	Interactive Dashboards	Provision of interactive dashboards to explore watering patterns, and water usage, facilitating easy data exploration.
FR10	Water Usage Reports	Generation of daily, monthly, and yearly reports on water usage with detailed volume and cost calculations, helping users track and manage their water usage over time.

3.3.3 Non-functional Requirement

The non-functional requirements of the FloraHub system include:

Table 3.10: Non-Functional Requirement of FloraHub

No	Non-functional Requirements	Description
NFR1	Robust Authentication	Authentication processes should be robust to prevent unauthorized access, employing secure login mechanisms such as encrypted passwords.
NFR2	Scalability and Responsiveness	The system should be responsive and scalable to accommodate increasing user demands, ensuring smooth performance even as the number of users and sensors grows.
NFR3	Reliability	The system should be reliable, ensuring accurate soil moisture measurements and timely watering, with minimal downtime and high availability.

NFR4	Consistent Notification Delivery	Notification alerts should be delivered consistently and promptly, ensuring users receive timely updates and alerts regarding their plants' status.
NFR5	User-friendly Interface	The user interface should be intuitive and user-friendly, facilitating ease of navigation and enabling users to interact with the system efficiently without extensive training.

3.3.4 Others Requirement

- Draw.io

Figure 3.11: Draw.io Tools

Figure 3.11 illustrate Draw.io which is a free online tool for making diagrams like flowcharts and network diagrams. The tool can be used in a web browser, or downloaded to a computer and run without installing. It's easy to use and integrates with cloud storage services like Google Drive, OneDrive, and Dropbox, so we can access our diagrams from anywhere. The tool is popular because it's versatile and user-friendly.

3.4 Conclusion

In conclusion, this chapter outlines the FloraHub system's analysis, focusing on its design and requirements. It identifies key challenges in traditional plant care methods and proposes innovative solutions to address them. By emphasizing real-time monitoring, user-friendly controls, and comprehensive reporting, FloraHub aims to streamline plant management practices. Moving forward, the focus will be on translating these insights into actionable development steps, ensuring system reliability, and enhancing user experience. Ultimately, FloraHub seeks to revolutionize plant care, making it more efficient and sustainable for users.

In the next chapter, we will turn our analysis and requirements into concrete development steps. This includes outlining the system's architecture and workflow, as well as detailing its functionalities and user interface. Our goal is to ensure reliability, enhance user experience, and set the stage for FloraHub's impact on plant care.

CHAPTER 4: DESIGN

4.1 Introduction

This chapter presents a comprehensive exploration of the design phase for the FloraHub: Smart Plant Hydration System, transitioning from the analysis of the existing situation and proposed solutions to specific design elements guiding development and implementation. The design phase includes both high-level architecture and detailed system logic, ensuring alignment with project goals and user needs. By systematically detailing this process, the chapter lays a solid foundation for development, aiming for a seamless transition to system implementation.

4.2 High–Level Design

The high-level design of the FloraHub system outlines the overall structure and organization of the system components to ensure modularity, scalability, and effectiveness in addressing user needs. The design is built around a layered architecture that organizes the system into distinct layers, each with a specific responsibility. This approach helps in managing complexity, improving system maintenance, and facilitating future upgrades. At this stage, key system elements such as the user interface, data processing mechanisms, network communication, and sensing devices are designed to work together seamlessly. This high-level blueprint guides the detailed design and implementation phases, ensuring that all components integrate smoothly and function cohesively to meet the system's goals.

4.2.1 System Architecture

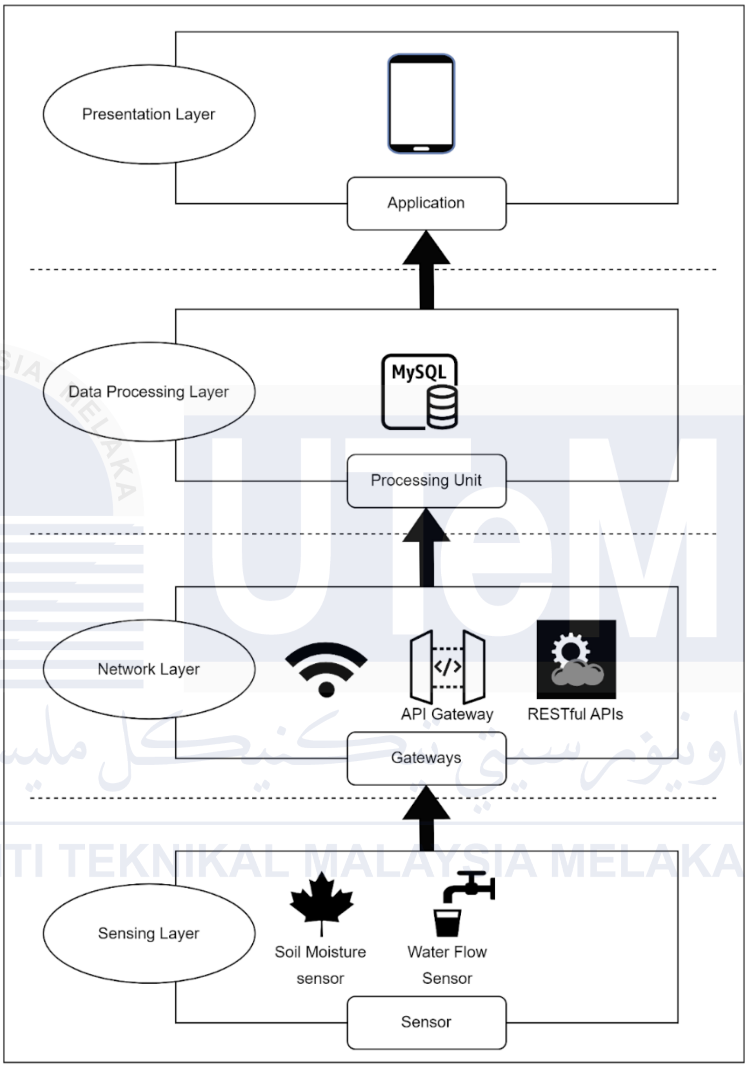


Figure 4.1: System Architecture of FloraHub

Based on the figure above, I've employed a layered architecture approach to ensure modularity, scalability, and maintainability. This architecture comprises four layers:

1. **Presentation Layer:** This layer encompasses user interfaces such as mobile applications, analytics modules, and authentication systems. It focuses on presenting information to users and capturing user inputs.
2. **Data Processing Layer:** Here, I leverage MySQL and Firebase databases to handle data storage, retrieval, and manipulation. These databases serve as the

backend storage mechanism for user information, sensor readings, and system data.

3. **Network Layer:** In this layer, I utilize Wi-Fi connectivity along with API gateways and RESTful APIs to enable communication between different components of the system. This layer ensures seamless interaction between the presentation and data processing layers over the network.
4. **Sensing Layer:** This layer comprises physical sensors such as soil moisture sensors and water flow sensors. These sensors collect real-world data, such as soil moisture levels and water usage, which is then processed and utilized by the upper layers of the architecture.

By adopting a layered architecture approach, I've been able to design systems that are modular, scalable, and robust, facilitating efficient development and enhancing overall system performance.

4.2.2 User Interface Design

The user interface (UI) design of the FloraHub system prioritizes simplicity and usability, providing a clear and intuitive experience for users. Built with Flutter, the mobile application features responsive layouts, ensuring compatibility across different devices. The UI offers easy navigation, with visual indicators for plant conditions, real-time data visualization for moisture levels and water usage, and user-friendly controls for managing notifications and settings. By focusing on user-centered design principles, the interface supports seamless interaction and enhances overall system usability.

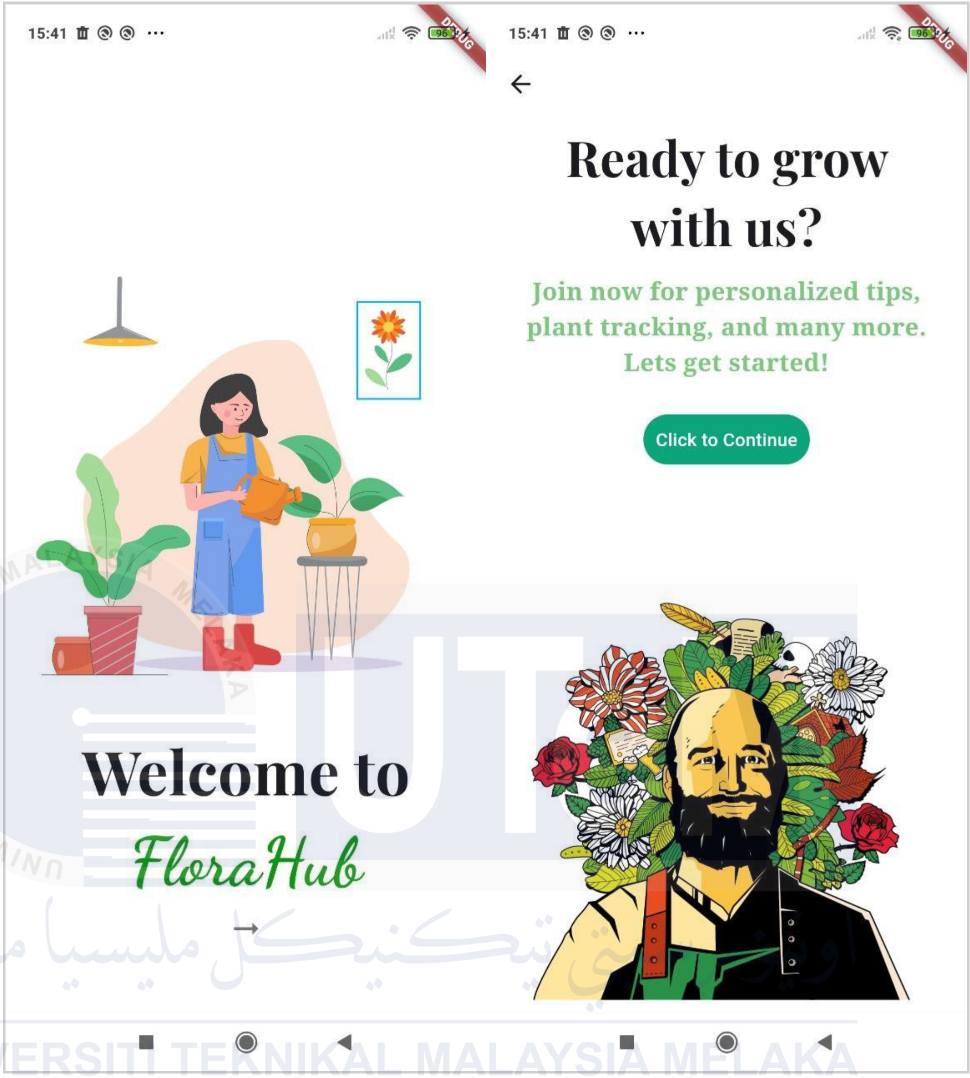


Figure 4.2: Welcome Page of FloraHub

This Figure 4.2 shows the welcome page when the user is first time installing the application.

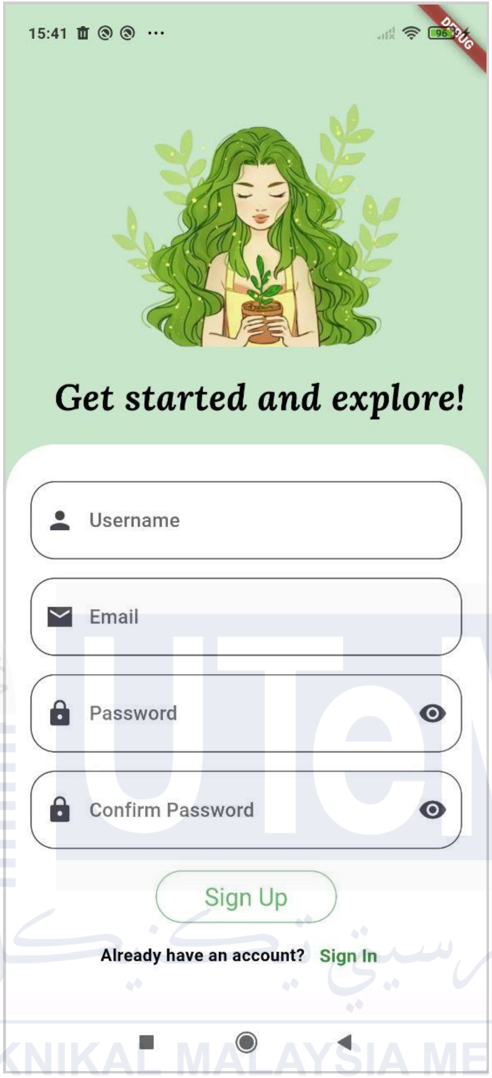


Figure 4.3: Register Page

Figure 4.3 illustrates the user interface for creating a new account. It includes fields for username, email, password, and confirm password. The form is designed to be user-friendly, helping users register quickly and easily.

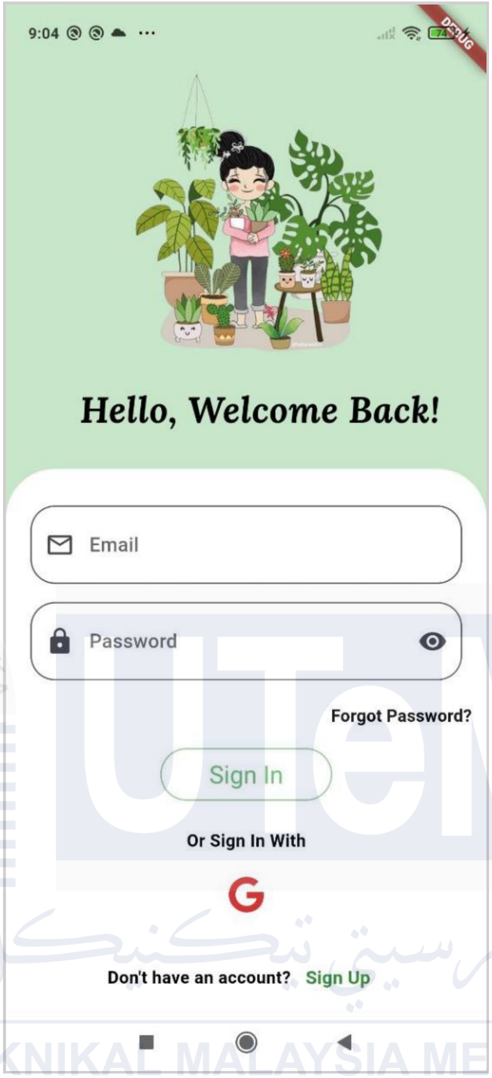


Figure 4.4: Login Page

Figure 4.4 illustrates the user interface for logging into the application. It includes fields for entering the email and password, allowing users to access their accounts.

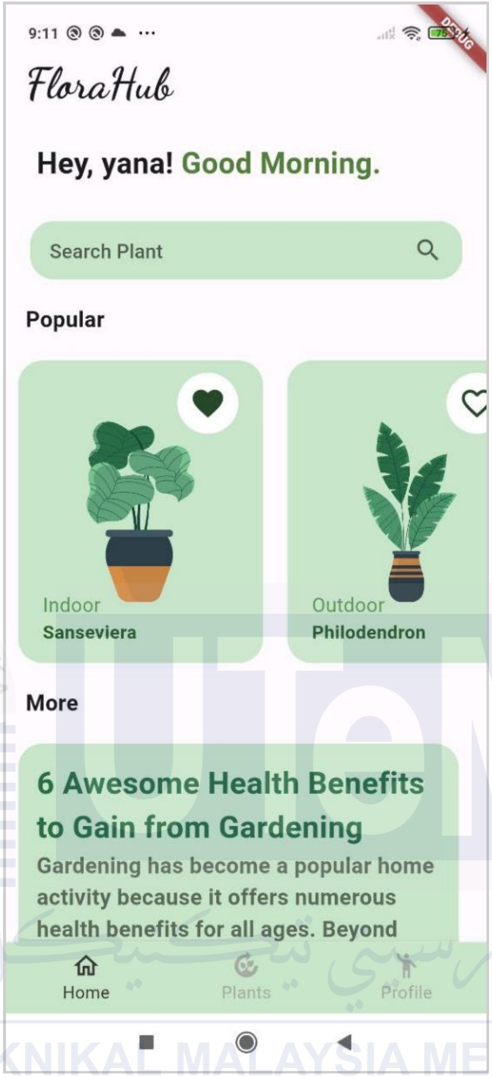


Figure 4.5: Home Page

Figure 4.5 shows the main landing page of the application after a user logs in. It provides an overview of search plants, popular plants, more information about plants, and bottom navigation options, offering quick access to various sections such as the user's profile, and plants.

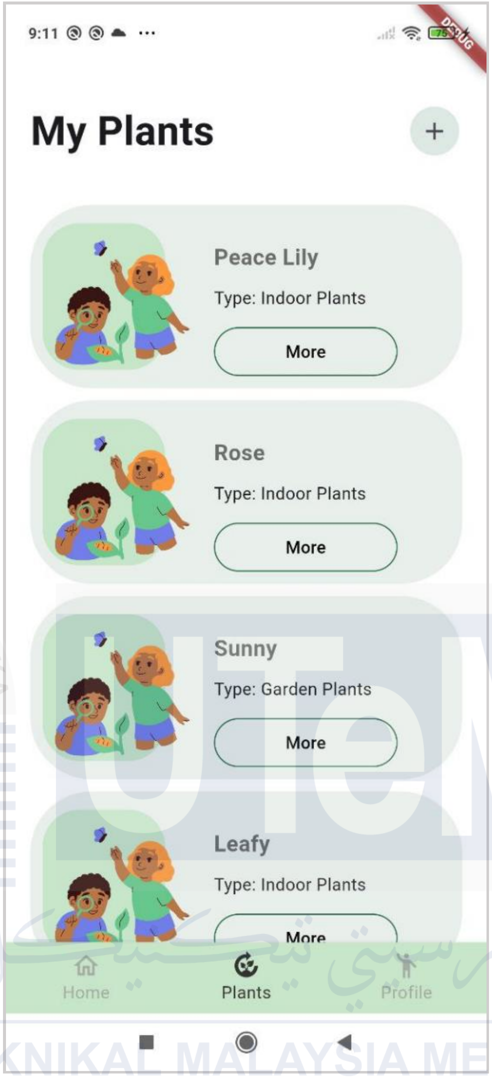


Figure 4.6: Plants Page

Figure 4.6 illustrates the page where users can view a list of their plants. It displays each plant with basic information and possibly images. The layout is organized to allow users to easily browse through their plants and their detailed information.

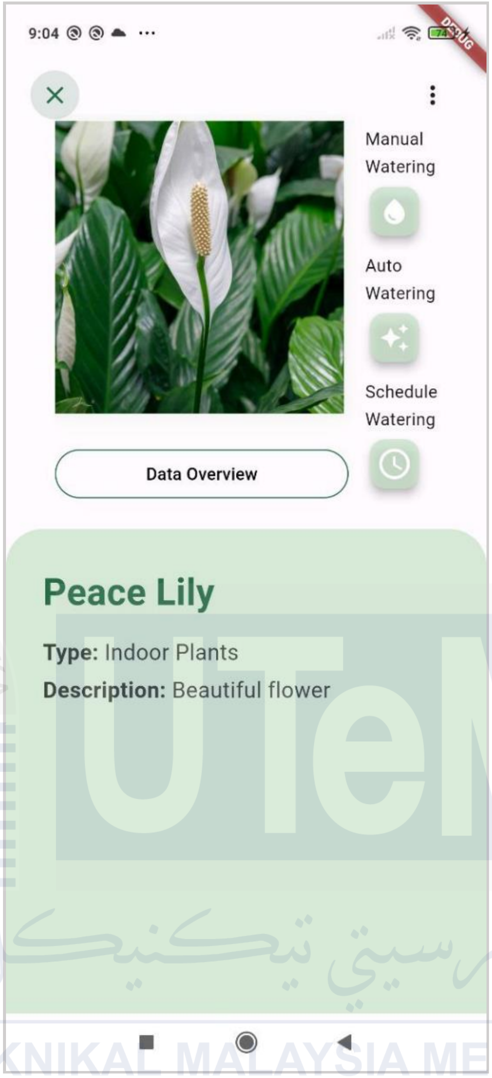


Figure 4.7: Plant Detail Page

Figure 4.7 depicts the detailed view of a specific plant. It includes comprehensive information about the selected plant, such as its watering feature, plant name, plant type, plant description, and data overview. The page is designed to provide all necessary details in a clear and accessible manner.

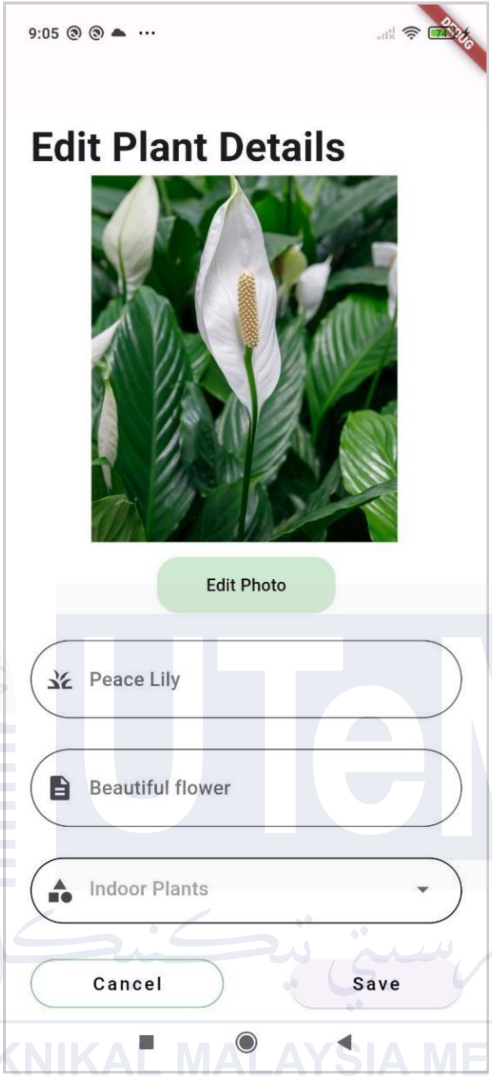


Figure 4.8: Edit Plant Details Page

Figure 4.8 shows the interface for editing the details of a specific plant. Users can update information such as the plant's name, description, and category. The page is designed to be user-friendly, making it easy to modify plant details and save changes.

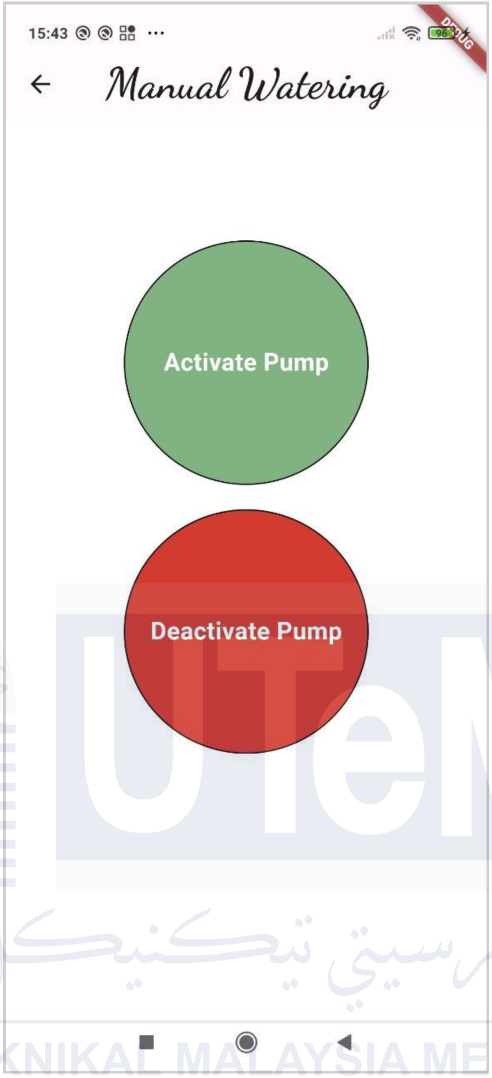


Figure 4.9: Plant's Manual Watering Page

Figure 4.9 illustrates the page for manually watering a plant. Users can simply tap the button to activate or deactivate the water pump to water the plants. The interface ensures a straightforward process for logging manual watering activities.

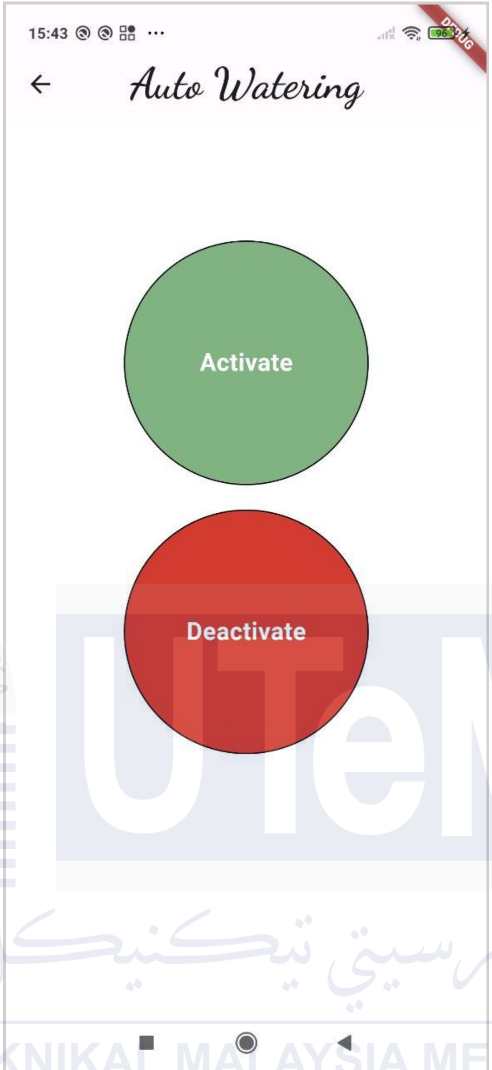


Figure 4.10: Plant's Auto Watering Page

Figure 4.10 illustrates the page for auto-watering plants that use the real-time soil moisture sensor to detect the moisture level. Users can simply tap the button to activate or deactivate the auto-watering plants.

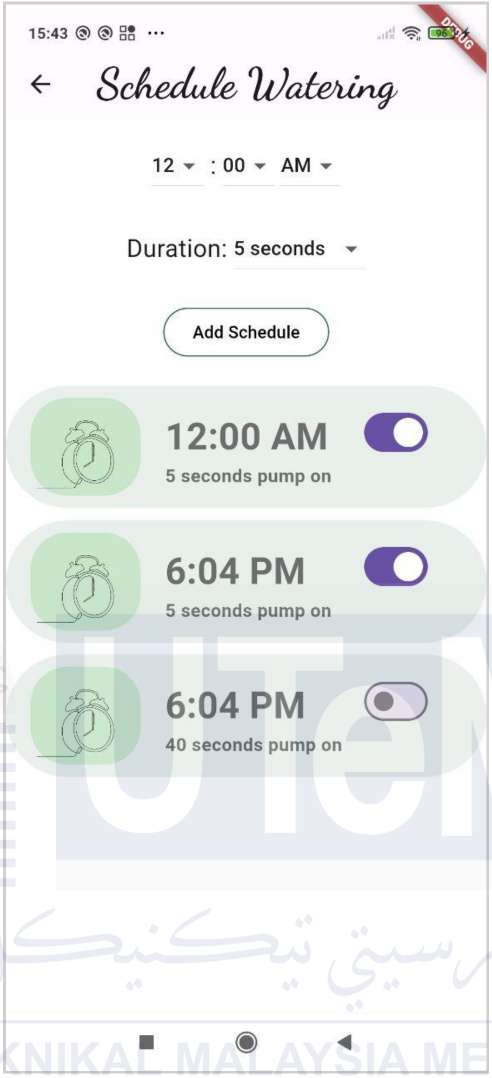


Figure 4.11: Plant's Schedule Watering Page

Figure 4.11 shows the interface for scheduling watering times for a plant. Users can set the time and the duration of the water pump to water the plants. The layout is organized to help users plan and maintain a regular watering routine.

Figure 4.12: Daily Water Volume Report

Figure 4.12 presents the daily water volume report, providing users with data on the amount of water used for their plants each day. The report is displayed in a clear format which is in cubic metres, making it easy to track daily watering habits.

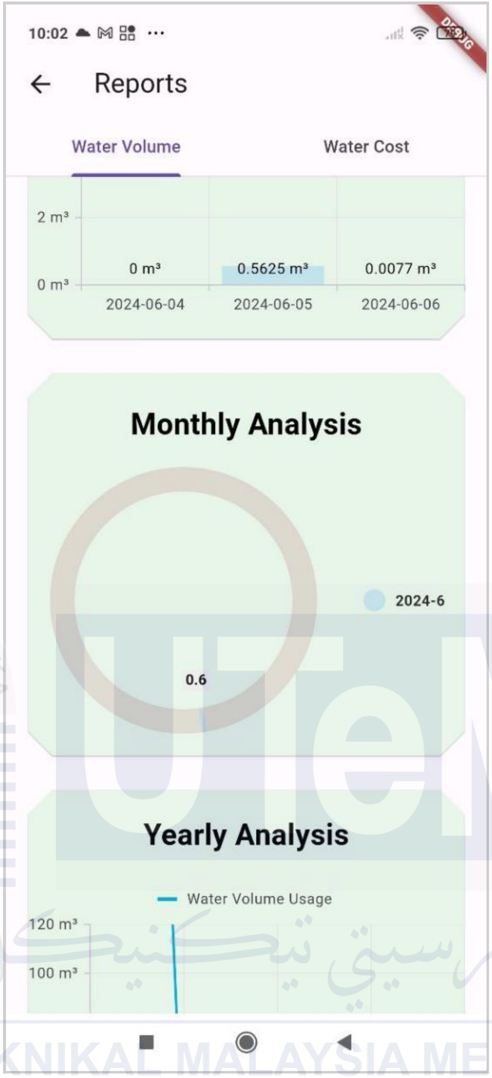


Figure 4.13: Monthly Water Volume Report

Figure 4.13 displays the monthly water volume report, summarizing the total water usage for plants over the course of a month.

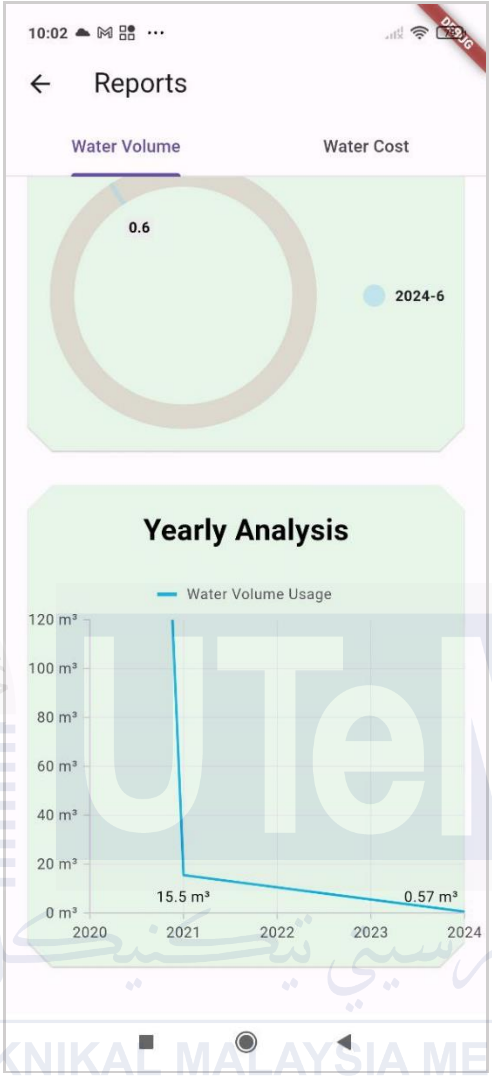


Figure 4.14: Yearly Water Volume Report

Figure 4.14 presents the yearly water volume report, providing an overview of the total water usage for plants throughout the year starting 2020 until 2024.

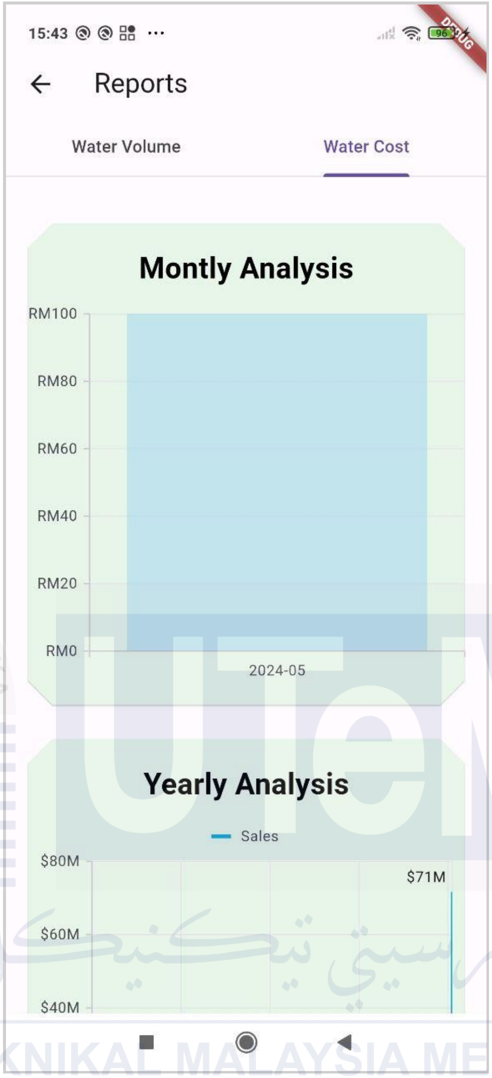


Figure 4.15: Monthly Water Cost Report

Figure 4.15 displays the monthly water cost report, detailing the expenses associated with watering plants each month. It offers users insights into their water usage costs, aiding in budget management and cost-saving strategies.

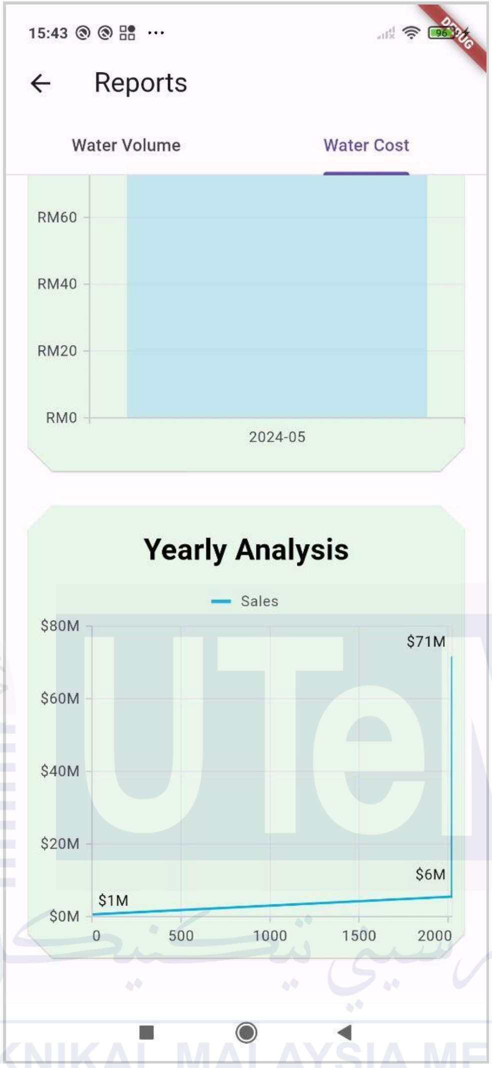


Figure 4.16: Yearly Water Cost Report

Figure 4.16 illustrates the yearly water cost report, summarizing the total expenses incurred for watering plants over the year.

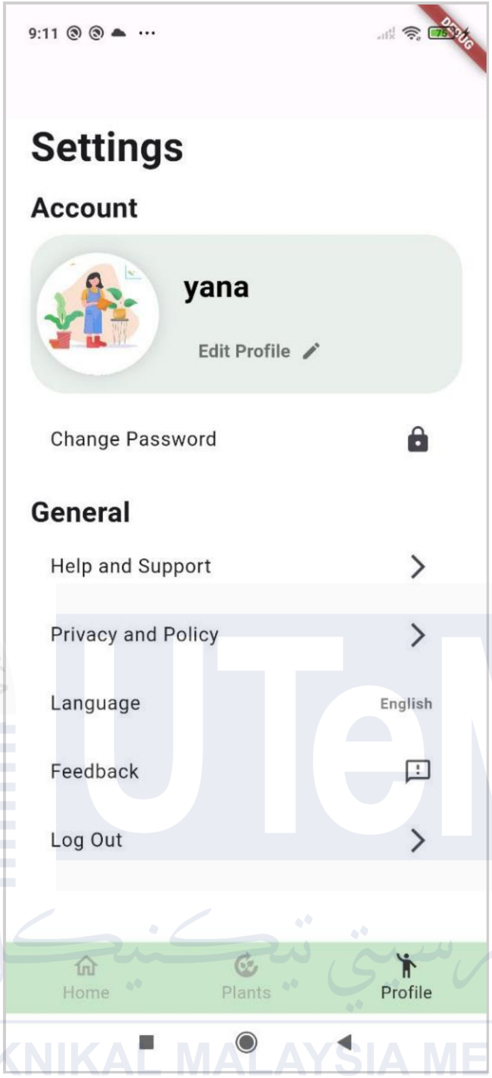


Figure 4.17: Settings Page

Figure 4.17 shows the settings page, where users can adjust various preferences and configurations for the application. Options may include account profile, change password, and account general features.

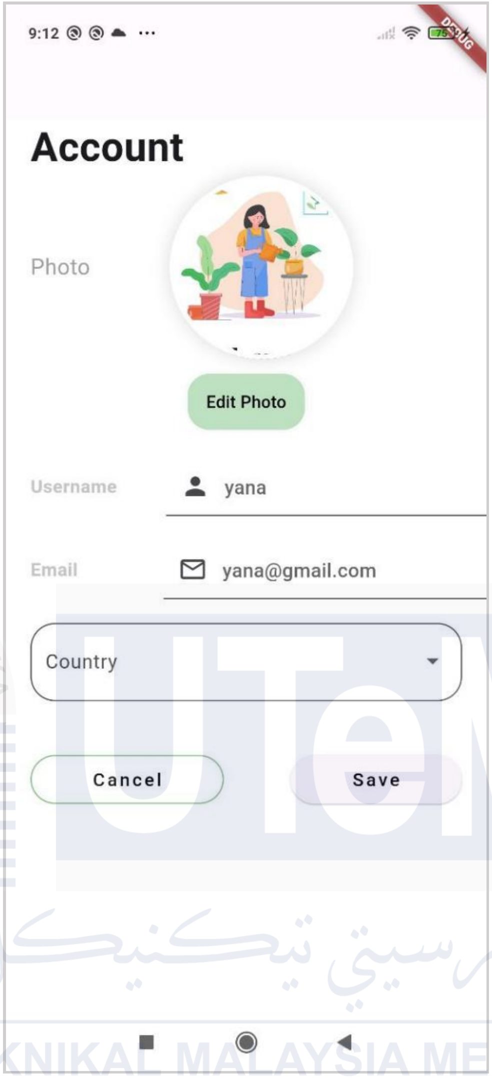


Figure 4.18: Edit Profile Page

Figure 4.18 presents the edit profile page, enabling users to update their personal information such as username, email, country and profile picture. The interface is designed to be straightforward, ensuring easy and efficient profile management.

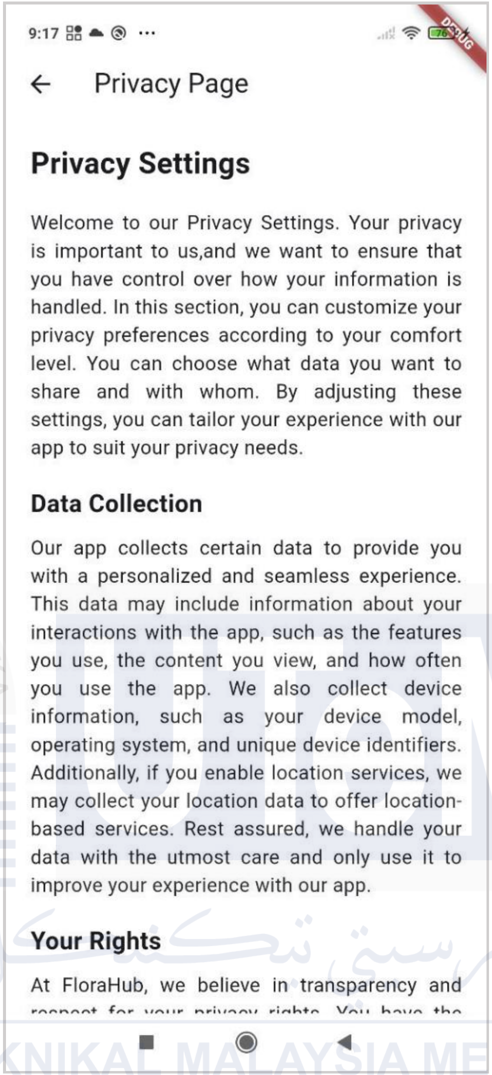


Figure 4.19: Privacy Page

Figure 4.19 illustrates the privacy page, where users can review their privacy settings.

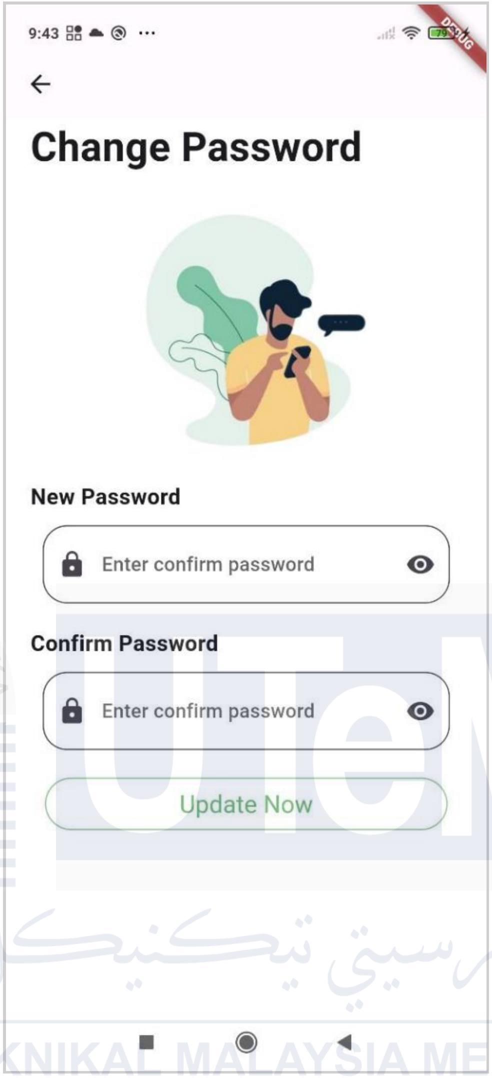


Figure 4.20: Change Password Page

Figure 4.20 shows the change password page, allowing users to update their account password. The page includes fields for entering the new password and confirming the password, with clear instructions to ensure a secure password change process.

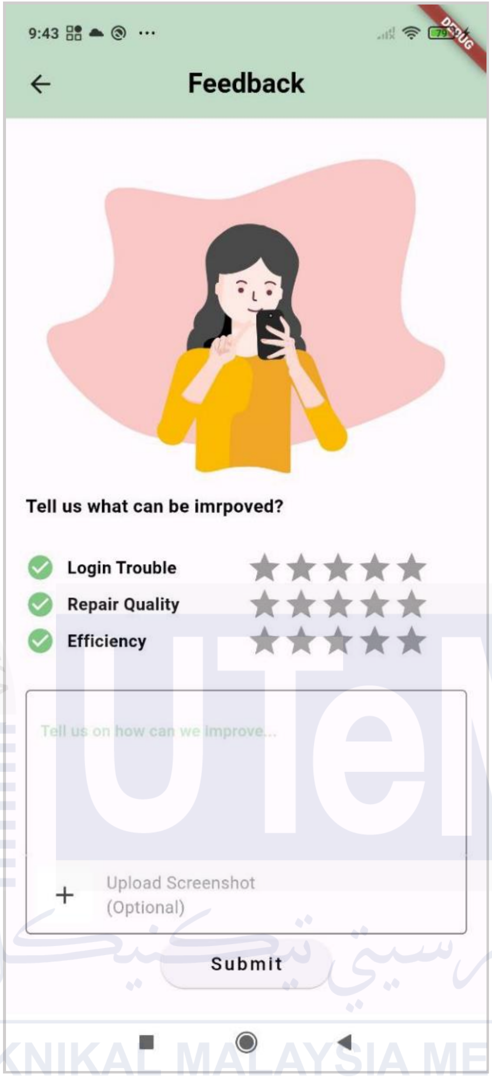


Figure 4.21: Feedback Page

Figure 4.21 displays the feedback page, where users can provide feedback about the application. It includes fields for rating, comments, and suggestions, helping developers gather user input to improve the app's functionality and user experience.

4.2.3 Database Design

4.2.3.1 Conceptual and Logical Database Design

The conceptual and logical database design phases are essential for establishing the structure and relationships of the data that will be stored in the FloraHub system's database. These phases aid in comprehending the data requirements and in drafting a blueprint for the physical database design.

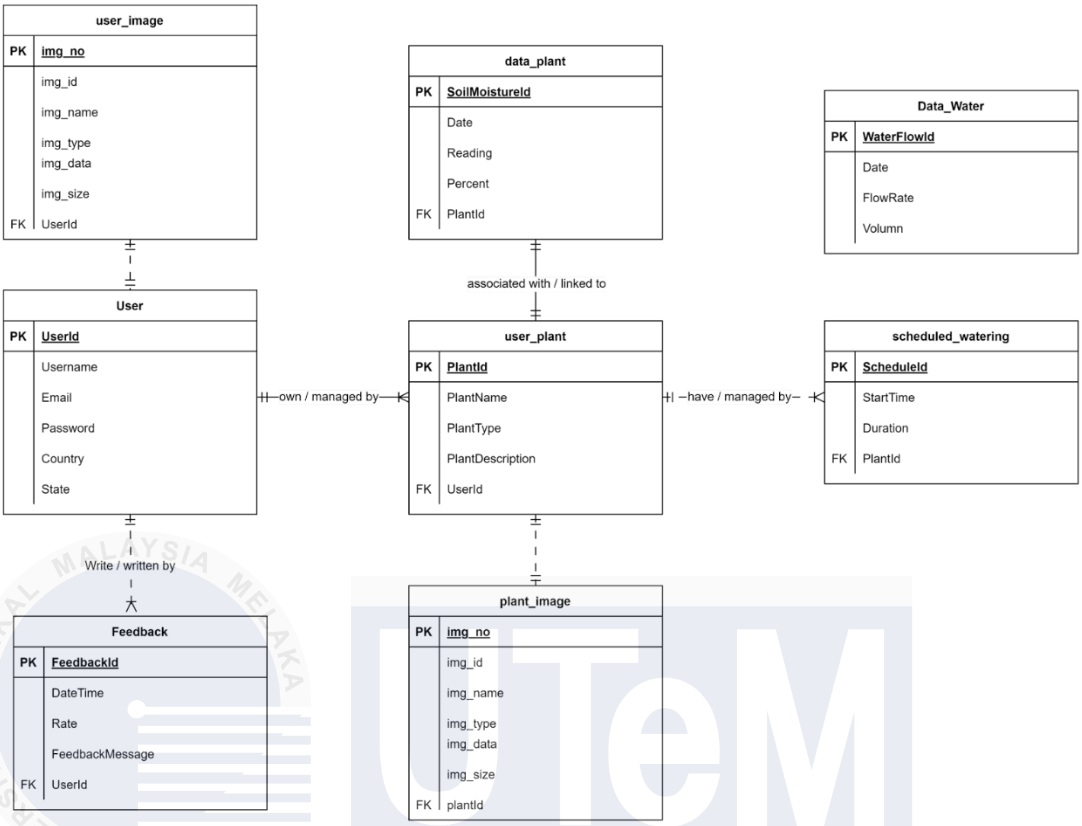


Figure 4.22: Entity Relational Diagram (ERD)

4.3 Detailed Design

The detailed design phase will cover both the software design and physical database design of the FloraHub system. During this stage, the system's software architecture and components will be further developed and elaborated to ensure the design is comprehensive enough for implementation. Additionally, the internal structure and functional behavior of each component within the FloraHub: Smart Plant Hydration System will be clearly defined.

4.3.1 Software Design

The software design process involves transforming the software requirements of FloraHub into an executable system. This approach necessitates a solid understanding of user needs to accurately identify system specifications and determine the best solution. The design strategy should be well-planned to ensure that the intended solution for the FloraHub system can be effectively implemented.

4.4 Conclusion

This chapter thoroughly examined the design strategies for the FloraHub: Smart Plant Hydration System, covering both high-level and detailed aspects. The high-level design described the system's overall structure, including architecture, user interface, and data management. This section highlighted the system's logical flow, interaction patterns, and data processing, providing a solid foundation for development.

The detailed design went deeper into the system's components, specifying program specification, program description, input and output, pseudocode and the sample screen. This level of detail ensures the system's performance, scalability, and reliability, and helps with seamless integration with existing systems.

Next, we will implement the designed system, paying close attention to the specifications outlined in this chapter. Following these design guidelines will ensure the FloraHub system meets its requirements, providing an efficient and reliable solution for smart plant hydration. Successfully implementing these design concepts will facilitate a smooth transition to development, testing, and deployment, leading to the delivery of a high-quality system.

CHAPTER 5: IMPLEMENTATION

5.1 Introduction

This chapter provides an overview of the implementation phase of the FloraHub: Smart Plant Hydration System. The implementation phase is crucial as it translates design concepts into working software. This chapter details the steps involved in setting up the software development environment, managing software configurations, and tracking version control.

The activities covered include setting up the development environment, configuring software components, and managing version control to ensure seamless integration and functionality. By the end of this phase, we expect to have a fully operational system ready for testing, with all components integrated and properly managed.

5.2 Software Development Environment Setup

1. **MySQL**

Figure 5.1: MySQL

Based on the Figure 5.1,I use MySQL, an open-source relational database management system (RDBMS), for managing and manipulating data. Unlike NoSQL databases, MySQL employs relational tables to store data. It organizes data into tables with rows and columns, making it straightforward to structure and query complex datasets. MySQL also supports data replication across multiple servers for redundancy and load balancing, which ensures high availability and reliability of the data.

2. Visual Studio Code

Figure 5.2: Visual Studio Code

Based on the Figure 5.2, I am using Visual Studio Code (VS Code), developed by Microsoft, as the primary Integrated Development Environment (IDE) and source code editor. It is available for Windows, Linux, and macOS. VS Code is ideal for developing mobile applications, as it supports multiple programming languages, allowing me to work seamlessly without needing to switch editors.

3. Mobile Application

Figure 5.3: Flutter

Based on the Figure 5.3, I used the Flutter framework for the front-end development of the mobile application and Dart as the primary programming language for designing the interface. I am using real devices to launch the application output.

4. Backend

Figure 5.4: Spring Boot

Based on the Figure 5.4, I am using the Spring Boot framework for my backend development. Spring Boot is an open-source Java framework designed for creating standalone, production-grade applications with minimal effort. It is a convention-over-configuration extension of the Spring Java platform, aimed at reducing configuration complexities while building Spring-based applications.

Spring Boot handles HTTP requests through its built-in support for the Spring MVC framework. It uses annotations like `@RestController` and `@RequestMapping` to map HTTP requests to specific handler methods, making it easy to define endpoints for various HTTP methods such as GET, POST, PUT, and DELETE. This streamlined process allows developers to create robust server-side applications quickly and efficiently.

5.3 Software Configuration Management

5.3.1 Configuration environment setup

To develop the FloraHub: Smart Plant Hydration System, several environments need to be set up to enable the necessary libraries for the programming languages. The data store will use MySQL, a relational database management system that efficiently stores information. The setup process involves installing and configuring MySQL, resulting in the availability of the database for real-time operations. FloraHub is a mobile application supported by a backend server application using Spring Boot with Java. The system is based on a client-server architecture, where the client side is the mobile application developed using Flutter with Dart programming language in the Visual Studio Code (VSCode) Integrated Development Environment (IDE). The backend, developed in Spring Boot with Java, is built in the Eclipse IDE. All logical processing, such as authorization and CRUD operations, will be handled by the backend. To ensure smooth interaction between the frontend and backend, REST (Representation State Transfer) APIs will be used. Postman will be utilized to test the connections between the front end and backend. The backend developed in Spring Boot will provide the necessary endpoints for the frontend to interact with. For the IoT component, MicroPython will be used to develop the IoT devices, also in the VSCode IDE. The connection of data between the IoT devices and the server will be tested using HTTP protocols.

5.3.2 Version Control Procedure

Figure 5.5: GitHub

Figure 5.5 illustrates GitHub Desktop which I use to track and control all changes during development. It helps me manage and update the source code efficiently, ensuring that all modifications are documented and synchronized. This tool simplifies version control and keeps the development process organized.

5.4 Implementation Status

The progress of the development status for each of the modules is shown below in the table:

Table 5.1: Progress of the Development Status

Module	Description	Duration to complete	Date completed
Authentication Module	Module to allow the user to log in to the account, create a new account, and view or edit the user profile	2 weeks	31/3/2024
Notification Module	Module to receive alerts based on the plant's information	2 weeks	11/6/2024
Soil Moisture Module	Module to monitor plant's moisture level and control the watering manually.	3 weeks	14/4/2024
Water Flow Module	Module to measure the volume of water, record the readings, and calculate water bill based on the water volume	3 weeks	5/5/2024
Analytics Module	Module to view interactive dashboards and reporting on water usage.	2 weeks	18/5/2024

5.5 Conclusion

In this chapter, we have explored the implementation phase of the FloraHub: Smart Plant Hydration System, focusing on setting up the development environment, managing software configurations, and applying version control. We detailed the processes for configuring the environment and controlling version changes to ensure efficient development and integration.

CHAPTER 6: TESTING

6.1 Introduction

The software development lifecycle's testing and evaluation stage is essential for ensuring the created system satisfies the requirements and operates as planned. Thorough testing was done on the FloraHub: Smart Plant Hydration System to confirm its dependability, performance, and functionality. This chapter provides an overview of the testing methodologies employed, the test cases designed and executed, and the results obtained. The objective is to identify and rectify any issues, ensuring the system is robust and user-friendly.

6.2 Test Plan

6.2.1 Test Organization

The testing process for the FloraHub: Smart Plant Hydration System involves three key individuals. As the sole developer, I am responsible for all aspects of testing, including writing and executing test cases, managing the test environment, and documenting results. My supervisor plays a crucial role by providing guidance, reviewing test plans and results, and ensuring the testing process meets academic standards. Additionally, the evaluator will review the final test results and verify that the system meets the required specifications and performance criteria, providing an unbiased assessment of the system's functionality and performance.

6.2.2 Test Environment

The test environment for the FloraHub system is a critical setup that ensures comprehensive testing during the system's development and validation phases. It consists of both hardware and software configurations specifically selected to support the diverse testing needs of the system, including performance, functionality, and usability. Key hardware components include development laptops, Android devices for real-world testing, and stable internet access. Essential software tools such as Visual Studio Code, Eclipse, and Postman are employed to support frontend, backend, and API testing. These components collectively ensure that the FloraHub system is rigorously tested across various platforms and scenarios, ensuring reliability and efficiency.

The specific test environment used for the FloraHub system is outlined in **Table 6.1**, which details the hardware and software configurations employed during testing.

Table 6.1: Detailed Test Environment Configuration for FloraHub

Test Environment	Requirement	Description
Hardware Configuration	Laptop	A laptop is required for development and testing, equipped with Windows 10 or above. It must have sufficient processing power, memory, and storage to run development tools. A high-resolution screen is also recommended to ensure accurate UI testing.
	Internet	A stable internet connection is essential for accessing online resources, version control, and API testing.
	Android Phone	The FloraHub mobile application is tested on real Android devices like the Redmi Note 9 Pro. Testing on actual devices helps ensure that the app works well across different screen sizes,

		hardware configurations, and Android versions. This step is crucial for verifying the user interface and experience, particularly for the app's interaction with the IoT components.
Software Configuration	Windows 10 or above	This operating system supports the tools and environments necessary for developing and testing the FloraHub system. It is used to install and manage all software required for the project, such as IDEs, API testing tools, and documentation platforms.
	Google Docs	Google Docs is used to document the testing processes, plans, and results for FloraHub.
	Visual Studio Code	Visual Studio Code is one of the main development environment for the FloraHub system which are used to develop the frontend code in Flutter, and the MicroPython code for IoT devices.
	Eclipse	Eclipse is one of the main development environment for the FloraHub system which are used for writing the backend code in Spring Boot
	Postman API Platform	Postman is a tool used to test APIs in Spring Boot applications. It helps the developer to check if the backend services are working correctly by creating and testing HTTP requests to make sure the APIs return the right responses.

6.2.3 Test Schedule

The test schedule outlines the specific testing cycles and their corresponding durations for the FloraHub system. It provides a structured timeline detailing the different phases of testing, including Unit Testing, Integration Testing, System Testing, Acceptance Testing, and Usability Testing. Each phase focuses on a particular aspect of the system, from validating individual components to ensuring the overall functionality and user experience. This systematic approach ensures that all critical components and integrations are thoroughly evaluated and meet the defined requirements within the scheduled timeframe.

Table 6.2 suggests a clear overview of this schedule, providing specific details about the task name, description, start and end dates, and the duration of each testing phase.

Table 6.2: FloraHub System Test Schedule and Duration

Task Name	Task Description	Start Date	End Date	Duration
Unit Testing	Detailed examination and validation of individual components and features within the FloraHub system, such as soil moisture sensor data retrieval, watering activation, and notification triggers.	15/07/2024	19/07/2024	5 days
Integration Testing	Testing the combined modules within the FloraHub system to ensure seamless data flow and proper functioning between components like soil moisture sensors, the MySQL database, and the Flutter app.	20/07/2024	25/07/2024	6 days
System Testing	Comprehensive evaluation of the entire FloraHub system, including performance, reliability, and compliance	26/07/2024	03/08/2024	9 days

	with specified requirements. This includes stress testing to handle multiple sensors and users.			
Acceptance Testing	Final evaluation by end-users or stakeholders to verify system readiness for deployment. This includes real-world testing during events like the FYP Innovation Competition and online sessions.	04/08/2024	10/08/2024	7 days
Usability Testing	Assessment of the FloraHub system's ease of use and user-friendliness. This involves gathering feedback on the app's interface, navigation, and overall user experience to identify and address UX issues.	11/08/2024	15/08/2024	5 days

6.3 Test Strategy

6.3.1 Classes of Tests

The testing phases for the FloraHub system are critical in ensuring that the software meets its functional and non-functional requirements, delivering a reliable and user-friendly experience. These phases include Unit Testing, Integration Testing, System Testing, and Acceptance Testing, each of which plays a vital role in the development and deployment of FloraHub.

1. **Unit Testing**

Unit testing checks individual components of the FloraHub system to ensure they work as expected. Developers test features like soil moisture data retrieval, watering activation, and notifications. For example, it ensures that when dryness is detected, the right notification is sent, and the water pump activates. This testing happens during development to verify the reliability of each part before integrating them into the system.

2. **Integration Testing**

   Integration testing checks if different parts of the FloraHub system work together smoothly. It ensures that data from soil moisture sensors flows correctly from the Raspberry Pi Pico to the MySQL database and triggers notifications in the Flutter app when needed. This phase helps catch any issues that occur when combining components, ensuring they work together as expected.

3. **System Testing**

   System testing evaluates the entire FloraHub system, ensuring all components work together seamlessly. The QA team checks performance, reliability, and compliance with requirements. This includes verifying that soil moisture is accurately monitored, watering actions are triggered correctly, and notifications are sent promptly. System testing also involves stress tests to ensure the system can handle multiple sensors and users without issues, ensuring FloraHub meets both functional and non-functional requirements.

4. **Acceptance Testing**

   Acceptance testing is the final step where the FloraHub system is tested by end-users or stakeholders to ensure it's ready for deployment. This testing happens in real-world environments, like the FYP Innovation Competition and online sessions, where users provide feedback on usability, functionality, and performance. It ensures FloraHub meets user expectations and is fully prepared for release, with adjustments made based on the feedback to guarantee user satisfaction.

5. **Usability Testing**

Usability Testing is focused on evaluating how easy and intuitive the FloraHub system is for users. This testing ensures that the app's interface is user-friendly, allowing users to navigate through features like setting up watering schedules and receiving notifications without difficulty. The goal is to identify any user experience issues and make necessary improvements, ensuring that even those with limited technical skills can use the app effectively. By addressing feedback from this testing phase, FloraHub can provide a smoother and more satisfying user experience.

6.4 Test Design

6.4.1 Test Description

This section provides a detailed explanation of the test cases designed to verify the functional requirements of the FloraHub system. Each functional requirement is mapped to specific test cases that target key modules, such as Authentication, Notification, Water Flow, and Analytics. The tests are aimed at validating the system's behavior in real-world scenarios, ensuring that the system performs as expected across various functionalities.

The description of each test case outlines the purpose, testing criteria, and expected results, as shown in **Table 6.3: FloraHub Functional Requirements and Test Cases Overview**. This comprehensive approach ensures that the system meets its functional goals and satisfies user requirements.

Table 6.3: FloraHub Functional Requirements and Test Cases Overview

Functional Requirement Id	Test Requirement Id	Module Name	Description	Expected Results
FR1	T01	Authentication Module	Validate that when the user needs to create a password, they need to use 8 or more characters with a mix of letters, numbers, and symbols.	The system successfully accepts the password if it meets the criteria, and the user is allowed to proceed.
	T02		Verify that when the password and email address that the user entered is correct, it will log in to the app.	The user is successfully logged into the app and redirected to the homepage.
	T03		Validate that if the user forgot their password, they can recover it by using their email address to change their password.	The system allowing the user to reset their password and successfully log in.
	T04		Verify that when the user creates an account, they need to enter their username, email, password and re-enter the password.	The system successfully creates the account after validating the entered information, and the user is redirected to the login page.

FR2	T05		Verify that users can update their username successfully.	Username is changed and reflected in the system.
	T06		Verify that users can upload and change their profile image.	Profile image is updated successfully.
FR3	T07	Notification Module	Verify that the system sends a notification when soil moisture is dry and the watering is activated.	The system successfully sends a notification to the user when soil moisture levels drop below the set threshold and watering is activated.
	T08		Verify that the system sends a notification when watering is deactivated.	A notification is sent to the user indicating watering has stopped.
FR5	T09	Water Flow Module	Verify that the system automatically triggers watering when soil moisture falls below the user-defined threshold.	Watering is initiated when moisture levels reach the threshold.
FR6	T10		Verify that the user can manually initiate watering through the mobile app.	Watering starts as soon as the user initiates it.
	T11		Verify that the system can stop watering manually.	Watering stops immediately when the user cancels it.

FR7	T12		Verify that scheduled watering data is saved correctly in the system.	Scheduled watering data is stored accurately.
	T13		Verify that the system executes scheduled watering tasks as planned.	Watering starts and stops according to the schedule.
FR8	T14	Analytics Module	Verify that soil moisture and watering data is displayed in Grafana dashboards.	Data is visualized appropriately in Grafana.
FR9	T15		Verify that users can interact with dashboards to explore watering patterns and water usage.	Dashboards provide interactive features for data exploration.
FR10	T16		Verify that the system generates daily, monthly, and yearly water usage reports.	Reports contain water volume in cubic metre.

6.4.2 Traceability Matrix

The traceability matrix in this section establishes a clear link between the functional requirements (FRs) and the corresponding test cases (TCs) for the FloraHub system. It ensures that all requirements are adequately tested by mapping each test case to its associated functional requirement. This matrix provides an organized structure to verify that the system's features have been validated through appropriate testing, ensuring that each requirement is met by specific test cases. It also helps in tracking testing progress and ensuring complete coverage of the system functionalities during the testing process, as outlined in **Table 6.4.**

Table 6.4: FloraHub Traceability Matrix for Functional Requirements and Test Cases

Test Requirement ID	Test Case ID	Description
T01	TC01 TC02 TC03	Validate that when the user needs to create a password, they need to use 8 or more characters with a mix of letters, numbers, and symbols.
T02	UC01	Verify that when the password and email address that the user entered is correct, it will log in to the app.
T03	UC02	Validate that if the user forgot their password, they can recover it by using their email address to change their password.
T04	UC03	Verify that when the user creates an account, they need to enter their username, email, password and re-enter the password.
T05	UC04	Verify that users can update their username successfully.
T06	UC05	Verify that users can upload and change their profile image.
T07	UC06	Verify that the system sends a notification when soil moisture is dry and the watering is activated.

T08	UC07	Verify that the system sends a notification when watering is deactivated.
T10	UC08	Verify that the user can manually initiate watering through the mobile app.
T11	UC09	Verify that the system can stop watering manually.
T12	UC10	Verify that scheduled watering data is saved correctly in the system.
T13	UC11	Verify that the system executes scheduled watering tasks as planned.
T14	UC12	Verify that soil moisture and watering data is displayed in Grafana dashboards.
T16	UC12	Verify that users can interact with dashboards to explore watering patterns and water usage.
T17	UC13	Verify that the system generates daily, monthly, and yearly water usage reports.

6.6 Test Result and Analysis

6.6.1 Test Cases

A. Black Box Technique 1: Equivalence Partitioning Testing

1. T01: Validate that when the user needs to create a password, they need to use 8 or more characters with a mix of letters, numbers, and symbols.

Table 6.5: Test Cases TC01-TC03

Test Case ID	Partition Tested	Test Data	Test Steps	Expected Results	Actual Results	Status
TC01	Alphabets only	Password: abc	1. The user selects "Sign Up" 2. The user enters their username. 3. The user enters their email. 4. The user selects "Password". 5. The user enters the password. 6. The user re-enters the password.	Invalid password	Invalid password	Pass
TC02	Alphabets, numbers, and symbols	Password: abc1234_	1. The user selects "Sign Up" 2. The user enters their username. 3. The user enters their email. 4. The user selects "Password". 5. The user enters the password. 6. The user re-enters the password.	Valid password	Valid password	Pass
TC03	Numbers only	Password: 1234	1. The user selects "Sign Up"	Invalid	Invalid	Pass

			2. The user enters their username. 3. The user enters their email. 4. The user selects "Password". 5. The user enters the password. 6. The user re-enters the password.	password	password	

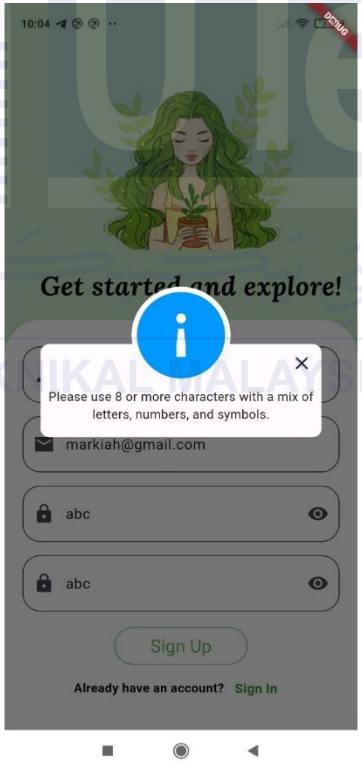


Figure 6.1: Password Validation in FloraHub

B. Black box Technique 2: Use Case Testing

2. T02: Verify that when the password and email address that the user entered are correct, it will log in to the app.

Table 6.6: Test Case UC01

Test Case ID:	UC01
Module Name	Authentication Module
Use Case Name	Login Account
Use Case Description	The process to log into the account by verifying the user password and email address.
Actor	User
Pre-Conditions	The user must already have an Account.
Test Data	Email: markiah@gmail.com Password: abc1234_
Basic Flow	1. User click "Email". 2. User enters the email. 3. User click "Password". 4. User enter password. 5. User click the "Sign in" button.
Post Conditions	The account successfully signed in.
Exception Flow	An error message will display if the user enters the wrong email or password as shown in **Figure 6.3**
Expected Result	The user is successfully logged into the app and redirected to the homepage.
Actual Result	The user is successfully logged into the app and redirected to the homepage as shown in **Figure 6.2**

Status	Success

Figure 6.2 shows the successful outcome of the login attempt, where the user is correctly redirected to the homepage after user click "Ok" button. Conversely, **Figure 6.3** illustrates the scenario where an error message is displayed if incorrect login credentials are used.

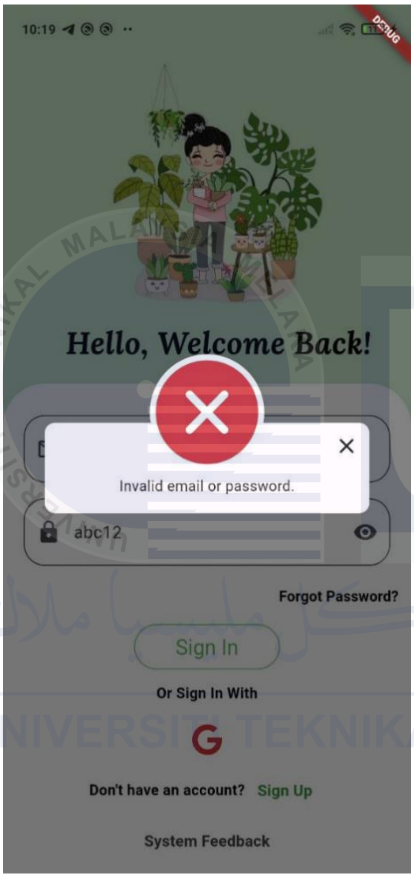


Figure 6.3: Error message for invalid email or password

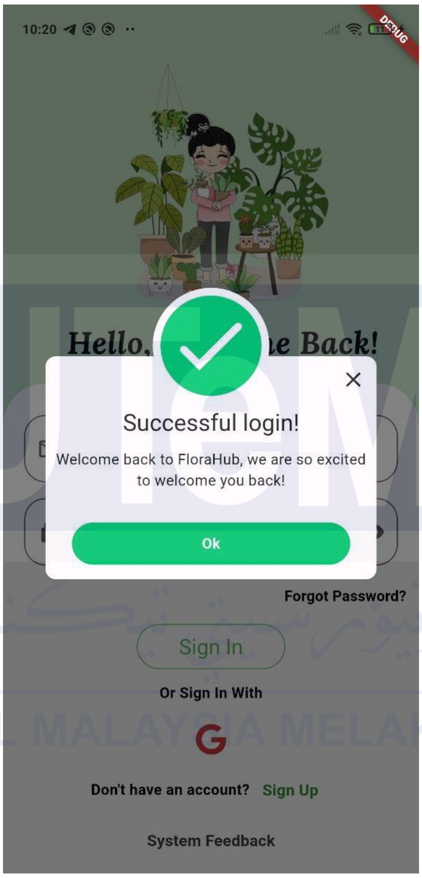


Figure 6.2: Successful message for valid email or password

3. T03:Validate that if the user forgot their password, they can recover it by using their email address to change their password.

Table 6.7: Test Case UC02

Test Case ID:	UC02
Module Name	Authentication Module

Use Case Name	Change Password
Use Case Description	Process to recover the user account by changing their password. The system will check wether the user email is exist or not. If the email is exist, the system will redirect the user to another page to change their password.
Actor	User
Pre-Conditions	User must already have an Account.
Test Data	Email: markiah@gmail.com
Basic Flow	1. User's current page are sign in page. 2. User click the "Forgot Password?" button. 3. User must enter their email address and click "Send" button.
Post Conditions	If the user's email exist, the system will display a message and will redirect to another page to let user to set their new password.
Exception Flows	Error message will display if user entered wrong email.
Expected Result	The system allowing the user to reset their password and successfully log in.
Actual Result	The system allowing the user to reset their password and successfully log in.
Status	Success

Figure 6.4 displays the "Forgot Password" page where the user starts the recovery process by entering their email address. Upon submission, if the email exists in the system, the user is redirected to **Figure 6.7**, where they can set a new password. This successful flow is depicted in **Figure 6.7**, showing the page where users can enter their new password. Conversely, **Figure 6.6** demonstrates the error message displayed if the user enters an invalid email address. This figure shows that the system will alert the user to the issue and prompt them to re-enter their email.

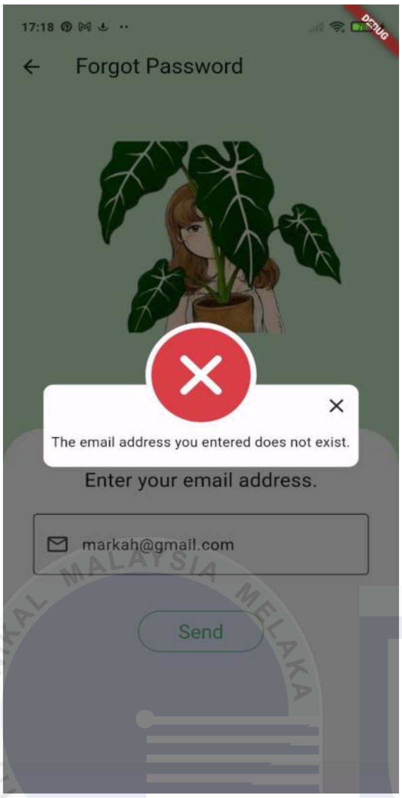


Figure 6.6: Error Message Displayed for Invalid Email

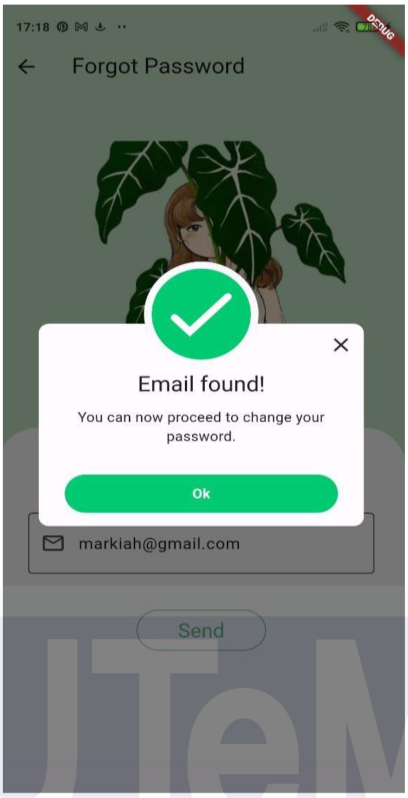


Figure 6.5: Success Message Displayed for Valid Email

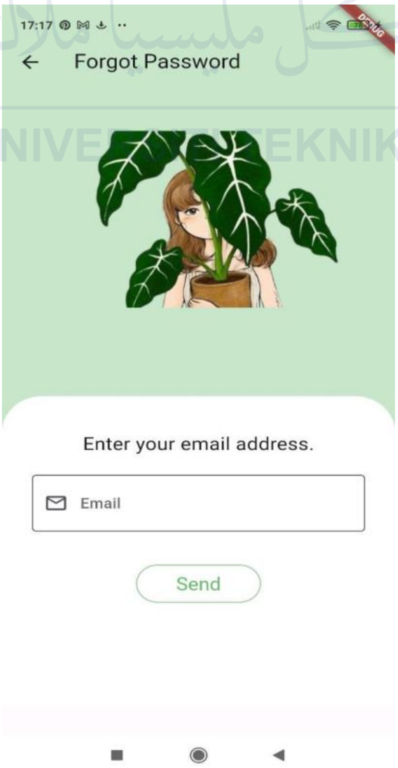


Figure 6.4: Forgot Password Email Page

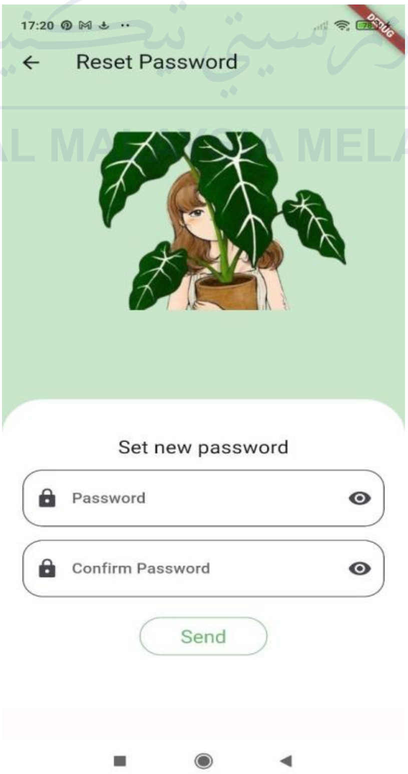


Figure 6.7: Set New Password Page

4. T04: Verify that when the user creates an account, they need to enter their username, email, password and re-enter the password.

Table 6.8: Test Case UC03

Test Case ID:	UC03
Module Name	Authentication Module
Use Case Name	Register Account
Use Case Description	Create a FloraHub account is the function to register account new FloraHub account
Actor	User
Pre-Conditions	-
Test Data	Username: MarkiahOwsem Email: markiah@gmail.com Password:abc1234_ Confirm Password: abc1234_
Basic Flow	1. User enter “Username”. 2. User click “Email”. 3. User enter the email. 4. User click “Password”. 5. User enter password. 6. User click “Confirm Password”. 7. User enter confirm password. 8. User click “Sign in” button.
Post Conditions	Account successfully sign up.
Exception Flows	Error message will display if user does not fill in any required fields (username, email, password and confirm password).
Expected Result	The system successfully creates the account after validating the entered information, and the user is redirected to the login page.
Actual Result	The system successfully creates the account after validating the entered information, and the user is redirected to the login page.
Status	Success

Figure 6.8 shows the "Registration Success" page, where the user is redirected after successfully creating an account with valid details. Conversely, **Figure 6.9** displays the error message shown when required fields such as username, email, password, or confirm password are missing, preventing account creation.

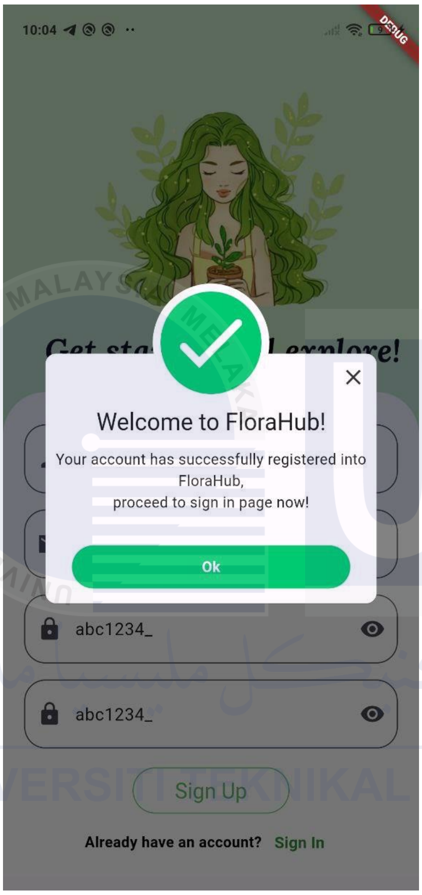


Figure 6.8: Registration Success Page

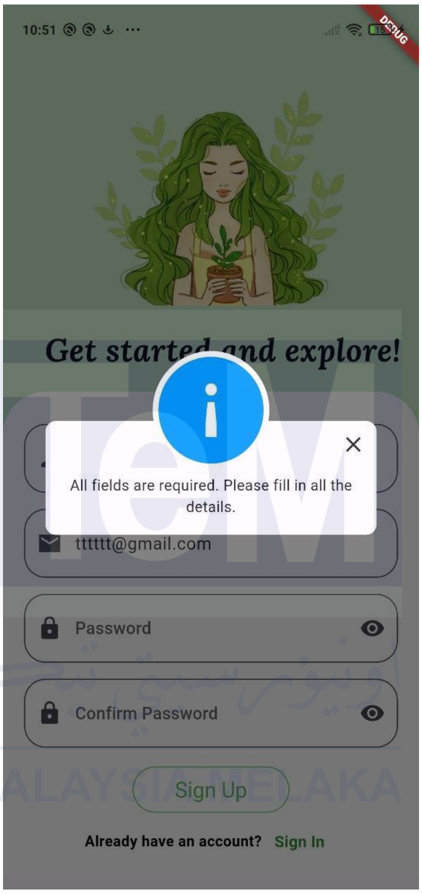


Figure 6.9: Error Message for Missing Required Fields

5. T05: Verify that users can update their username successfully.

Table 6.9: Test Case UC04

Test Case ID:	UC04
Module Name	Authentication Module
Use Case Name	Change username.

Use Case Description	Allow user to change and update their username.
Actor	User
Pre-Conditions	The user has already log in their account.
Test Data	Username: yanaezani
Basic Flow	1. User are in the "Profile" page. 2. User click on "Edit Profile" button. 3. User entered their desired username to change in the username fields. 4. User enter "Save" button.
Post Conditions	Username successfully changed.
Exception Flows	Error message will display.
Expected Result	Username is changed and reflected in the system.
Actual Result	Username is changed and reflected in the system.
Status	Success

Figure 6.10 displays the "Greeting Message" shown after a user successfully changes their username, confirming the update was successful. **Figure 6.11** illustrates the "Update Profile" page where users enter their new username and save the changes.

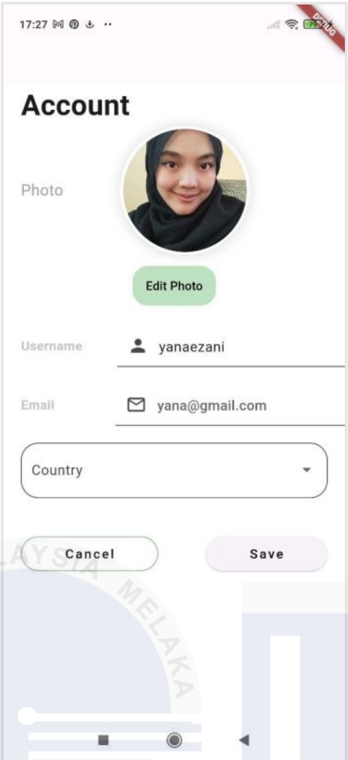


Figure 6.11: Update Profile Change

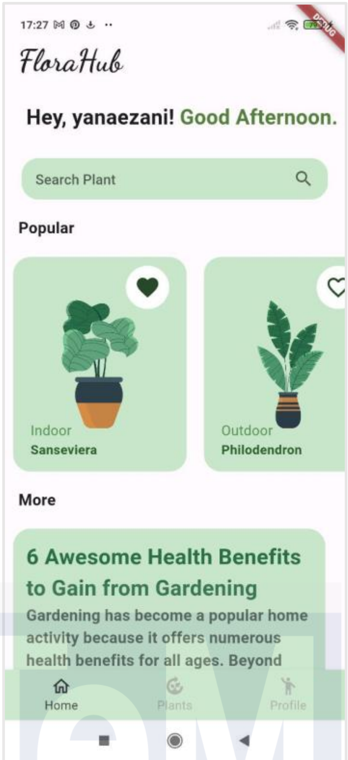


Figure 6.10: Greeting Message After Username Change

6. T06: Verify that users can upload and change their profile image.

Table 6.10: Test Case UC05

Test Case ID:	UC05
Module Name	Authentication Module
Use Case Name	Update Profile
Use Case Description	Allow user to change and update their profile image.
Actor	User
Pre-Conditions	The user has successfully logged into their account and is currently viewing the Settings Page, as shown in Figure 6.12.
Test Data	Username: yanaezani
Basic Flow	1. User are in the "Profile" page. 2. User click on "Edit Profile" button.

	3. User click on "Edit Photo" button and a popup message display for user to choose either upload image using camera or pick a photo from gallery. 4. User choose their photo and upload their desired photo. 5. User enter "Save" button.
Post Conditions	Profile image successfully changed.
Exception Flows	Error message will display.
Expected Result	Profile image is updated successfully.
Actual Result	Profile image is updated successfully.
Status	Success

Figure 6.12 shows the "Settings Page," where users initiate the process to change their profile image. This figure provides context for how users interact with the interface to upload a new profile photo.

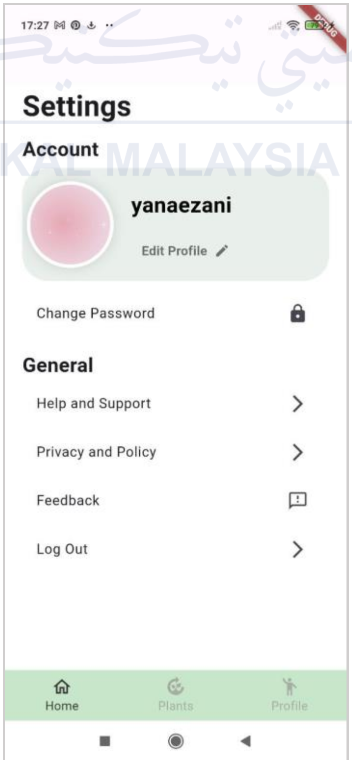


Figure 6.12: Settings Page

7. T07: Verify that the system sends a notification when soil moisture is dry and the watering is activated.

Table 6.11: Test Case UC06

Test Case ID:	UC06
Module Name	Notification Mo
Use Case Name	Receive Notification for Dry Soil and Watering Activation
Use Case Description	This test case verifies that the system sends a notification when the soil moisture level is dry and the watering system is activated.
Actor	User
Pre-Conditions	The user has already log in their account.
Test Data	-
Basic Flow	1. The system continuously monitors soil moisture levels. 2. Soil moisture levels fall below the predetermined threshold. 3. The system identifies that the soil is dry. 4. The system activates the watering system to restore moisture levels. 5. The system sends a notification to the user indicating that the soil moisture is dry and the watering system has been activated.
Post Conditions	1. The user receives a notification on their mobile device informing them of the dry soil condition and the subsequent activation of the watering system. 2. The soil moisture level begins to increase as a result of watering.
Alternate Flows	If the soil moisture level does not fall below the threshold, the system does not send a notification, and watering is not activated.
Expected Result	The system successfully sends a notification when the soil moisture level is dry and the watering system is activated.

Actual Result	The system successfully sends a notification when the soil moisture level is dry and the watering system is activated.
Status	Success

Figure 6.13 shows the system alert that notifies users about dry soil conditions and the activation of the watering system. This figure illustrates the notification users receive on their mobile device, confirming that the system has detected low soil moisture and has started watering to address the issue.

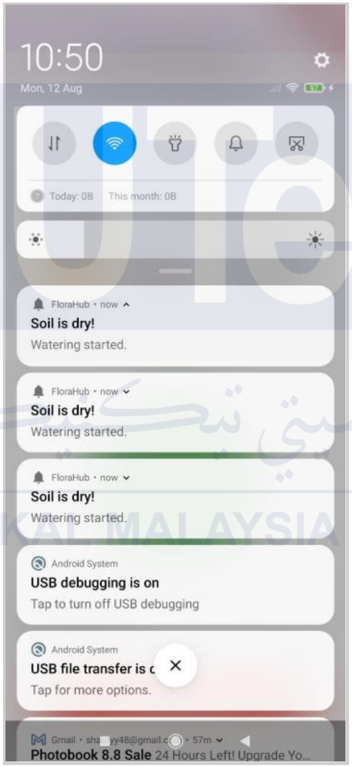


Figure 6.13: The system alert to the user about the dry soil

8. T08: Verify that the system sends a notification when watering is deactivated.

Table 6.12: Test Case UC07

Test Case ID:	UC07
Module Name	Notification Module
Use Case Name	Receive Notification for Watering Deactivation

Use Case Description	This test case verifies that the system sends a notification when the soil moisture level is moist and the watering system is deactivated.
Actor	User
Pre-Conditions	The user has already logged in to their account.
Test Data	-
Basic Flow	1. The system continuously monitors soil moisture levels. 2. Soil moisture levels fall below the predetermined threshold. 3. The system identifies that the soil is dry. 4. The system activates the watering system to restore moisture levels. 5. The system sends a notification to the user indicating that the soil moisture is dry and the watering system has been activated.
Post Conditions	1. The user receives a notification on their mobile device informing them soil condition are already moist and the watering system has already stop.
Alternate Flows	-
Expected Result	A notification is sent to the user indicating watering has stopped.
Actual Result	A notification is sent to the user indicating watering has stopped.
Status	Success

Figure 6.14 illustrates the system alert that notifies users when the watering system has been deactivated due to sufficient soil moisture. This figure displays users' notifications on their mobile device, indicating that the soil condition is now moist and the system has stopped watering.

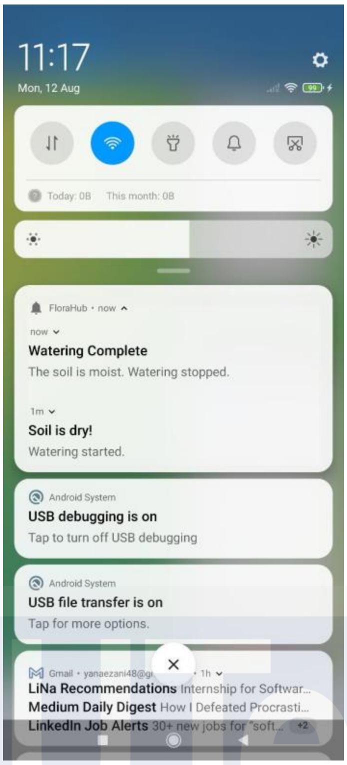


Figure 6.14: The system alert to the user about the dry soil

9. T10: Verify that the user can manually initiate watering through the mobile app.

Table 6.13: Test Case UC08

Test Case ID:	UC08
Module Name	Notification Module
Use Case Name	Manual Watering
Use Case Description	The user receives a pop-up notification when watering is activated after they manually activate the water pump to water their desired plant.
Actor	User
Pre-Conditions	The user has already log in their account and have added their specific plants to water them.
Test Data	-
Basic Flow	1. User are in "Plants" page. 2. User click the desired plant to water them. 3. User click the "Manual Watering" button. 4. User click the "Activate Pump" button.

Post Conditions	-
Exception Flows	Error message will display if user does not fill in any required fields (username, email, password and confirm password).
Expected Result	Watering starts as soon as the user initiates it.
Actual Result	Watering starts as soon as the user initiates it.
Status	Success

Figure 6.15 displays the success message shown on the screen after the user manually activates the watering system. This figure confirms that the water pump has been successfully activated for the selected plant.

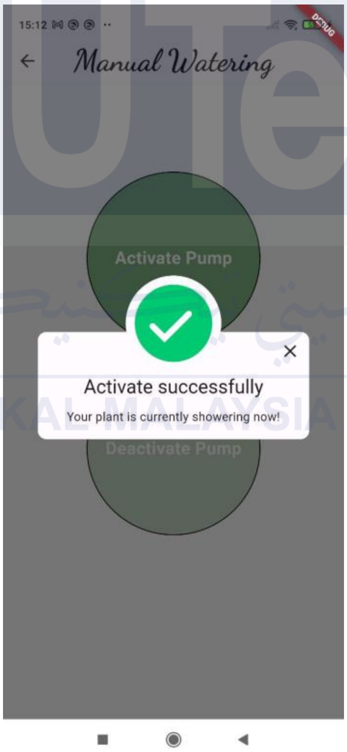


Figure 6.15: Success message appears on the screen indicating that manual watering has been activated.

10. T11: Verify that the system can stop watering manually.

Table 6.14: Test Case UC09

Test Case ID:	UC09
Module Name	Notification Module
Use Case Name	Manual Watering
Use Case Description	The user receives a pop-up notification when watering is deactivated after the user manually activates the water pump to water their desired plant.
Actor	User
Pre-Conditions	1. The user has already log in their account 2. The user is already at the specific plant's detailed page. 3. The user have click the "Activate Pump" button inside "Manual Watering" page.
Test Data	-
Basic Flow	1. User click the "Deactivate Pump" button.
Post Conditions	A pop-up notification appear, notify the user that the water pump has stop water their plants.
Alternate Flows	-
Expected Result	Watering stops immediately when the user cancels it.
Actual Result	Watering stops immediately when the user cancels it.
Status	Success

Figure 6.16 shows the success message displayed on the screen when manual watering is deactivated. This figure confirms that the water pump has been successfully turned off.

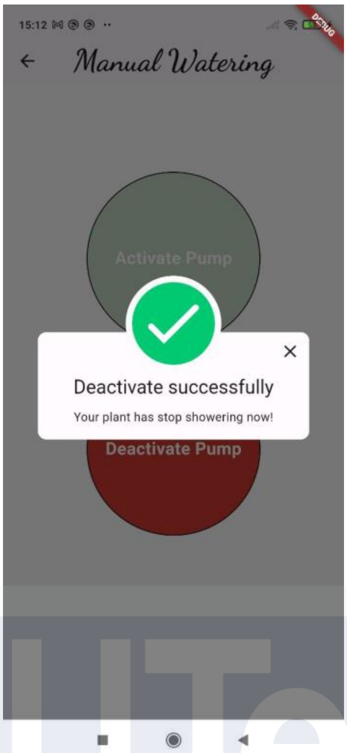


Figure 6.16: Success message on the screen indicating that manual watering has been deactivated.

11. T12: Verify that scheduled watering data is saved correctly in the system.

Table 6.15: Test Case UC10

Test Case ID:	UC10
Module Name	Soil Moisture Module
Use Case Name	Schedule Watering
Use Case Description	The user interacts with the FloraHub system to set up a new watering schedule for a specific plant.
Actor	User
Pre-Conditions	1. The user has already logged in to their account. 2. The user is on the specific plant's detailed page.
Test Data	Schedule Time: 6:30PM Duration: 60 seconds
Basic Flow	1. The user click on the "Schedule Watering" button and the system will display the Schedule Watering page, as shown in Figure 6.16.

	2. The user enter the time desired to schedule the watering. 3. User enter the duration of water pumps to be activated. 4. User click “Add Schedule” button.
Post Conditions	A pop-up notification appear, notify the user that the schedule is successfully added.
Alternate Flows	-
Expected Result	Scheduled watering data is stored accurately.
Actual Result	Scheduled watering data is stored accurately.
Status	Success

Figure 6.17 illustrates the success message displayed on the screen when a new watering schedule is added successfully. This confirms that the schedule was saved correctly. **Figure 6.18** shows the “Schedule Watering” page where the user sets up a new watering schedule. The test case involves the user selecting a time and duration for watering and clicking “Add Schedule.” The expected result is a pop-up notification indicating that the schedule was successfully added, and the actual result confirms that the data was saved accurately.

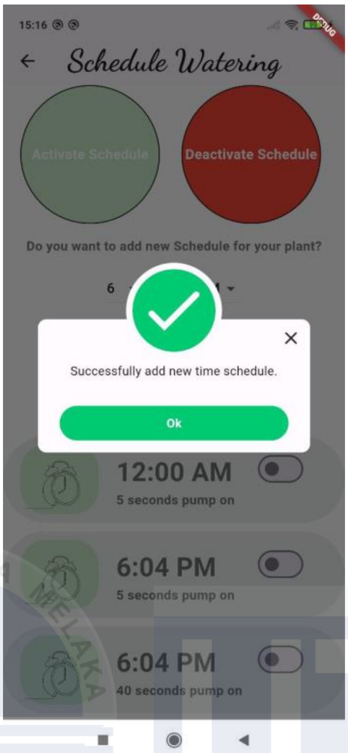


Figure 6.18: Success Message for Adding New Time Schedule

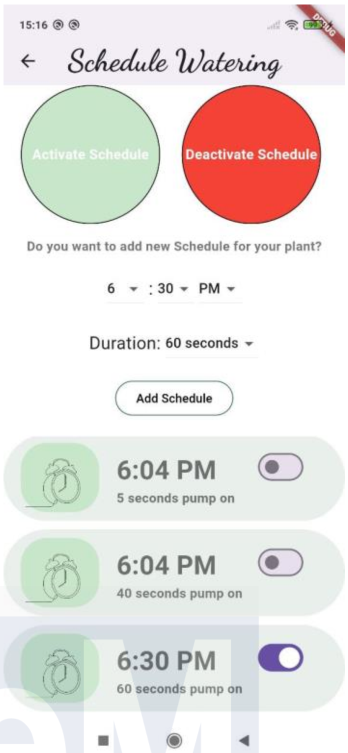


Figure 6.17: Schedule Watering Page

12. T13: Verify that the system executes scheduled watering tasks as planned.

Table 6.16: Test Case UC11

Test Case ID:	UC11
Module Name	Soil Moisture Module
Use Case Name	Schedule Watering
Use Case Description	The user interacts with the FloraHub system to set up a new watering schedule for a specific plant.
Actor	System
Pre-Conditions	1. The user has already logged in to their account. 2. The user is on the specific plant's detailed page.
Test Data	Schedule Time: 6:30PM Duration: 60 seconds
Basic Flow	1. The user click on the "Schedule Watering" button and the system will display the Schedule Watering page.

	2. The user enter the time desired to schedule the watering. 3. User enter the duration of water pumps to be activated. 4. User click “Add Schedule” button.
Post Conditions	A pop-up notification appear, notify the user that the schedule is successfully added.
Alternate Flows	-
Expected Result	Scheduled watering data is stored accurately.
Actual Result	Scheduled watering data is stored accurately.
Status	Success

Figure 6.19 demonstrates the system executing scheduled watering tasks as planned. This figure confirms that when the scheduled time matches the current time, the system performs the watering task as specified. In contrast, the **Table 6.16: Test Case UC11** outlines the process for setting up a watering schedule and verifying that it executes correctly. The expected result is that the system accurately follows the scheduled time and duration for watering, as shown in Figure 6.19, where the system successfully executes the watering task according to the plan.

Figure 6.19: The system successfully executes schedules watering task as planned

13. T14: Verify that soil moisture and watering data is displayed in Grafana dashboards.

Table 6.17: Test Case UC12

Test Case ID:	UC12
Module Name	Analytics Module

Use Case Name	Reporting
Use Case Description	The soil moisture pattern and watering data are displayed correctly to the user in Grafana dashboard so the user can know more about their plants soil and water pattern.
Actor	User
Pre Conditions	-
Test Data	Dummy data of the soil moisture data and water data
Basic Flow	-
Post Conditions	-
Alternate Flows	-
Expected Result	Data is visualized appropriately in Grafana.
Actual Result	Data is visualized appropriately in Grafana.
Status	Success

Figure 6.20 and **Figure 6.21** illustrate the Grafana dashboards displaying soil moisture trends and water pattern trends using dummy data. These figures show how the data is presented to the user, allowing them to view and analyze their plant's soil moisture and watering patterns effectively.

Figure 6.20: Grafana Dashboard for Soil Moisture Trends using Dummy Data

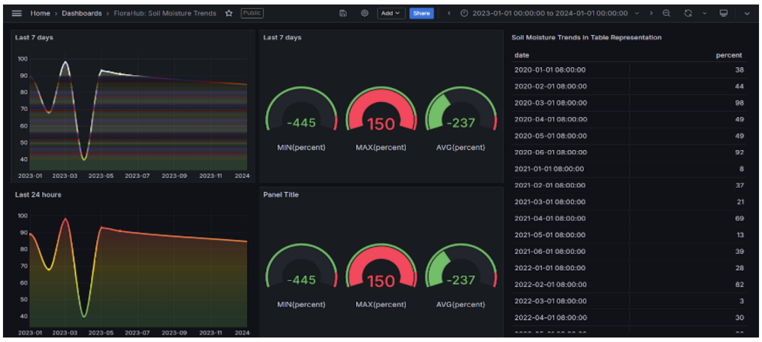


Figure 6.21: Grafana Dashboard Water Pattern Trends using Dummy Data

14. T15: Verify that users can view data overview to explore water cost based on their plants.

Table 6.18: Test Case UC13

Test Case ID:	UC13
Module Name	Analytics Module
Use Case Name	Reporting
Use Case Description	The system calculates water cost based on data collected from water volume. Reports display water cost in monthly and yearly analysis (in Ringgit Malaysia).
Actor	User
Pre Conditions	1. User has already log in to their FloraHub account. 2. User in their current plant's detail page.
Test Data	-
Basic Flow	1. User click "Data Overview" button. 2. User click "View Reports" button. 3. User click "Water Cost" button.
Post Conditions	-
Alternate Flows	-
Expected Result	Dashboards provide interactive features for data exploration.
Actual Result	Dashboards provide interactive features for data exploration.
Status	Success

Figure 6.22 shows the Water Cost Report Page, where users can view detailed reports on water costs associated with their plants. **Table 6.18: Test Case UC13** details the process of verifying that users can access and explore water cost data. The expected result, as depicted in Figure 6.22, is that the system displays interactive and detailed water cost reports, allowing users to analyze their water expenses effectively.

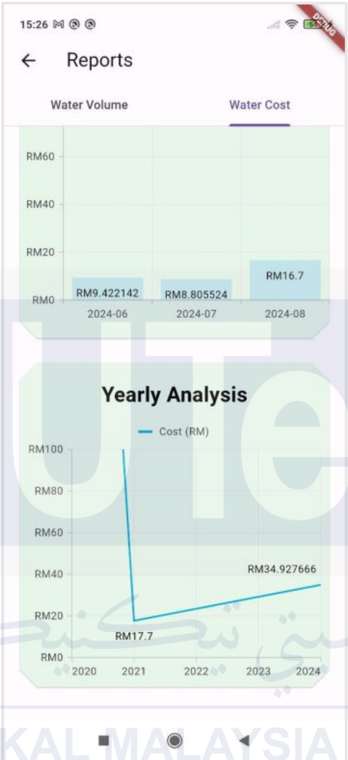


Figure 6.22: Water Cost Report Page

15. T16: Verify that the system generates daily, monthly, and yearly water usage reports.

Table 6.19: Test Case UC14

Test Case ID:	UC14
Module Name	Analytics Module
Use Case Name	Reporting
Use Case Description	User can check water reports such as water volume based on their plants that the system generates daily, monthly and yearly in (cubic metre).
Actor	User

Pre-Conditions	1. User has already log in to their FloraHub account. 2. User in their current plant's detail page.
Test Data	-
Basic Flow	1. User click "Data Overview" button. 2. User click "View Reports" button.
Post Conditions	The reports page will be displayed to the user.
Alternate Flows	-
Expected Result	Reports contain water volume in cubic metre.
Actual Result	Reports contain water volume in cubic metre.
Status	Success

Figure 6.23 illustrates the Water Cost Reports Page, where users can view detailed water usage reports. **Table 6.19: Test Case UC14** outlines the process for verifying that users can access daily, monthly, and yearly water usage reports. The expected result, as shown in Figure 6.23, is that the system generates and displays water volume reports in cubic meters for daily, monthly, and yearly periods. This allows users to effectively monitor and analyze their water usage over different time frames.

Figure 6.23: Water Cost Reports Page

6.6.2 Test Analysis

During the development of the FloraHub: Smart Plant Hydration System, two distinct questionnaires were conducted to gather feedback at different stages. The first questionnaire was carried out during the FYP Innovation Competition on 12 June 2024 as part of Projek Sarjana Muda 1 (PSM 1). It gathered responses from 9 participants, including 2 UTEM lecturers, 1 professional from Infineon, and 6 UTEM students. This questionnaire aimed to collect initial feedback on the system's functionality, innovation, and usability, providing valuable insights for early-stage improvements.

The second questionnaire, conducted during Projek Sarjana Muda 2 (PSM 2), was distributed online through social media platforms such as Instagram and WhatsApp. It received responses from 35 participants and focused on evaluating the overall system functionality. The broader feedback obtained from this diverse audience was instrumental in identifying areas for refinement and ensuring the system met its performance goals. Both questionnaires played essential roles in shaping the FloraHub system throughout its development.

1. Questionnaire 1 (PSM 1)

The following is an analysis of the responses to the questionnaire sent to participants as part of the system test:

A. Section 1: Respondent Background

a. Respondent's university

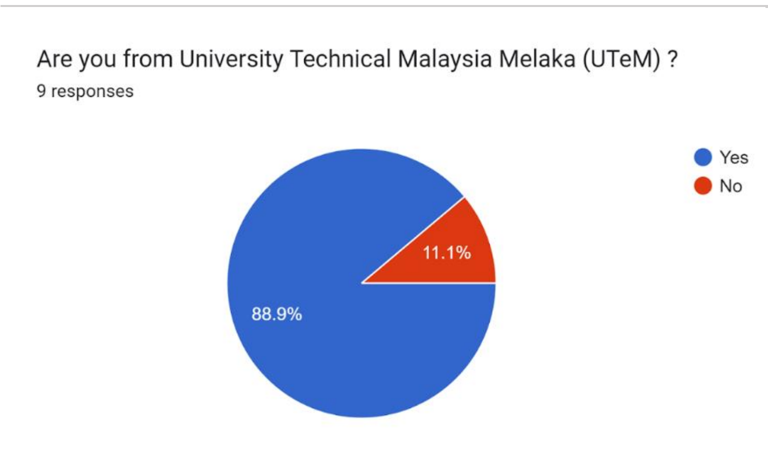


Figure 6.24: Survey Responses RegardingUniversity Affiliation

Table 6.20: Survey Results Summary Regarding University Affiliation

Results	Number of respondents	Percentage
No	1	11.1%
Yes	8	88.9%
Total	**9**	**100%**

Figure 6.23 and Table 17 present the results of a survey asking participants if they are from University Technical Malaysia Melaka (UTeM). The pie chart in Figure 6.23 shows that out of 9 respondents, the vast majority (88.9%) answered "Yes," while only 11.1% answered "No." Table 17 provides a more detailed breakdown of these results, indicating that 8 respondents confirmed their affiliation with UTeM, and 1 respondent indicated they were not affiliated. Together, the figure and table effectively illustrate that the majority of participants are from UTeM.

b. Respondent's faculty

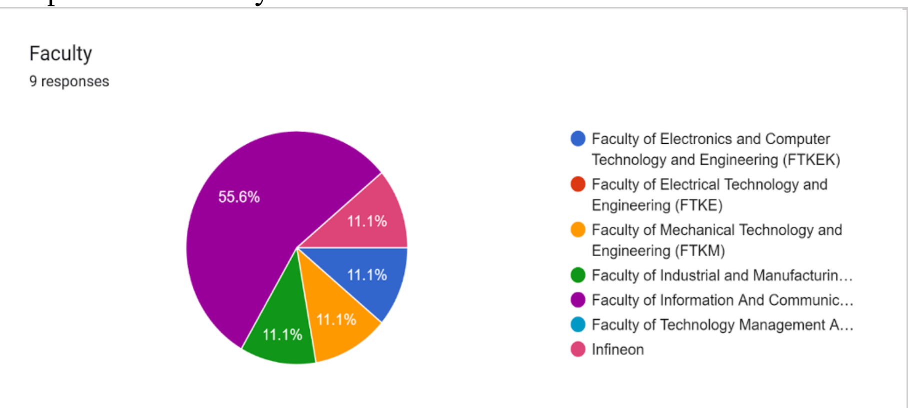


Figure 6.25: Faculty Affiliation Survey Results

Table 6.21: Faculty Affiliation Survey Results Summary

Results	Number of respondents	Percentage
FTKEK	1	11.1%
FTKE	0	0%
FTKM	1	11.1%
FTKIP	1	11.1%
FTMK	5	55.6%
FPTT	0	0%
INFINEON	1	11.1%
Total	**9**	**100%**

Figure 6.24 and Table 18 present the results of a survey asking participants about their faculty affiliation. The pie chart in Figure 6.24 shows that out of 9 respondents, the Faculty of Information and Communicationl Technology (FTMK) was the most popular choice, receiving 55.6% of the responses. Other faculties with significant representation include FTKE and FTKIP, each receiving 11.1% of the responses. Table 18 provides a more detailed breakdown of these results, indicating that 5 respondents were affiliated with FTMK, while 1 respondent was affiliated with each of FTKE, FTKIP, FTTM, and INFINNEON. The remaining faculties (FTKM, FPPTT) did not receive any responses.

B. Section 2: Experience with FloraHub

a) Respondent's rate their satisfaction towards the automatic watering feature in FloraHub app

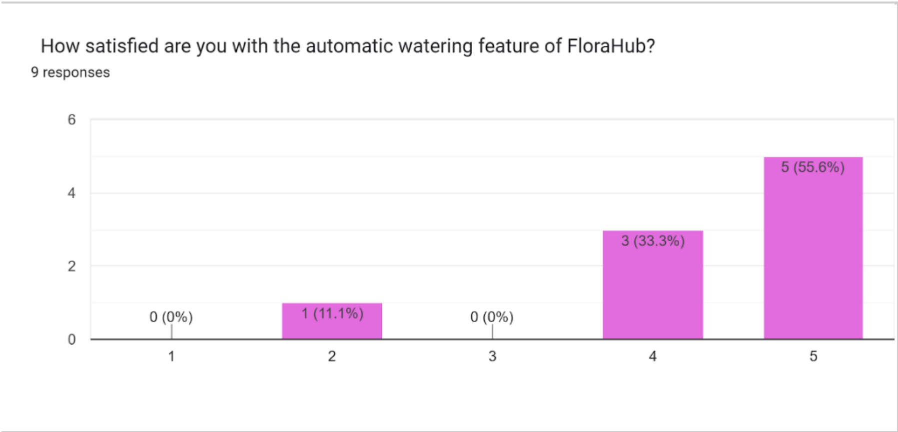


Figure 6.26: User Satisfaction with FloraHub's Automatic Watering

Table 6.22: Summary of User Satisfaction with FloraHub's Automatic Watering

Rate	Number of respondents	Percentage
Very Unsatisfied (1)	0	0%
Unsatisfied (2)	1	11.1%
Neutral (3)	0	0%
Satisfied (4)	3	33.3%
Very Satisfied (5)	5	55.6%
Total	**9**	**100%**

Figure 6.25 and Table 19 present the findings from a survey evaluating user satisfaction with the automatic watering feature of the FloraHub app. Figure 6.25 illustrates that out of the 9 respondents, the majority expressed high levels of satisfaction, with 55.6% (5 respondents) rating their experience as "Very Satisfied" and 33.3% (3 respondents) rating it as "Satisfied." Only one respondent (11.1%) indicated dissatisfaction by selecting the "Unsatisfied" option. Table 19 further breaks down these results, showing that no respondents chose "Very Unsatisfied" or "Neutral," confirming an overall positive reception of the automatic watering feature.

b. Respondent’s rate the usefulness of real-time data updates on soil moisture and water flow.

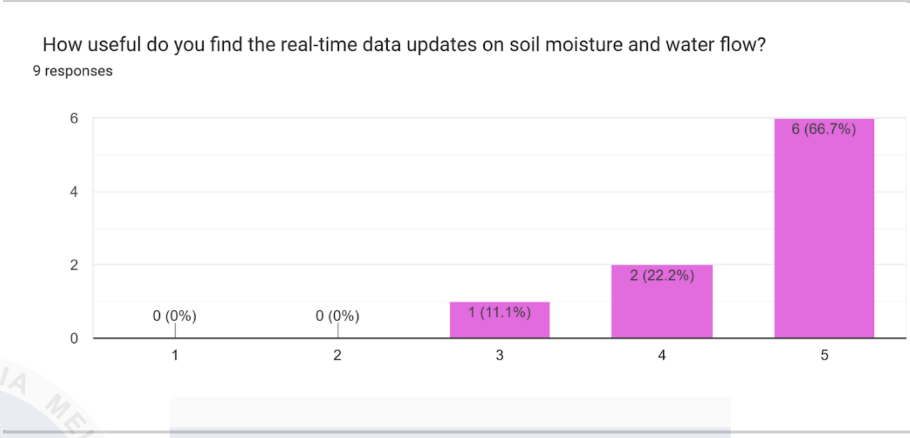


Figure 6.27: Respondents' Ratings on the Usefulness of Real-Time Data Updates on Soil Moisture and Water Flow

Table 6.23: Summary of Respondents' Ratings on the Usefulness of Real-Time Data Updates

Rate	Number of respondents	Percentage
Not helpful at all (1)	0	0%
Slightly helpful (2)	0	0%
Moderately helpful (3)	1	11.1%
Very helpful (4)	2	22.2%
Extremely helpful (5)	6	66.7%
Total	**9**	**100%**

Figure 6.26 and Table 20 display the results from a survey where respondents rated the usefulness of real-time data updates on soil moisture and water flow. According to Figure 6.26, the majority of respondents found the feature highly valuable, with 66.7% (6 respondents) rating it as "Extremely Helpful," and 22.2% (2 respondents) considering it "Very Helpful." Only 11.1% (1 respondent) rated the feature as "Moderately Helpful," and none of the respondents selected the lower ratings of "Slightly Helpful" or "Not Helpful at All."

Table 20 provides a detailed summary of these ratings, confirming that all 9 respondents recognized the usefulness of the real-time data updates, with the majority offering the highest level of praise for this feature.

c. Respondent's rate of how useful the dashboard analysis that are provided by FloraHub app.

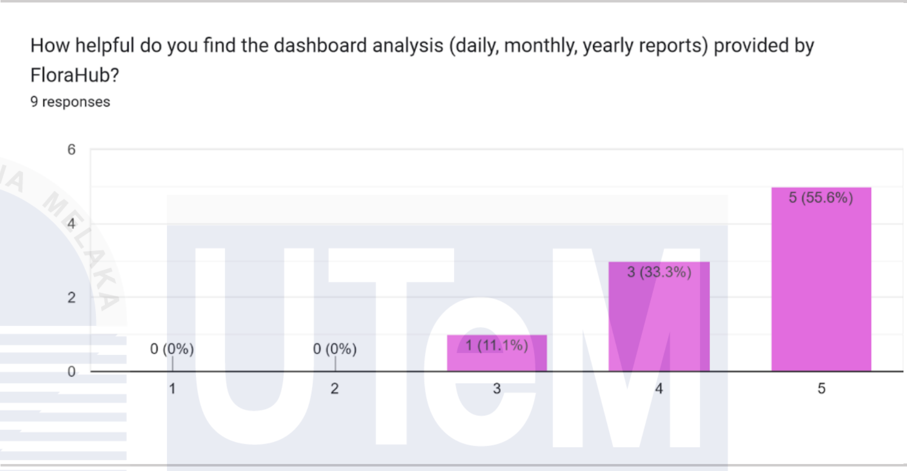


Figure 6.28: Respondents' Rating of the Usefulness of FloraHub App's Dashboard Analysis

Table 6.24: Distribution of Respondents Based on Their Perception of the Usefulness of FloraHub's Dashboard Analysis

Rate	Number of respondents	Percentage
Not helpful at all (1)	0	0%
Slightly helpful (2)	0	0%
Moderately helpful (3)	1	11.1%
Very helpful (4)	3	33.3%
Extremely helpful (5)	5	55.6%
Total	**9**	**100%**

Figure 6.27 and Table 21 present the feedback of 9 respondents regarding the usefulness of FloraHub's dashboard analysis, which provides daily, monthly, and yearly reports. The bar chart in Figure 6.27 visualizes how respondents rated the dashboard analysis on a scale of 1 to 5, where 1 indicates "Not helpful at all" and 5 indicates "Extremely helpful". This data shows that the majority of users find the dashboard analysis very or extremely useful, indicating that the reporting features of the FloraHub app are well-received and meet user expectations.

2. Questionnaire 2 (PSM 2)

The results of the questionnaire that was given to respondents in order to test the system are examined as follows:

A. Section 1: Respondent Background

a. Respondent’s university

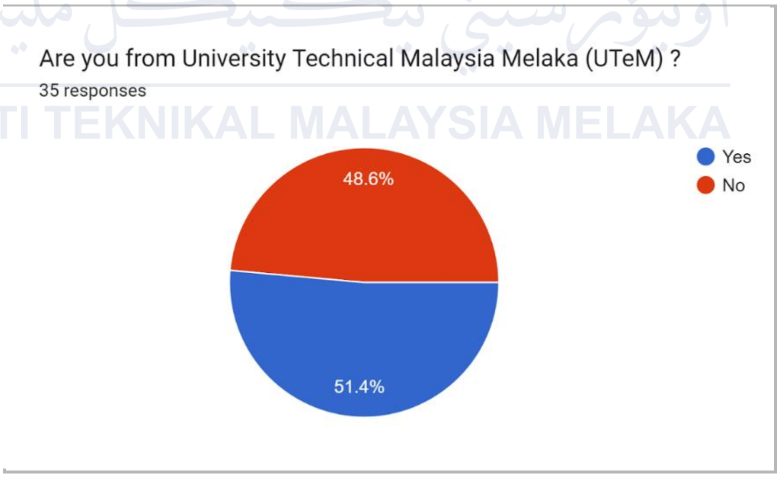


Figure 6.29: Respondents' Affiliation with University Technical Malaysia Melaka (UTeM)

Table 6.25: Distribution of Respondents Based on Affiliation with University Technical Malaysia Melaka (UTeM)

Results	Number of respondents	Percentage
No	17	48.6%
Yes	18	51.4%
Total	**35**	**100%**

Figure 6.28 and **Table 22** present the results indicate that out of 35 respondents, 51.4% (18 individuals) are from University Technical Malaysia Melaka (UTeM), while 48.6% (17 individuals) are not. This reflects a fairly balanced distribution of respondents with a slight majority being affiliated with UTeM.

b. Respondent's gender

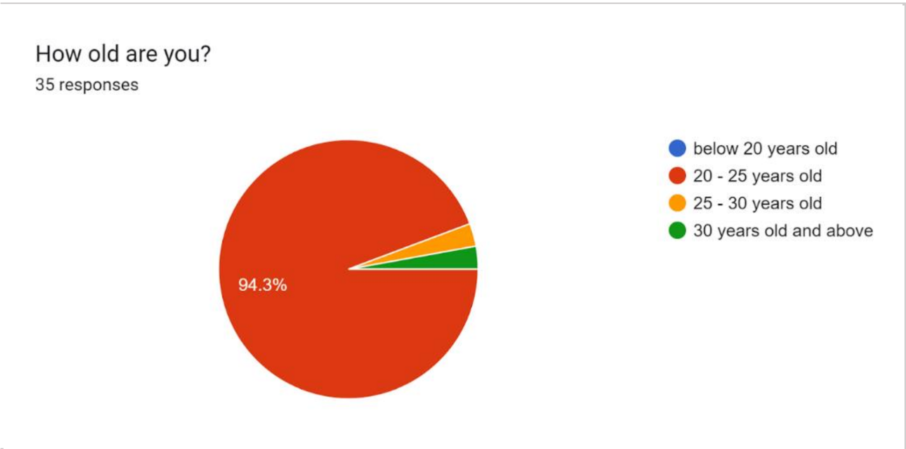


Figure 6.30: Respondent Age Demographics

Table 6.26: Summary of Respondent Ages

Age	Number of respondents	Percentage
below 20 years old	0	0%
20 - 25 years old	33	94.3%
25 - 30 years old	1	2.9%
30 years old and above	1	2.9%
Total	**35**	**100%**

Figure 6.29 and Table 23 present the results indicate that the majority of respondents, 94.3% (33 individuals), are between 20-25 years old. Additionally, 2.9% (1 individual) of the respondents fall into the 25-30 years old category, and another 2.9% (1 individual) are 30 years old and above. No respondents were below 20 years old, making the total sample population 35 individuals.

c. Respondent's occupation

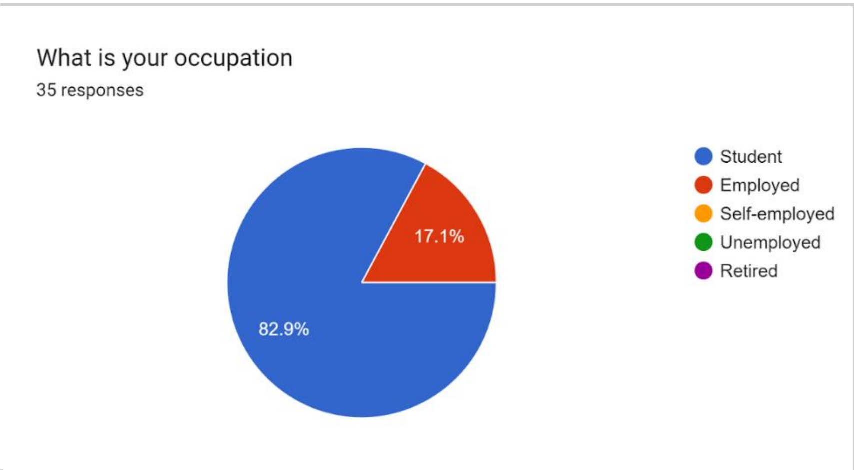


Figure 6.31: Occupational Distribution of Respondents

Table 6.27: Summary of Respondent Occupations

Occupation	Number of respondents	Percentage
Student	29	82.9%
Employed	6	17.1%
Self-employed	0	0%
Unemployed	0	0%
Retired	0	0%
Total	**35**	**100%**

d. Respondent's experienced with gardening or plant care

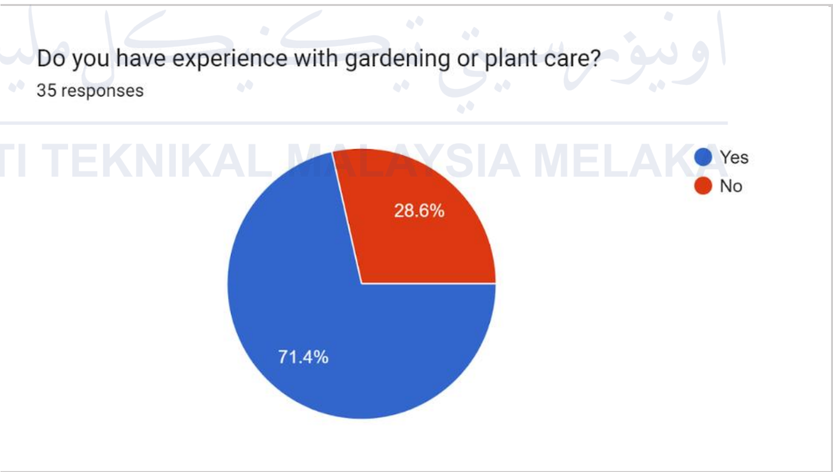


Figure 6.32: Respondent Gardening Experience

Table 6.28: Summary of Gardening Experience

Results	Number of respondents	Percentage

Yes	25	71.4%
No	10	28.6%
Total	**35**	**100%**

Figure 6.31 and Table 25 present the data collected from the respondent forms reveals the level of gardening experience among participants. A majority of respondents (71.4%) indicated having some level of experience with gardening or plant care, while 28.6% reported having no experience.

e. The quantity of plant that respondent's currently taking care.

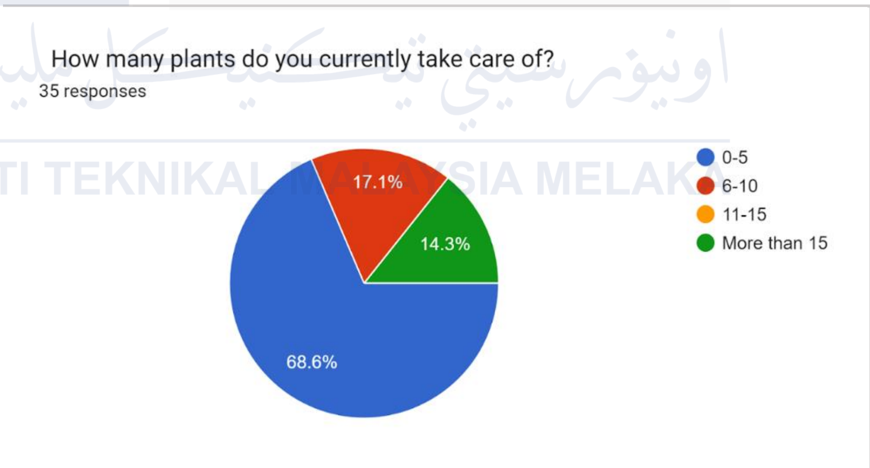


Figure 6.33: Number of Plants Cared For

Table 6.29: Summary of Plant Quantities

Quantity of Plant	Number of respondents	Percentage
0 - 5	24	68.6%

6 - 10	6	17.1%
11 - 15	0	0%
More than 15	5	14.3%
Total	**35**	**100%**

Figure 6.32 and Table 26 shows the data collected from the respondent forms reveals the number of plants currently being cared for by participants. The majority of respondents (68.6%) own 0-5 plants, indicating a relatively small number of plants per user. This information suggests that a significant portion of the FloraHub user base has a manageable number of plants, which may influence their specific needs and preferences for plant care tools and resources.

f. The type of plants that the Respondent's mostly care of.

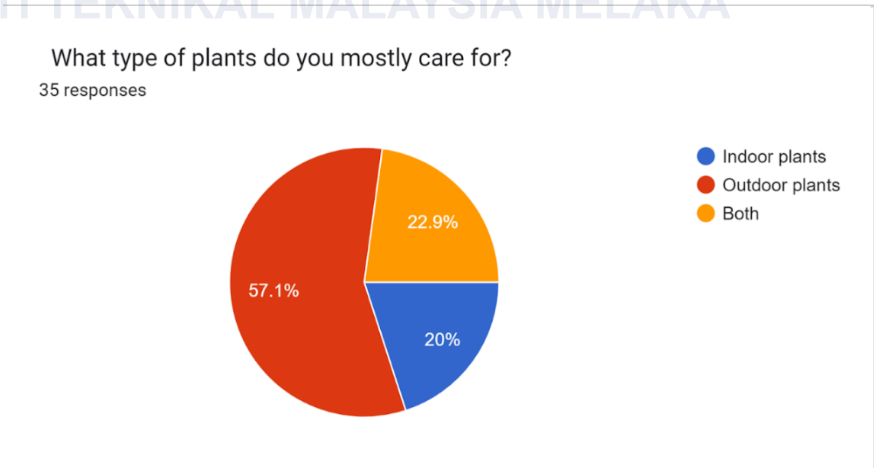


Figure 6.34: Respondent Plant Type Preference

Table 6.30: Summary of Plant Type Preferences

Type of Plant	Number of respondents	Percentage
Indoor Plants	7	20%
Outdoor Plants	20	57.1%
Both	8	22.9%
Total	**35**	**100%**

Figure 6.33 and Table 27 shows the data collected from the respondent forms reveals the types of plants that participants primarily care for. The majority of respondents (57.1%) prefer outdoor plants, followed by those who care for both indoor and outdoor plants (22.9%). A smaller percentage (20%) focus solely on indoor plants. This information suggests that outdoor plants are the most popular among FloraHub users, although there is a significant interest in caring for both indoor and outdoor plants.

B. Section 2: Experience with FloraHub

a) How user-friendly do the Respondent's find the FloraHub app interface?

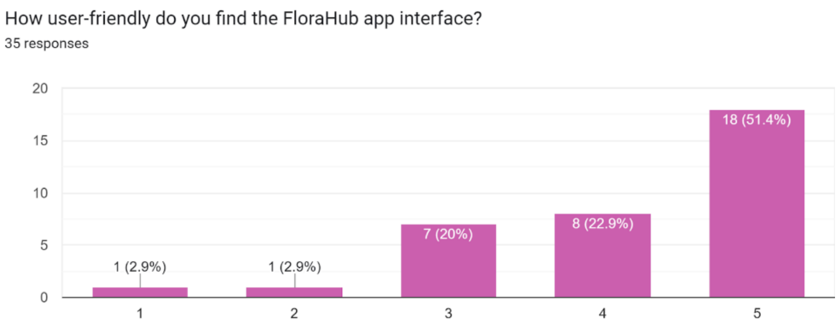


Figure 6.35: Respondent FloraHub App User-Friendliness

Table 6.31: Summary of FloraHub App Ratings

Rate	Number of respondents	Percentage
Very difficult to use (1)	1	2.9%
Difficult to use (2)	1	2.9%
Moderate (3)	7	20%
User-friendly (4)	8	22.9%
Very user-friendly (5)	18	51.4%
Total	**35**	**100%**

The data collected from the respondent forms reveals a highly positive perception of the FloraHub app interface. A significant majority of respondents (51.4%) rated the interface as "Very User-Friendly," indicating that they find it intuitive and easy to navigate. This overwhelmingly positive feedback suggests that the FloraHub app interface is well-designed and effectively meets the needs of users, contributing to a positive overall user experience.

b. How effective does the Manual Watering method to the Respondent’s?

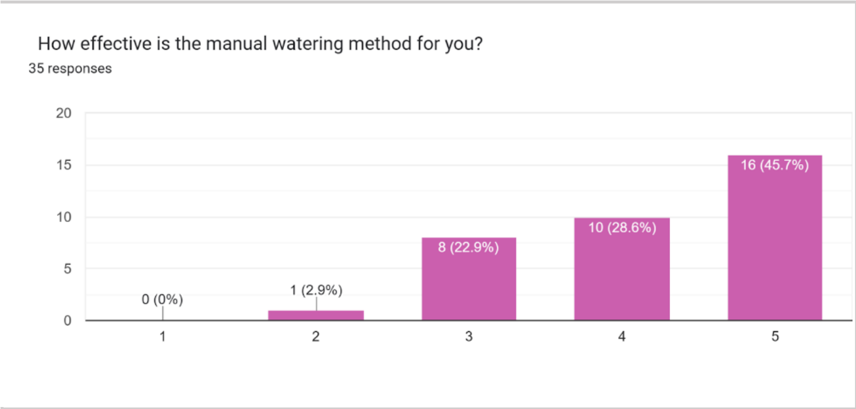


Figure 6.36: Respondent Manual Watering Effectiveness

Table 6.32: ***Summary of Manual Watering Ratings***

Rate	Number of respondents	Percentage
Very ineffective (1)	0	0%
Ineffective (2)	1	2.9%
Moderate (3)	8	22.9%
Effective (4)	10	28.6%
Very effective (5)	16	45.7%
Total	**35**	**100%**

Figure 6.35 and Table 29 show the data collected from the respondent forms reveals a generally positive perception of the manual watering method. A significant majority of respondents (45.7%) rated the method as "Very Effective," indicating that they find it to be a suitable and efficient approach for watering their plants. This positive feedback suggests that the manual watering method is well-regarded by FloraHub users, although there is a minority who find it less effective.

c. How convenient does the Respondent's think about the Scheduling Watering method?

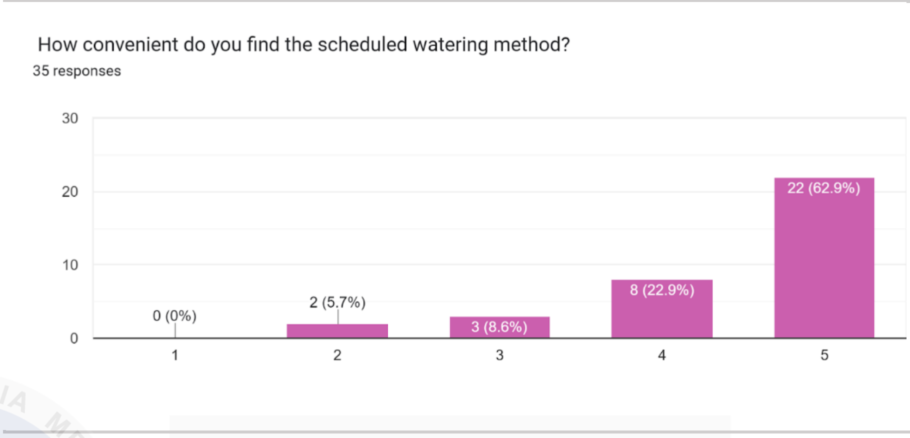


Figure 6.37: Respondent Scheduled Watering Convenience

Table 6.33: Summary of Scheduled Watering Ratings

Rate	Number of respondents	Percentage
Very inconvenient (1)	0	0%
Inconvenient (2)	2	5.7%
Neutral (3)	3	8.6%
Convenient (4)	8	22.9%
Very convenient (5)	22	62.9%
Total	**35**	**100%**

Figure 2.36 and Table 30 shows the data collected from the respondent forms reveals a highly positive perception of the scheduled watering method. A significant majority of respondents (62.9%) rated the method as "Very Convenient," indicating that they find it to be a user-friendly and time-saving feature. This overwhelmingly positive feedback

suggests that the scheduled watering feature is a valuable addition to the FloraHub app, providing users with a convenient way to automate their plant care routines.

d. Respondent's rate the overall performance of FloraHub System.

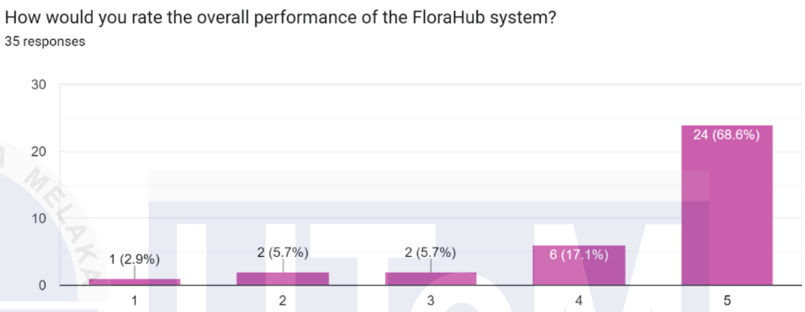


Figure 6.38: Respondent FloraHub System Overall Rating

Table 6.34: Summary of FloraHub System Ratings

Rate	Number of respondents	Percentage
Very poor (1)	1	2.9%
Poor (2)	2	5.7%
Neutral (3)	2	5.7%
Excellent (4)	6	17.1%
Very excellent (5)	24	68.6%
Total	**35**	**100%**

Figure 6.37 and Table 31 shows the data collected from the respondent forms reveals a highly positive perception of the FloraHub system. A large majority of respondents rated the system "Very Excellent"

(68.6%) or "Excellent" (17.1%), totaling 85.7% of respondents who expressed satisfaction. Only a small minority expressed dissatisfaction, with 5.7% rating it "Poor" and 2.9% rating it "Very Poor." This suggests that the FloraHub system effectively meets the needs of its users and provides a valuable plant care solution.

e. Respondent's rate the overall design and aesthetics of the FloraHub app.

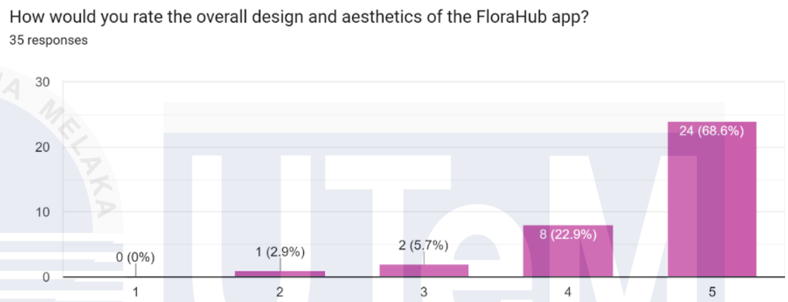


Figure 6.39: Respondent FloraHub App Design and Aesthetics Rating

Table 6.35: Summary of FloraHub App Design Ratings

Rate	Number of respondents	Percentage
Very poor (1)	0	2.9%
Poor (2)	1	2.9%
Neutral (3)	2	5.7%
Excellent (4)	8	22.9%
Very excellent (5)	24	68.6%
Total	**35**	**100%**

Figure 6.38 and Table 32 shows the data collected from the respondent forms reveals a highly positive perception of the FloraHub app's design

and aesthetics. A large majority of respondents rated the design as "Very Excellent" (68.6%) or "Excellent" (22.9%), totaling 91.5% of respondents who expressed satisfaction. This suggests that the FloraHub app's design is well-received by users and contributes to a positive overall user experience.

6.6.3 Result Analysis from both Questionnaire

The analysis of both questionnaires highlights the evolution of the FloraHub system from its early development stages to its current iteration, revealing valuable insights into its usability and features.

Questionnaire 1: Projek Sarjana Muda 1 (Usability Feedback and Suggestions)

The feedback from the first questionnaire, conducted during the FYP Innovation Competition, primarily focused on the early-stage functionality and potential areas for future improvements. Although some respondents had no specific comments, a few critical suggestions emerged:

- **Future Development Ideas**: Several respondents suggested adding an admin web application to improve system management, as well as the ability to differentiate between different types of plants. This included automated detection and care adjustments based on the plant's specific needs (e.g., humidity, water levels, sunlight).
- **System Usability**: While the feedback was generally positive with comments like "good" and "nothing," some respondents saw room for improvements, particularly in expanding system features to cater to multiple plant types simultaneously.

Questionnaire 2: Projek Sarjana Muda 2 (Evaluation of System Functionality)

The second questionnaire, distributed during PSM 2, provided more detailed insights into the user experience, with a significant focus on the system’s usability. The feedback from a broader audience, including plant enthusiasts and tech-savvy users, was overwhelmingly positive:

- **User-Friendly Design**: Respondents praised the system's graphical user interface (GUI), describing it as "nice," "easy to use," and "user-friendly." The convenience offered by FloraHub for busy individuals, along with its reliable features, resonated well with respondents. Many found the aesthetics and interface design to be well-organized and appealing.
- **Core Functionality**: The watering system and the ability to monitor pH levels were highlighted as significant strengths. Users appreciated the ability to schedule watering, which aligned with the target market of users who want to care for plants without dedicating constant attention.
- **Positive Feedback**: Comments such as "great initiative," "good job," and "recommended to all flower enthusiasts" indicated that FloraHub was well-received. Users saw it as a valuable tool for those with limited time for plant care.

The feedback from both questionnaires highlights the progress made between FYP 1 and FYP 2. Initially, during the physical competition, the respondents provided feedback that was not primarily focused on usability but rather on the system's potential for enhancements, particularly regarding scalability and plant differentiation. This feedback during FYP 1 served as a foundation for guiding the development process toward a more refined and versatile solution. As the system progressed and was completed, I decided to conduct a second round of feedback, which focused on the system's usability and overall functionality.

By FYP 2, the feedback shifted towards recognizing the system's user-friendly design, intuitive interface, and practical functionalities, reflecting the improvements made based on earlier suggestions. Key recommendations for future development, such as integrating auto-fertilization, enhancing the user experience, and providing more detailed plant information, highlight areas for further growth. Overall, the respondents expressed high satisfaction with FloraHub, recognizing it as a convenient and innovative solution for plant care, especially for those with busy lifestyles. The strong positive feedback and the thoughtful suggestions indicate a promising future for FloraHub, with the potential to expand its features and reach a broader audience.

6.7 Conclusion

In this chapter, a comprehensive analysis of the testing strategies employed for the FloraHub system has been provided. Detailed testing phases, including Unit Testing, Integration Testing, System Testing, Acceptance Testing, and Usability Testing, were discussed to ensure the system met both functional and non-functional requirements. The test environment, including the necessary hardware and software configurations, was defined to facilitate accurate testing conditions. The results from both questionnaires highlighted the system's progression, from early feedback focused on functionality improvements to later praise for its user-friendly design and practical features. These insights were critical in refining the system and ensuring its readiness for deployment. In the final chapter, the outcomes of the research project will be reviewed, alongside a broader discussion of its potential impact and areas for future development.

CHAPTER 7: PROJECT CONCLUSION

7.1 Observation on Weaknesses and Strengths

The FloraHub system stands out for its robust design and user-centric approach. One of its key strengths is the integration of a smart hydration system that ensures plants receive optimal moisture levels, which is essential for their health and growth. The use of advanced technologies like IoT with MicroPython and secure backend infrastructure developed with Spring Boot and MySQL enables precise soil moisture monitoring and timely watering. Additionally, the system's mobile application, built with Flutter, offers a user-friendly interface that simplifies plant care management. The inclusion of features such as real-time notifications via OneSignal enhances user engagement by providing timely updates and alerts. The comprehensive testing strategy, including event-based and online testing phases, as well as regular supervisor checks and expert presentations, ensures that the system is continually refined and improved based on diverse feedback.

Despite its strengths, the FloraHub system faces some challenges. One potential weakness is its reliance on the stability and performance of the underlying technologies and platforms. Issues such as system latency or integration problems with IoT devices could affect performance and user experience. Additionally, while the mobile application provides a user-friendly interface, there may be a learning curve for users who are less familiar with technology, which could impact overall usability. The system's performance and reliability are contingent on efficient resource management, and any inefficiencies could lead to increased battery consumption and reduced device performance. Furthermore, although comprehensive testing is conducted, there may still be unforeseen issues or edge cases that only emerge in broader, real-world use, potentially requiring further iterations and adjustments. Addressing these weaknesses will be crucial to enhancing the system's overall robustness and user satisfaction.

Based on the feedback collected from Google Forms, the FloraHub system exhibits several notable strengths and areas for improvement. Users consistently highlight the system's convenience, particularly for individuals with busy lifestyles, as it effectively simplifies plant care through automated watering schedules and accurate monitoring of soil moisture and pH levels. The user-friendly and intuitive interface is praised for its ease of navigation and aesthetic design, enhancing the overall user experience. However, some feedback points to the need for improvements, such as enhancing the user experience further and providing more detailed information about flora. Suggestions for additional features, like automatic fertilizer integration, indicate a desire for further functionality. Addressing these areas could refine the app's effectiveness and broaden its appeal.

7.2 Propositions for Improvement

To address the feedback on improving user experience, it's essential to focus on refining the app's interface and overall usability. Conducting a detailed user experience (UX) analysis can identify pain points and areas where users may face challenges. Incorporating user feedback through iterative design improvements, such as streamlining navigation paths, enhancing visual clarity, and simplifying interactions, can make the app more intuitive. Additionally, implementing user tutorials or help guides can assist new users in understanding and maximizing the app's features. Regular updates based on user feedback will ensure that the app evolves to meet user expectations and maintains a high level of satisfaction.

To enhance the app's value for users, integrating more comprehensive information about different types of plants can be beneficial. This could include detailed care instructions, growth stages, and specific needs for various plant species. By expanding the database of plant information, users can better understand how to care for their plants and make informed decisions about their care routines. Incorporating educational content, such as tips on pest management and fertilization, can further enrich the user experience. This addition will not only make the app more informative but also position it as a valuable resource for plant enthusiasts seeking to improve their gardening knowledge.

In response to requests for additional features, integrating an automatic fertilizer system could significantly enhance the functionality of FloraHub. This feature would allow users to set fertilization schedules based on the specific needs of their plants, ensuring that plants receive the right nutrients at the right times. By incorporating sensors or manual input for nutrient levels, the system could provide tailored recommendations for fertilization. This addition would make plant care even more convenient and effective, addressing a key area of interest and potentially increasing user engagement and satisfaction.

To improve the app's appeal and functionality, expanding the user interface design to include more customizable options and additional features is recommended. This could involve enhancing the visual design with more themes or color schemes, allowing users to personalize their experience according to their preferences. Additionally, incorporating more interactive elements, such as graphs or visualizations of soil moisture and pH levels, can provide users with clearer insights into their plant's health. An expanded UI can also include features like push notifications for plant care reminders and personalized tips, further improving user engagement and overall satisfaction.

7.3 Project Contribution

The FloraHub project represents a significant advancement in IoT-based plant care systems by delivering an innovative solution that integrates technology with environmental sustainability. The project demonstrates how advanced technologies, such as IoT and mobile app development, can be effectively combined to address real-world challenges in plant care. By providing an automated hydration system, FloraHub offers a practical tool for individuals seeking to simplify plant maintenance, particularly those with busy schedules. The project showcases the potential of smart technology in enhancing everyday tasks and serves as a model for future innovations in automated gardening solutions.

7.4 Conclusion

The FloraHub project successfully met its set objectives, offering a reliable and user-friendly automated plant hydration system. Through continuous testing, refinement, and user feedback, FloraHub has evolved into a robust solution that simplifies plant care for users, especially those with busy lifestyles. The system's strengths, such as its smart hydration features, real-time notifications, and user-friendly design, were well received, with room for future enhancements like fertilization and expanded plant information. In conclusion, FloraHub provides an innovative and convenient solution for modern plant care, contributing positively to the field and laying the foundation for future improvements.

REFERENCES

Manaha, R. R. Y. (2019, January 1). *Automatic watering system for plants with IOT monitoring and notification*. CogITo Smart Journal. https://www.academia.edu/85394780/Automatic_Watering_System_for_Plants_with_IoT_Monitoring_and_Notification?uc-g-sw=94076074

S. Bhardwaj, S. Dhir and M. Hooda, "Automatic Plant Watering System using IoT," 2018 Second International Conference on Green Computing and Internet of Things (ICGCIoT), Bangalore, India, 2018, pp. 659-663, doi: 10.1109/ICGCIoT.2018.8753100. https://ieeexplore.ieee.org/document/8753100/citations#citations

S. D, S. J. P. C, K. I, S. K. S and G. J. W. Katherine, "Smart Pot Using Internet of Things for Plant Hydration," 2023 2nd International Conference on Edge Computing and Applications (ICECAA), Namakkal, India, 2023, pp. 1307-1310, doi: 10.1109/ICECAA58104.2023.10212119. https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=10212119&isnumber=10212098

L. P. Dewi, J. Andjarwirawan and R. P. Wardojo, "Android Application for Monitoring Soil Moisture Using Raspberry Pi," 2017 International Conference on Soft Computing, Intelligent System and Information Technology (ICSIIT), Denpasar, Indonesia, 2017, pp. 178-184, doi: 10.1109/ICSIIT.2017.63. https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8262564&isnumber=8262523

Z. Muhammad, M. A. A. M. Hafez, N. A. M. Leh, Z. M. Yusoff and S. A. Hamid, "Smart Agriculture Using Internet of Things with Raspberry Pi," 2020 10th IEEE International Conference on Control System, Computing and Engineering (ICCSCE), Penang, Malaysia, 2020, pp. 85-90, doi: 10.1109/ICCSCE50387.2020.9204927.

https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9204927&isnumber=9204918

Kumar, M. S., Chandra, T. R., Kumar, D. P., & Manikandan, M. S. (n.d.-a). *Monitoring moisture of soil using low cost homemade soil moisture sensor and Arduino Uno: IEEE conference publication: IEEE Xplore*. ieeexplore.ieee.org. https://ieeexplore.ieee.org/document/7586312

R. Dagar, S. Som and S. K. Khatri, "Smart Farming – IoT in Agriculture," 2018 International Conference on Inventive Research in Computing Applications (ICIRCA), Coimbatore, India, 2018, pp. 1052-1056, doi: 10.1109/ICIRCA.2018.8597264. https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8597264&isnumber=8596764

C. Mageshkumar and K. R. Sugunamuki, "IOT Based Smart Farming," 2020 International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, India, 2020, pp. 1-6, doi: 10.1109/ICCCI48352.2020.9104103. https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9104103&isnumber=9104045

APPENDICES

1. YouTube link for the demonstration of the FloraHub mobile application: https://youtu.be/grQzcS69L74?si=eyxgqZMNA6Kv4cow

2. YouTube link for the demonstration of the FloraHub prototype:

   https://youtu.be/HdVnUw3bS9o?si=UMzeBJKllYoDzgsC

3. Link of the questionnaire for the Projek Sarjana Muda 1:

   https://docs.google.com/forms/d/e/1FAIpQLSfPvuftXeojDEVisj3GfiQVN5PrekMa1WKqNOHSG9Ri36wcQQ/viewform?usp=sf_link

4. Link of the questionnaire for the Projek Sarjana Muda 2:

   https://docs.google.com/forms/d/e/1FAIpQLSeASiRwPTuHkDcch9ZeUdYCF54BrLOGjhdtAJjY9RYxJLX8Yg/viewform?usp=sf_link

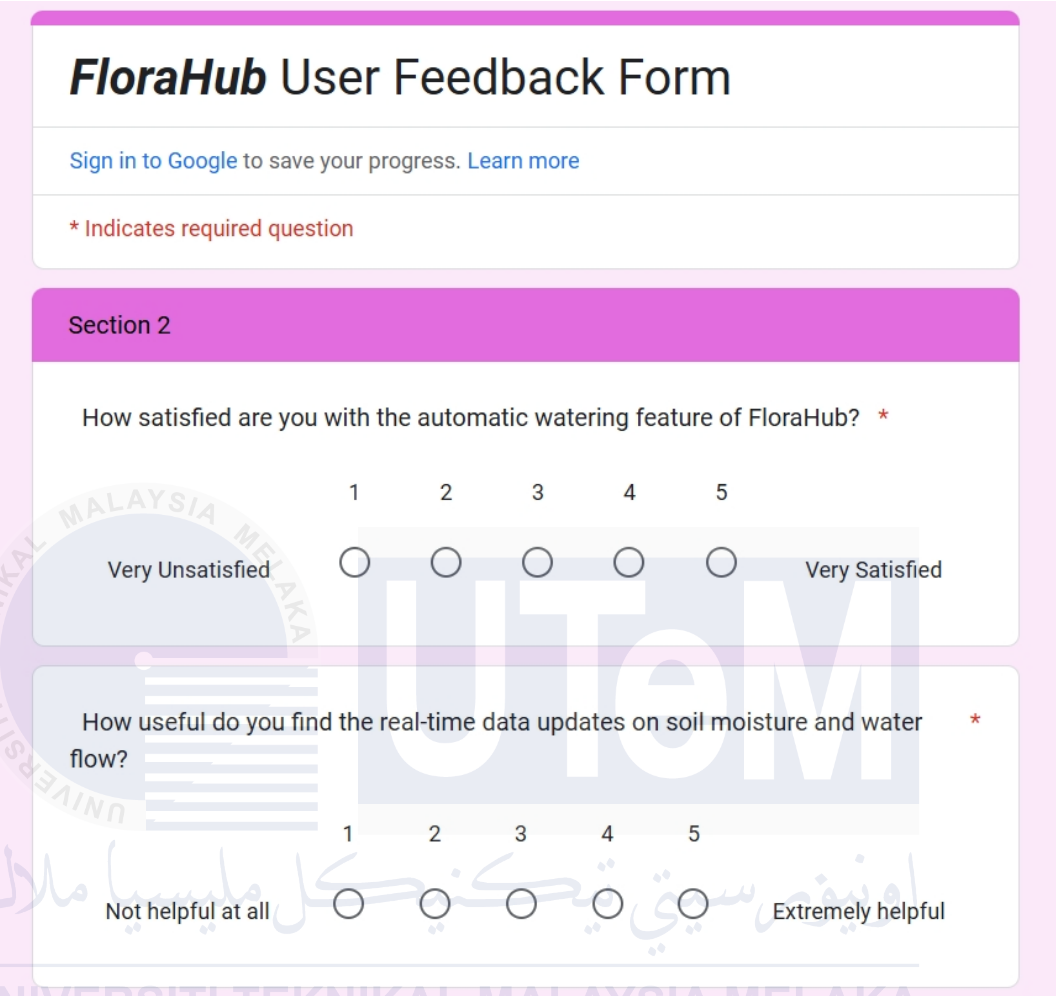
5. Summary of the questionnaire 1:

No.	Question	Rate				
		1	**2**	**3**	**4**	**5**
1	How satisfied are you with the automatic watering feature of FloraHub?	0	1	0	3	5
2	How useful do you find the real-time data updates on soil moisture and water flow?	0	0	1	2	6
3	How helpful do you find the dashboard analysis (daily, monthly, yearly reports) provided by FloraHub?	0	0	1	3	5
4	How user-friendly do you find the FloraHub mobile application?	0	0	1	3	5
5	How would you rate your overall experience with the FloraHub system?	0	0	1	2	6
6	How likely are you to recommend FloraHub to other plant enthusiasts?	0	0	1	1	7

6. Summary of the questionnaire 2:

No.	Question	Rate				
		1	**2**	**3**	**4**	**5**
1	How user-friendly do you find the FloraHub app interface?	1	1	7	8	18
2	How effective is the manual watering method for you?	0	1	8	10	16
3	How convenient do you find the scheduled watering method?	0	2	3	8	22
4	How would you rate the overall performance of the FloraHub system?	1	2	2	6	24
5	How would you rate the overall design and aesthetics of the FloraHub app?	0	1	2	8	24

7. Screenshots of the questionnaire for the Projek Sarjana Muda 1:



FloraHub User Feedback Form

Sign in to Google to save your progress. Learn more

* Indicates required question

Section 2

How satisfied are you with the automatic watering feature of FloraHub? *

	1	2	3	4	5	
Very Unsatisfied	○	○	○	○	○	Very Satisfied

How useful do you find the real-time data updates on soil moisture and water flow? *

	1	2	3	4	5	
Not helpful at all	○	○	○	○	○	Extremely helpful

How helpful do you find the dashboard analysis (daily, monthly, yearly reports) provided by FloraHub? *

	1	2	3	4	5	
Not helpful at all	○	○	○	○	○	Extremely helpful

How user-friendly do you find the FloraHub mobile application? *

	1	2	3	4	5	
Not user-friendly at all	○	○	○	○	○	Extremely user-friendly

What features would you like to see added to the FloraHub system in the future? *

Your answer

Section 3

How would you rate your overall experience with the FloraHub system? *

1	2	3	4	5
○	○	○	○	○

How likely are you to recommend FloraHub to other plant enthusiasts? *

	1	2	3	4	5	
Very unlikely	○	○	○	○	○	Very likely

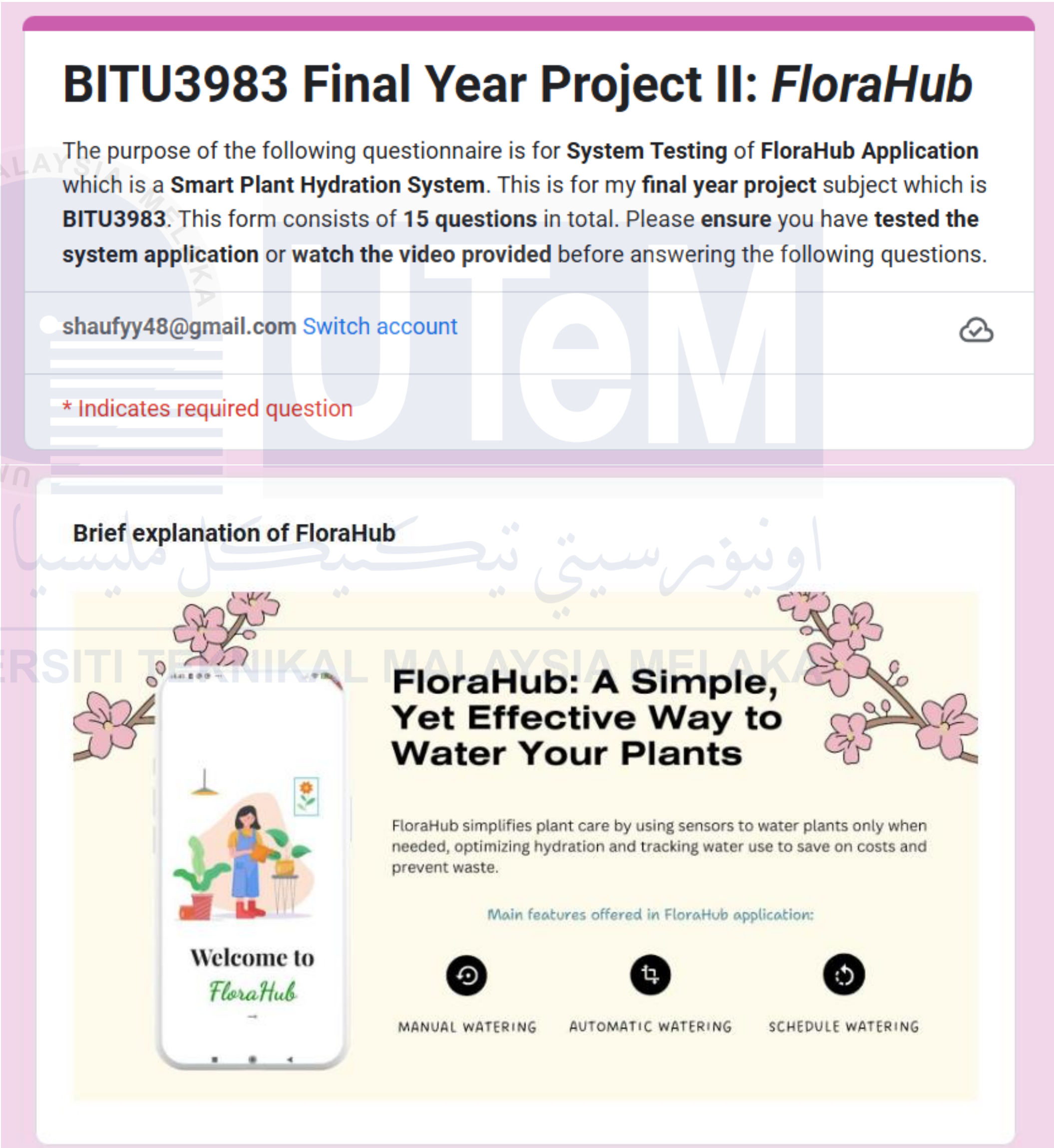
Any additional comments or feedback? *

Your answer


Thank you for your response!

Back Submit Clear form

8. Screenshots of the questionnaire for the Projek Sarjana Muda 1:


BITU3983 Final Year Project II: *FloraHub*

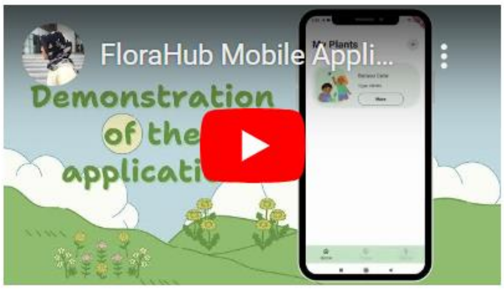
The purpose of the following questionnaire is for **System Testing** of **FloraHub Application** which is a **Smart Plant Hydration System**. This is for my **final year project** subject which is **BITU3983**. This form consists of **15 questions** in total. Please **ensure** you have **tested the system application** or **watch the video provided** before answering the following questions.

shaufyy48@gmail.com Switch account

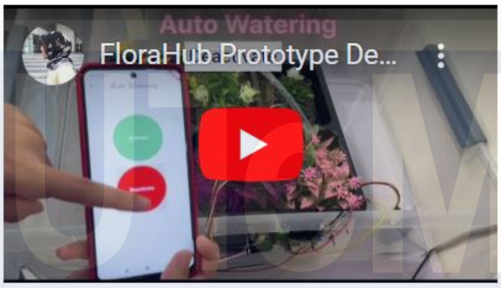
* Indicates required question

Brief explanation of FloraHub

Mobile Application Demonstration



Prototype Demonstration



Link to install the application ->
If you wish to experience the application firsthand, you can download through this link

FloraHub: Smart Plant Hydration System. Please note that this application is **available** for **Android only**.

Section 1: Respondent Background

Are you from University Technical Malaysia Melaka (UTeM) ? *

- ◯ Yes
- ◯ No
- ◯ Other: ________

How old are you? *

- below 20 years old
- 20 - 25 years old
- 25 - 30 years old
- 30 years old and above

What is your occupation *

- Student
- Employed
- Self-employed
- Unemployed
- Retired
- Other:

Do you have experience with gardening or plant care? *

- Yes
- No

How many plants do you currently take care of? *

- 0-5
- 6-10
- 11-15
- More than 15

What type of plants do you mostly care for? *

- ◯ Indoor plants
- ◯ Outdoor plants
- ◯ Both

Section 2: Experience with FloraHub

Before you answer the question below, please refer to the YouTube video demonstration provided or installation link that has been given.

How user-friendly do you find the FloraHub app interface? *

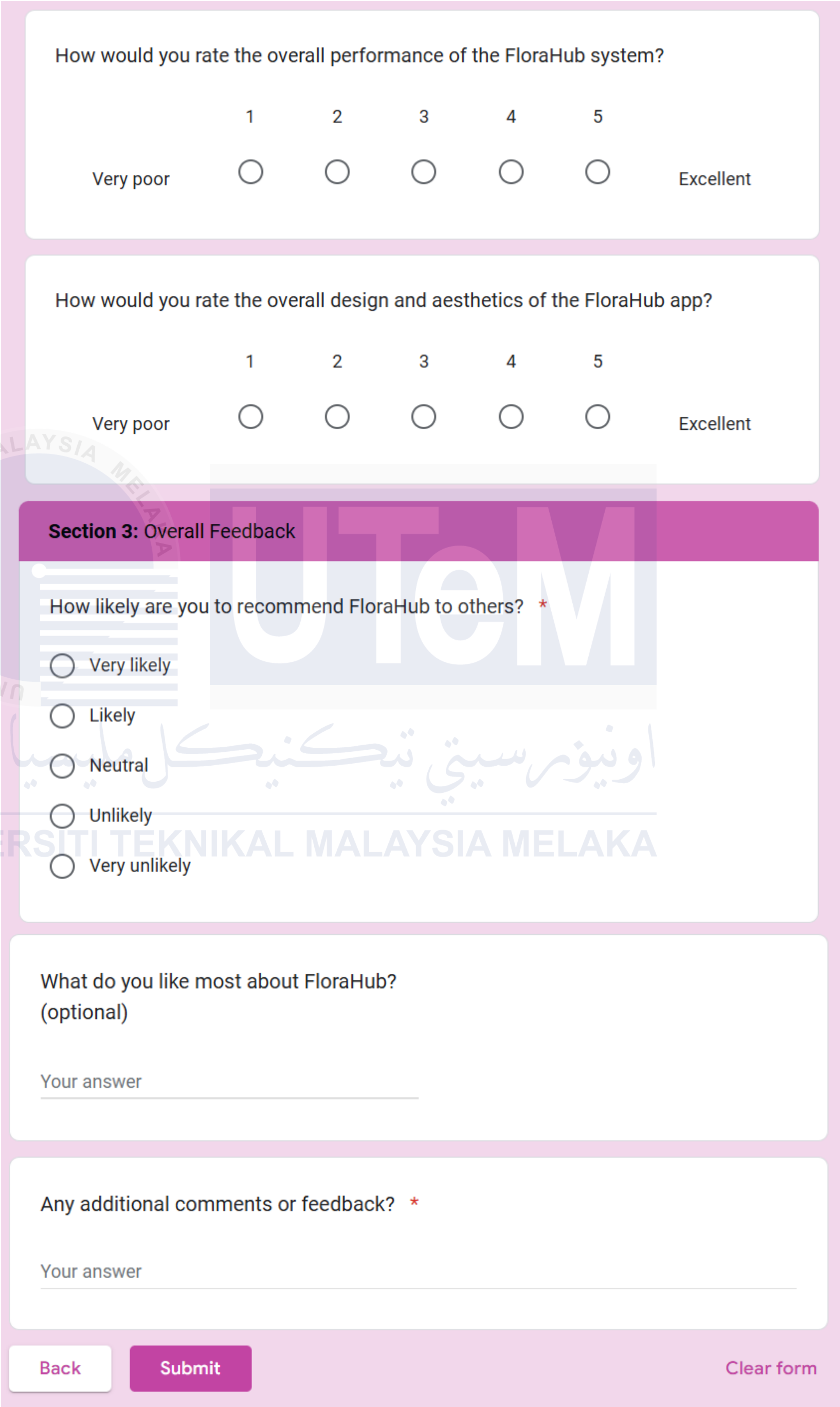
	1	2	3	4	5	
Very difficult to use	◯	◯	◯	◯	◯	Very user-friendly

How effective is the manual watering method for you? *

	1	2	3	4	5	
Very ineffective	◯	◯	◯	◯	◯	Very effective

How convenient do you find the scheduled watering method? *

	1	2	3	4	5	
Very inconvenient	◯	◯	◯	◯	◯	Very convenient



How would you rate the overall performance of the FloraHub system?

	1	2	3	4	5	
Very poor	○	○	○	○	○	Excellent

How would you rate the overall design and aesthetics of the FloraHub app?

	1	2	3	4	5	
Very poor	○	○	○	○	○	Excellent

Section 3: Overall Feedback

How likely are you to recommend FloraHub to others? *

- ○ Very likely
- ○ Likely
- ○ Neutral
- ○ Unlikely
- ○ Very unlikely

What do you like most about FloraHub? (optional)

Your answer

Any additional comments or feedback? *

Your answer

Back | Submit | Clear form