HOUSEHOLD PLANT MONITORING SYSTEM

SUUGEN RAJ A/L VENUGOPAL

BACHELOR OF COMPUTER SCIENCE

(COMPUTER SYSTEM & NETWORKING)

UNIVERSITI MALAYSIA PAHANG

UNIVERSITI MALAYSIA PAHANG

DECLARATION OF THESIS AND COPYRIGHT

Author's Full Name	: SUUGEN RAJ A/L VENUGOPAL
Date of Birth	
Title	: HOUSEHOLD PLANT MONITORING SYSTEM
Academic Session	: SEMESTER 1 2022/2023

I declare that this thesis is classified as:

□	CONFIDENTIAL	(Contains confidential information under the Official Secret Act 1997)*
□	RESTRICTED	(Contains restricted information as specified by the organization where research was done)*
☑	OPEN ACCESS	I agree that my thesis to be published as online open access (Full Text)

I acknowledge that Universiti Malaysia Pahang reserves the following rights:

1. The Thesis is the Property of Universiti Malaysia Pahang
2. The Library of Universiti Malaysia Pahang has the right to make copies of the thesis for the purpose of research only.
3. The Library has the right to make copies of the thesis for academic exchange.

Certified by:

(Student's Signature)	(Supervisor's Signature)
New IC/Passport Number Date: 8 FEBRUARY 2023	DR.NOORHUZAIMI@KARIMA H BINTI MOHD NOOR Date: 8 FEBRUARY 2023

NOTE : * If the thesis is CONFIDENTIAL or RESTRICTED, please attach a thesis declaration letter.

SUPERVISOR'S DECLARATION

I hereby declare that I have checked this thesis and, in my opinion, this thesis is adequate in terms of scope and quality for the award of the degree of Bachelor of Computer Science (Computer Systems & Networking).

(Supervisor's Signature)

Full Name : DR. NOORHUZAIMI@KARIMAH BINTI MOHD NOOR

Position : Senior Lecturer

Date : 8 FEBRUARY 2023

Universiti
Malaysia
PAHANG
Engineering • Technology • Creativity

STUDENT'S DECLARATION

I hereby declare that the work in this thesis is based on my original work except for quotations and citations which have been duly acknowledged. I also declare that it has not been previously or concurrently submitted for any other degree at Universiti Malaysia Pahang or any other institutions.

Suugen Raj

(Student's Signature)

Full Name : SUUGEN RAJ A/L VENUGOPAL

ID Number : CA19069

Date : 8 FEBRUARY 2023

ACKNOWLEDGEMENTS

First of all, I would like to express my gratitude to DR. NOORHUZAIMI@KARIMAH BINTI MOHD NOOR, my supervisor for PSM 1 and 2, for her constant support and guidance. Always assisted me by suggesting new ideas and giving moral support. Following that, I would like to express my gratitude to my parents for their unwavering support in my life as I finish this project. Finally, I would like to express my gratitude to classmates and friends who was always there and helped me physically and mentally to finish this project on time.

ABSTRAK

Hampir setiap orang hari ini mempunyai tumbuhan di rumah mereka. Tumbuhan ini, juga dikenali sebagai tumbuhan rumah, perlu disiram setiap hari kerana air adalah bahagian penting dalam pertumbuhannya. *'Household Plant Monitoring System'* (HPMS) boleh mengumpul data kelembapan tanah, suhu dan kelembapan serta paparan pada aplikasi web. Pada masa yang sama, data akan disimpan dalam pangkalan data. Sistem ini beroperasi secara automatik untuk proses pengairan. Pengguna akan dihantar pemberitahuan melalui e-mel tentang status pengairan. Masalah yang membawa kepada pembangunan projek ini adalah kekurangan pemantauan tumbuhan oleh pengguna, kebanyakan sistem pertanian yang telah dibangunkan hanya boleh memantau dan mengumpul data untuk satu jenis tumbuhan dan akhir sekali, menyiram tanaman secara manual adalah tidak sistematik kerana air yang berlebihan mungkin dibekalkan. Matlamat projek ini adalah untuk membangunkan *'Household Plant Monitoring System'* (HPMS) untuk tiga tumbuhan berbeza. Beberapa objektif telah dikenal pasti untuk mencapai matlamat tersebut. Pertama, reka bentuk HPMS untuk memantau dan menyiram sendiri tumbuhan. Kedua, membangunkan HPMS bersepadu dengan aplikasi web untuk memantau tumbuhan. Ketiga, mengesahkan HPMS yang telah dibangunkan dengan menggunakan kaedah ketepatan kadar yang betul berdasarkan data yang sesuai untuk setiap tumbuhan. Terdapat empat skop yang telah dikenal pasti untuk projek ini. Jenis tumbuhan yang digunakan ialah tembikai, cili dan tomato. Teknologi yang digunakan ialah ESP32, sensor kelembapan tanah, sensor DHT11 (suhu & kelembapan), modul geganti, pam air, penukar voltan injak turun dan wayar pelompat. Pengguna adalah pengguna berkebun di rumah dan data yang sesuai terdiri daripada kelembapan tanah, suhu, kelembapan dan jenis tanah. Model SDLC yang sesuai dipilih untuk projek ini ialah model tambahan. Sambungan komponen dilakukan pada kenaikan pertama, diikuti oleh pangkalan data yang ditetapkan dalam kenaikan kedua dan pembangunan aplikasi web pada kenaikan ketiga. Selepas menyiapkan projek, sistem HPMS telah diuji selama lima hari. Sistem berjaya mendapatkan dan menyimpan data ke dalam pangkalan data, memaparkan bacaan sensor pada halaman web dan mengairi secara automatik mengikut syarat yang ditetapkan. Bacaan yang dikumpul untuk setiap tumbuhan kemudiannya digunakan untuk mengira kecekapan sistem menggunakan formula kadar yang betul.

Kesimpulannya, ketiga-tiga objektif telah dicapai kerana HPMS telah berjaya direka bentuk, dibangunkan dan disahkan.

ABSTRACT

Almost everyone today has plants in their houses. These plants, also known as household plants, need to be watered everyday as water is an essential part of their growth. Household Plant Monitoring System (HPMS) can gather data of soil moisture, temperature and humidity and display on the web application. At the same time, the data gets stored in the database. The system operates automatically for the irrigation process. User will be sent notifications via email about the irrigation status. Problems leading to the development of this project were lack of plant monitoring by users, most of the agriculture systems that have been developed can only monitor and collect data for one type of plant and lastly, watering the plants manually is not systematic as excess water might be supplied. The aim of this project is to develop a Household Plant Monitoring System (HPMS) for 3 different plants. Several objectives have been identified to achieve the aim. Firstly, design a HPMS to monitor and self-water the plant. Secondly, develop the HPMS integrated with web application to monitor the plants. Thirdly, validate the HPMS that has been developed by using the correct rate accuracy method based on the suitable data for each plant. There are four scopes that have been identified for this project. Plant types used are watermelon, chili and tomato. Technology used are ESP32, soil moisture sensor, DHT11 sensor (temperature & humidity), relay module, water pump, step-down voltage converter and jumper wires. Users are home gardening users and the suitable data consists of soil moisture, temperature, humidity and soil type. Suitable SDLC model chosen for this project is incremental model. Component connection was done in the first increment, following by database set up in the second increment and web application development in the third increment. After completing the project, HPMS system was tested for five days. The system successfully retrieved and stored data into database, displayed sensor readings on the webpage and irrigated automatically according to the conditions set. The readings gathered for each plant were then used to calculate the efficiency of the system using the correct rate formula. In conclusion, all three objectives have been achieved as the HPMS was successfully designed, developed and validated.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 BACKGROUND

Agriculture continues to be a major aspect of Malaysia's economy, producing 12% of the country's GDP and employing 16% of the population. Large-scale plantations were established, and new economic crops were introduced by the British (rubber in 1876, palm oil in 1917, and cocoa in the 1950s). Since then, the three primary crops which are rubber, palm oil, and cocoa have dominated agricultural exports, despite Malaysia's share of global supply of these crops continuously declining during the last two decades. Farmers in Malaysia also produce a variety of fruits and vegetables for the market, including bananas, coconuts, durian, pineapples, rice, rambutan, and other fruits and vegetables. The tropical weather of Malaysia is ideal for growing a variety of exotic fruits and vegetables, especially since hurricanes and droughts are rare in Peninsular Malaysia.

Covid-19 has presented issues to all economic sectors in 2019, particularly agriculture. The Movement Control Order (MCO) in Malaysia sparked some panic buying at first. Cooking at home has become more popular as people have stayed indoors. Fresh food items are in high demand at local markets and supermarkets as a result of this. At the same time, the quarantine imposed to limit the coronavirus outbreak has harmed labour supply and disrupted agricultural supply networks. Furthermore, the agriculture sector in Malaysia is significantly reliant on foreign labour.

The government has launched a number of agricultural-related projects. Lending facilities and capital injections for infrastructure development and short-term agriculture projects are among these measures. Apart from agriculture, the government also helps small and medium businesses (SMEs) and low-income earners, which benefits farmers, fishers, cattle, and agriculture employees [1]. Agriculture should be protected at any cost as it is the most important biological asset. It is the foundation for all food production and the means to feeding an enormous number of people in times of climate and environmental change.

1.2 PROBLEM STATEMENT

The first problem that led to the development of this project is lack of plant monitoring by users [2]. Monitoring and manually watering the plants makes it difficult for users when they are away from home. Secondly, most of the agriculture system developed can only monitor and collect data for one type of plant [3]. Different type of plants which have different requirements cannot be monitored using the same system. Thirdly, watering the plants manually is not systematic as excess water might be supplied [3]. There would not be enough air pockets if the soil moisture is too high. As a result, plants are unable to breathe due to a lack of oxygen.

1.3 OBJECTIVES

The aim of this project is to develop a Household Plant Monitoring System for 3 different plants. In order to achieve the aim, several objectives have been identified. The objectives are:

i. Design a Household Plant Monitoring System to monitor the plant and self-watering.
ii. Develop the Household Plant Monitoring System integrated with web application to monitor the plants.
iii. Validate the Household Plant Monitoring System that has been developed by using the correct rate accuracy method based on the suitable data for each plant.

1.4 SCOPE OF PROJECT

There are three objectives that have been identified for this project. In order to ensure all the objectives can be achieved, there are four scopes that have been identified. The scopes are:

i. Plant types: Watermelon, chili, tomato.

ii. Technology used: ESP32, soil moisture sensor, DHT11 Sensor (Temperature & humidity), relay module, water pump and jumper wires.

iii. User: Home gardening users.

iv. Suitable data consists of soil moisture, temperature, humidity and soil type.

1.5 REPORT ORGANIZATION

Chapter 1 explains the introduction of agriculture in Malaysia. This chapter also includes information about the project's problem statement. Finally, the project’s objective and scope have been outlined explicitly.

In chapter 2, the literature review of the project will be stated clearly. Similar existing projects will be discussed furtherly in this chapter which includes aim of the project, scope, method or technology, software and hardware used, results and understanding about the projects.

Chapter 3 explains in detail about the methodology used to develop this project. It explains clearly with the content such as proposed design and interface including the hardware and software specification.

In chapter 4, the implementation, testing, and outcome are all discussed in detail. This chapter went over every detail of the project's development in significant detail.

Chapter 5 is the conclusion. All the limitations occurred during the finishing of this thesis are stated in this chapter. Besides, future works that can be done are also stated.

CHAPTER 2

LITERATURE REVIEW

2.1 INTRODUCTION

In this chapter, the discussion on the work analysis based on the reviewed literature from ten sets of student’s final year projects and journals with the similar perspective based on the system are compared. The aim, scope, method, technology, result of projects and understanding are compared to gain more detailed knowledge regarding the projects.

2.2 AGRICULTURE

Agriculture is the method and science of nurturing the soil, growing crops, and livestock breeding. It comprises both the preparation and distribution of plant and animal products for human use. Most of the world's food and textiles are produced by agriculture. Out of many types of agriculture, the most commonly known are small-scale agriculture and industrial agriculture. Small-scale agriculture is a type of farming that relies more on human labour than on machinery [4]. In industrial agriculture, farmers seldom produce more than two or three different crops, and do virtually everything use heavy machinery. Household agriculture is also referred as small-scale agriculture and the type of plants used are spices, herbs, decorative and medicinal plants, vegetables, fruits and plantation crops [5]. For this project, three types of plants have been chosen which are tomato, chili and watermelon.

2.2.1 Household Plants Requirement

Temperature and humidity are significant elements because environmental factors have an impact on plant growth stages. Soil moisture is also vital because water in the soil acts as a solvent, breaking down the nutrients and minerals that plants require from the soil and allowing them to be absorbed into their systems [6]. Different plants have different requirements of surrounding temperature, humidity and soil moisture for optimum growth.

The suitable soil moisture for tomato plant is between 70%-80% [7]. For optimum tomato plant growth, the suitable temperature is 18°C - 32°C and 50%-70% humidity [8]. Tomato plants are best grown in sandy loam soil; belongs to *Solanaceae* plant family.

Suitable soil moisture for watermelon plant is 70%-80%. Temperature should range between 18 °C - 35 °C during growing season [9]. Besides that, optimum relative humidity is 60%-80%, sandy loam soil the best choice and lastly watermelon belongs to *Cucurbitaceae* plant family.

The ideal soil moisture for chili plants to grow well is 50%-70% [10]. In addition, the suitable temperature for the plant in Malaysia is 20 °C - 30 °C [11]. Humidity about 50%-70%, plant grown in sandy loam soil can help chili grow better. Chili belongs to *Solanaceae* plant family. Table 2.1 shows the summarization of household plants requirement.

Table 2.1 Household plants requirement based on three types of plants

Plant	Soil Moisture	Temperature	Humidity	Soil	Plant Family
Tomato plant	70%-80%	18°C - 32°C	50%-70%	Sandy loam soil	Solanaceae
Watermelon Plant	70%-80%	18 °C - 35 °C	60%-80%	Sandy loam soil	Cucurbitaceae
Chili Plant	50%-70%	20 °C - 30 °C	50%-70%	Sandy loam soil	Solanaceae

2.3 RELATED WORK

There are many similar projects that has been done by previous researchers. In this proposal, ten projects will be discussed; Smart Irrigation System [3], Remote Watering System for Home Garden [12], Iot Based Smart Agriculture [13], Smart Irrigation System [14], IoT based SMART FARMING SYSTEM[15], Internet of Things for Indoor Farming [16], IOT BASED AUTOMATIC IRRIGATION SYSTEM [17], A Plant monitoring and smart gardening system using IoT [18], Automated Plant Watering System [19] and An Internet of Things Based Smart Irrigation using Solenoid Valve [20].

‘Smart Irrigation System’ [3], which aims to make the farmer's life easier by allowing him to monitor the soil moisture and ensure that the plant's water intake is sufficient for growth. The scopes of the project are, farmers can monitor data from the system on the dashboard to verify that plant water consumption runs fine, user can view data from the dashboard using a web-based system and lastly, database for analyzing and presenting collected data and information in dashboard. This system uses Arduino Uno R3 with soil moisture sensor and water pump. The ARDUINO UNO Board was programmed using the Arduino program. The sensor will need to be calibrated as well, and it will be integrated into the ARDUINO Board.

‘Remote Watering System for Home Garden’ [12], aims to develop a system that allows the user to monitor their home garden while they are away as part of a home automation system. The scope is to develop a system that allows users to monitor their plants based on the current soil humidity levels and control the irrigation systems remotely via a web server. Arduino, soil humidity sensor, and a water pump were used to develop this system. Because of the lack of Wi-Fi in the development region, the prototype was created using an Ethernet connection rather than Wi-Fi.

‘Iot Based Smart Agriculture’ [13], aims to reduce manpower and issues in crops and other agriculture farms. The scope is smart data transfer through the cloud, smart irrigation with remote monitoring based on real-time data, and smart warehouse management, that includes temperature, humidity, soil moisture, and pH maintenance. The project makes use of a Node MCU and a wireless module that can connect to the internet. Using the cloud network Thingspeak, NodeMCU is initialized and synced with various sensors, allowing it to operate as a micro system to control the farm with the help of any smart smartphone or tool.

‘Smart Irrigation System’ aims to design, develop, and deploy a simple self-watering system that eliminates the need for users to monitor and care for their plants [14]. The scope is to reduce water waste by implementing a system that does not require any user input while watering the plants and veggies. Moisture sensors, solenoid valves, relay modules, GrovePi, and RaspberryPi will be the system's main components. The core functionality of the code was built in Python on the RaspberryPi, and the android mobile application was created in Java.

‘IoT based SMART FARMING SYSTEM’ [15], aim is farmers can use an IoT-based Smart Farming System to acquire real-time data (Temperature, Soil Moisture) for effective monitoring, allowing them to boost overall crop and product quality. The scope is to develop a system that allows farmers to get real-time data on soil moisture and temperature at a minimal cost, allowing for real-time monitoring. The system combines Arduino technology with several sensors and a Wi-Fi module to produce a real-time data broadcast that can be accessed at Thingsspeak.com.

‘Internet of Things for Indoor Farming’ [16], which aims indoor farming environments are created to provide a sustainable atmosphere in which plants or crops can thrive. The scope is to create a real-time monitoring system with a user-friendly interface and a remotely controlled system. Real-time monitoring of crops and farming environment parameters such as temperature, humidity, and soil moisture were given by the system. Furthermore, using Freeboard.io's local web server and dashboard service, the system created a simple user interface for users to review the data.

‘IOT BASED AUTOMATIC IRRIGATION SYSTEM’ [17], aims to establish a system that can measure and monitor temperature, light, humidity, and moisture levels in the greenhouse automatically. The scope is the system can efficiently monitor a range of environmental conditions in a greenhouse and fulfil the needs of actual agricultural production. Irrigation will occur if the temperature and humidity of the surrounding environment change. A solenoid valve controls the water flow. When a signal is sent via microcontroller, the valve opens and closes. The water was supplied drop by drop, and when the moisture level returns to normal, the sensor detects it and sends a signal to the microcontroller, which closes the valve.

‘A Plant monitoring and smart gardening system using IoT’ [18], aims to lower the cost of implementation in comparison to another system. The scope is to make people more convenient by minimizing manual labor and improving the overall functionality of any system without requiring human participation. Sensors such as a humidity sensor, moisture sensor, LDR, and temperature sensor are connected to the Raspberry Pi board to control temperature, humidity, moisture, and light intensity. Using IoT, this information about the garden may be monitored and controlled directly from a mobile phone. It then uses the database to upload the data to the cloud. Through the application, the user can manually regulate the water valve.

‘Automated Plant Watering System’ [19] project which aims to implement a simple automatic irrigation system to water a small potted plant or crop with minimal human interaction. The scope is, an automated system that will look after plants while taking into account all of the varied features of a home gardening system as well as the bigger landscape and assisting them in growing healthily. The Arduino IDE software is used to programme the system. If the soil moisture level is low, the Arduino microcontroller activates a water pump until the sensor reaches the limit. Following that, the system will check the soil moisture at regular intervals to see whether additional water is required or not. If there is no water or a comment during the initial inspection, the system waits 24 hours and repeats the process.

‘An Internet of Things Based Smart Irrigation using Solenoid Valve’ project [20], aims to design a simple, easy-to-install system for monitoring and indicating soil moisture levels that is organised and automated on a regular basis using GSM and GPRS technologies. The scopes are, low-cost sensors measure soil moisture, soil type, and meteorological variables in this system, the information acquired is then directly saved on the web page and could be viewed on a smart phone or on a computer and lastly, the data from the field is sent to the cloud via cellular networks that are part of the global system for mobile communication (GSM). The soil sensors implanted in the soil send data to the Pic-Microcontroller, which analyses the data and sends a signal to the solenoid valve, turning on the water pump based on the sensor values considered; similarly, the humidity and temperature sensors sense their levels and display them in the LCD.

Table 2.2 shows the summary of comparison of related work that have been discussed in the sub chapter. The comparison has been made based on watering method, live monitoring, storing data and sensors used. These features are compared because the related features will be referred to develop the proposed project, Household Plant Monitoring System.

Table 2.2 Comparison of related work

Author	**Watering Method**	**Live monitoring**	**Storing data**	**Sensors used**
Ariff Bin Azlan	Auto	✓	✓	Arduino, NodeMCU-esp12e, soil moisture sensor, solenoid, single relay module.
Siti Aisyah Binti Zainal Abidin	Auto & Manual	✓	✗	Arduino Uno, Soil moisture sensor, Immersible Pump, Arduino Ethernet Shield, Relay.
Manoj, Subba Rao, Vamsi, Vasantha	Auto	✓	✓	NodeMCU, DHT11, Moisture sensor, pH Sensor Module.
Sam Conneely	Auto	✓	✓	Raspberry Pi, GrovePi Kit, Ultrasonic Sensor, Moisture Sensor, Temperature & Humidity Sensor, Relay Module, Solenoid Valves.
Yasir Fahim, Tania Srakar	_	✓	✓	Arduino UNO, Wi-Fi Module ESP-8266, Temperature sensor-DS18B20, Soil moisture sensor-FC28.
Chong Kai Chian	Auto	✓	✓	Raspberry Pi 2, DHT11 sensor, 2 DC Fan, Camera (Logitech Webcam C270), Wi-Fi dongle, Flood light, FC-28 Soil Moisture Sensor, 4-Channel relay, Water pump pipe.
Satao, Chavan, Kale, Patil, Chaugule	Auto	✓	✓	Microcontroller (ATMEGA 16), Temperature sensor (LM35), Moisture sensor, 16 *2 LCD display, ESP8266 module, Relay and Motor.
Ramkumar, Nagarani, Rozario, Banu	Auto & Manual	✓	✓	Raspberry Pi, humidity sensor, moisture sensor, LDR, temperature sensor, fan, sprinkler.
Punitha, Gowda, Devarajnayaka	Auto	✓	✗	Arduino UNO, moisture sensor, water pump, relay module.
Kannan, Sai kumar, Manoj Kumar, Surya prakash	Auto	✓	✓	PIC Microcontroller, transformer, GSM, temperature sensor, soil moisture sensor, humidity sensor, water pump, solenoid valve.

2.4 SYSTEM DEVELOPMENT LIFE CYCLE

The Software Development Life Cycle (SDLC) is a project management conceptual framework that covers the phases of a software development project. In this proposal, three SDLC models will be discussed.

2.4.1 Waterfall Model

Figure 2.1 shows the six phases in waterfall model. The entire procedure is separated into phases, and it is important to complete each one correctly before moving on to the next.

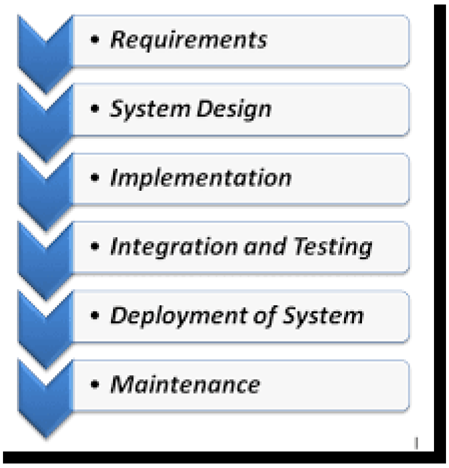


Figure 2.1 Waterfall model

The first phase is requirements analysis. All of the project's needs are analyzed and written in a specification document at this phase, and a feasibility study is performed to see if they are valid [21]. Any limitations or constraints like time, budget restraints that may affect the development process must be considered. A Requirements Understanding Documents (RUD) is prepared after complete analysis. Secondly, system design. During this phase, the system design is created, which determines hardware and system requirements such as data layers, programming languages, network infrastructure, user interface, and so on. It helps to create the overall system architecture, which is divided into two phases; high-level design and low-level design. Implementation is the third stage. The source code is written in accordance with the requirements. Code is created from the physical design specifications. The system is made up of units, which are little programs that are linked together. Before each unit is integrated, its functionality is tested by unit testing. Testing is the fourth phase. The code is handed over to the testing team at this point. Testers execute test cases manually or automatically to look for any potential weaknesses in the application. The client is also involved in the testing phase in order to ensure that all requirements are met. Any errors or problems detected during this phase are rectified to provide Quality Assurance. Deployment is the fifth phase. To be tested, the programme is deployed into a live environment. After the software has been deployed, end users can access it. In order to communicate the system's benefits, this phase may also incorporate real-time user training. Maintenance is the final phase. Following the deployment phase, providing software support and maintenance to ensure that it functions effectively is the next stage. The main purpose of this phase is to fix any issues, flaws, or errors that the client or consumers encounter when using the system.

2.4.2 Agile Model

Figure 2.2 shows the six phases in agile model. Iteratively, each incremental component is created. Each iteration is supposed to be simple and manageable, with each one taking only a few weeks to complete. At a time, one iteration is designed, created, and released to clients. There aren't any long-term plans in place.

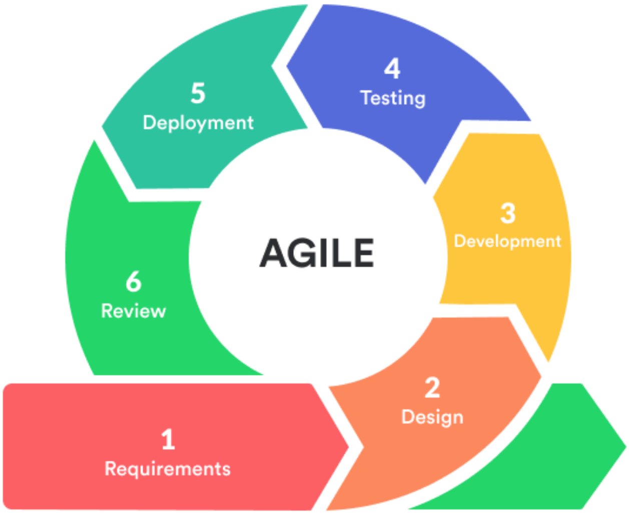


Figure 2.2 Agile model

The requirement phase is the initial step. Stakeholders assess the entire project to determine how much time and resources will be required to complete the development process [22]. The owner assesses the risks and prioritize the various functions according to their importance to the business. The second phase is the design phase. The software owner meets with the development team to present the requirements for the first stage. The committee then looks at the sequence in which functions should be presented, as well as the tools that are required, such as programming languages, syntax libraries, and basic frameworks. The ideal user interface can be prototyped by software engineers. Development and coding is the third phase. After the client and the team have agreed on the plan, the team produces the product. The product is being launched in phases, with each sprint aiming to improve the current version. Many changes to the initial version are likely to be made in order to improve functionality and add new features. Each cycle includes testing, and the final result must also be put to the test. Integration and testing make up the fourth phase. Since the product is now available to the public, the team needs to conduct a series of tests to ensure that it is fully functional. The developers will resolve any potential defects or errors as soon as they are detected. At this time, they also collect client feedback. Implementation and deployment is the fifth phase. Because the software has been fully deployed, customers can now access it. It is now in the maintenance phase. The software development team provides ongoing assistance during this phase to ensure that the system continues to function effectively and that any new issues are fixed. Iterations to improve an existing product or add new features can be made at any moment. The final stage is review. After completing all previous phases of development, the development team demonstrates the owner the outcome achieved in meeting the requirements.

2.4.3 Incremental Model

Figure 2.3 shows the four phases in each increment in incremental model. Each iteration goes through four phases. When a new system is released, it enhances on the prior release's functionality.

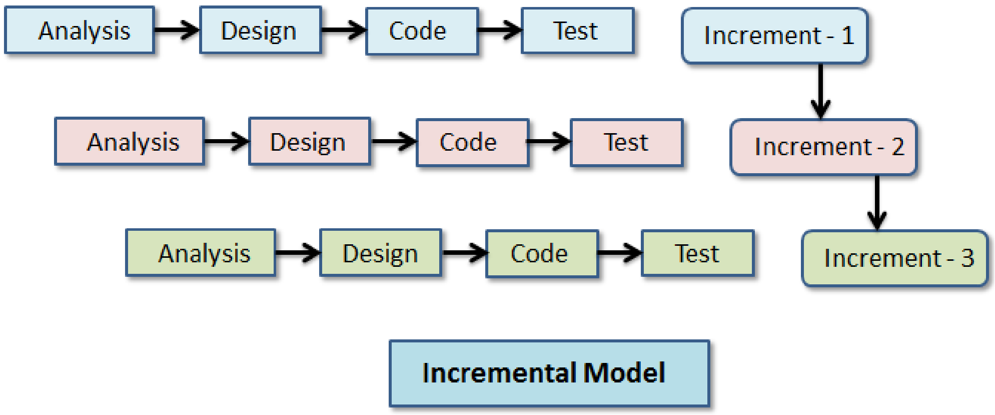


Figure 2.3 Incremental model

The first phase is requirement analysis. Throughout this phase, the product analysis experts define the product requirements, which include both functional and non-functional criteria, and confirm that the criteria are compatible [23]. This is a crucial step when using incremental models to create software. Design and development is the second phase. Throughout this phase, the development technique and system working designs had to be successful. After that, a design for archiving and carrying out this requirement is presented. When the software adds new functionality, the incremental model uses the development phase and style. Coding is the third phase. Coding is done in accordance with the objective of the requirements. The coding standards must be maintained without any additional hard codes or defaults. This phase also provides for the practical implementation of the designs. By completing this phase, the performance of the working product can be improved and updated. Testing is the final stage. During this phase, the performance of all existing functions, as well as any additional features, is evaluated. Furthermore, many approaches are used to test the various behaviours of each activity.

Table 2.3 shows the summary of the three SDLC models that have been discussed in the sub chapter. The comparison has been made based on watering method, live monitoring, storing data and sensors used. These features are compared because the related features will be referred to develop the proposed project, Household Plant Monitoring System.

Table 2.3 Summary of SDLC models

Model	Suitable project	Duration	Phases
Waterfall	Smaller and low budget projects where requirements are very well understood	7 – 8 months	Requirement, System Design, Implementation, Testing, Deployment, Maintenance.
Agile	Urgent project with a high level of complexity and uniqueness	5 – 6 months	Requirement, Design, Development and coding, Integration and testing, Implementation and deployment, Review.
Incremental	When all of the system's needs are precisely acknowledged	3 – 4 months	Requirement analysis, Design and development, Coding, Testing.

For this project, the suitable SDLC model is incremental model. This is because, smaller iteration makes testing and debugging easier. Because vulnerable aspects are detected and dealt with throughout the iteration process, this methodology delivers useable software quickly and early in the software development life cycle, and risk is easier to manage. The increment for the project may be divided as follows. In the first increment, setting up of input and output pins of microcontroller with sensors will be done. In the second increment, setting up of database and then in the third increment, setting up of the web application.

2.5 CONCLUSION

In Malaysia, mild climate related disasters are quite frequent to happen lately. Extreme climate conditions such as heavy rainfall and droughts will directly or indirectly affect the plants. Plant monitoring system can be utilised, to face and overcome these challenges. Essential data such as soil moisture, surrounding temperature and humidity of plants can be obtained and monitored for effective plant growth. Most of the system that have been developed are limited based on type of plants. Only one type of plant can be monitored at a time. The chosen SDLC model for this project is Incremental Model.

CHAPTER 3

METHODOLOGY

3.1 INTRODUCTION

This chapter aims to discuss in detail about the method that has used to implement plant monitoring in more efficient way using Internet of Things (IoT) integrated with database and web application. The method chosen for this project is Incremental model. There are four phases in incremental model which are requirement analysis, design and development, coding and lastly testing. Each iteration will go through all these phases.

3.2 REQUIREMENT ANALYSIS

Requirement analysis have been done by investigating agriculture from journals, student projects, websites and comparison of existing projects. For this project, the requirement analysis has been classified into two which are functional and non-functional requirement.

3.2.1 Functional Requirement

Functional requirements that have been identified are hardware and software used in this project. Hardware includes the components, sensors and microcontrollers used.

3.2.1.1 Hardware

Figure 3.1 is the example of DHT11 sensor. This sensor will detect the surrounding temperature and humidity. Temperature will be displayed in degree Celsius and humidity in percentage form. The three pins in the sensor are voltage, ground and data pin. These pins are used to connect with the microcontroller via jumper wires.

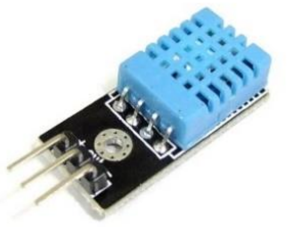


Figure 3.1 DHT11 Sensor

Figure 3.2 is the example resistive soil moisture sensor. This sensor will detect the moisture of soil. To measure the soil moisture, current will pass through the two rods and gets the resistance value. The rods must be connected with the small chip through the long wire. The four pins in the chip are voltage, ground, digital and analogue pin. These pins are used to connect with the microcontroller via jumper wires. The analogue output pin provides an analogue signal between 0V and the supply voltage while the internal comparator circuit's digital output is provided by digital output pin. There are specific inputs for both analogue and digital pins in the ESP32 microcontroller.

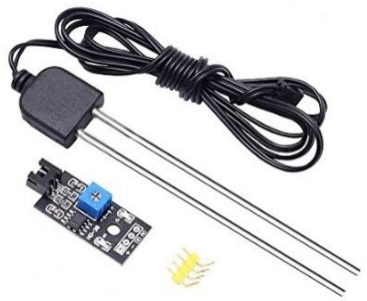


Figure 3.2 Soil moisture sensor

Figure 3.3 is the example of ESP32. This microcontroller is the main component in the system. It helps to collect readings from the sensors, send data to database, control relay module which helps to turn water pump on and off. Codes will be uploaded via Arduino IDE software. This microcontroller has Wi-Fi and Bluetooth features which makes this more suitable for this project. Through this, data retrieved can be sent via wireless connection to the database.

Figure 3.3 ESP32 DEVKIT V1 DOIT

Figure 3.4 is the example of relay module and Figure 3.5 is the example of water pump. Relay module is controlled by ESP32 microcontroller deciding the flow of current through it. If the current flows through the relay, then the water pump starts to pump water out. If current doesn't flow through the relay module, then water pump won't function.

Figure 3.4 Relay module

Figure 3.5 Water pump

Figure 3.6 is example of male to male jumper wires. These wires are used for the connection of circuit. Jumper wires helps to connect the pins in sensors with the pins in ESP32 microcontroller. The different types of jumper wires are male to male, male to female and female to female.

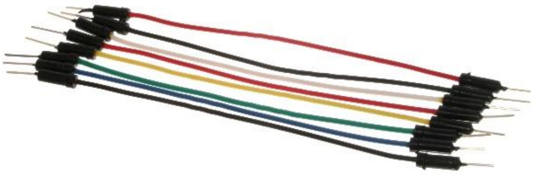


Figure 3.6 Jumper wires

Figure 3.7 shows 5V Step-Down Voltage Converter. This voltage converter can be given input voltage from 12V to 24V. Output volatge is fixed which is 5V. This component is used to give additional power supply as the ESP32 output voltage is not sufficient for all sensors.

Figure 3.7 5V Step-Down Voltage Converter

3.2.1.2 Software

Arduino Software (IDE)

This software is used to write codes in C language and upload it to the microcontroller. The codes make the sensors to collect readings like temperature, humidity and soil moisture. The readings are then sent and stored in the database. Serial monitor in the software displays the real-time data collected by the sensors.

Visual Studio Code

This software is used to write codes to develop web application and connect with database. The programming languages that will be used are PHP, HTML, CSS JavaScript. Through this software, data from database can be collected and displayed on the web application. Database can be controlled using VS code where data can be deleted, inserted, edited and make changes to the tables. Besides collecting and displaying data, the code to send email notification to the user is written in this software also.

Database (phpMyAdmin)

phpMyAdmin is mainly written in PHP. All the data retrieved from the microcontroller will be stored in database. The soil moisture, temperature and humidity data from the sensors for each plant will be stored according to the time interval set. There will be specific tables for each plant and each table consists of timestamp, soil moisture, temperature and humidity columns.

3.2.2 Non-functional requirement

Non-functional requirements that have been identified for this project are performance, usability and security. Performance is how quick can a system function for a user’s action. Usability is how well a user can try to utilize a system. Security is the protection or prevention from any threats or attacks.

Performance

Performance is important as data must be stored accurately according to the time interval set and also the plants. Sensors must collect accurate data and then store it in the database correctly. Then data from database must be fetched and displayed immediately within 5 seconds in the web page according to the plants. This is because, data display involves real-time data which needs to be precise and fast.

Usability

The system’s user interface will be easy to change the readings set for the sensors for each plant and also view the real time data of the sensors in the web page. Data to be set includes soil moisture and temperature which is on top of the plant interface. Real-time data of the plants like soil moisture, temperature and humidity can be seen below ‘Live Data’. User can click at the side bar which consists of Plant A, Plant B, Plant C and Profile to view or edit any information. There’s also a table at the bottom which displays data retrieved at each hour. Users no need to go database to view previous data as this table displays simplified data.

Security

Login feature in the system will allow authorized user to access and monitor the plants condition. User need username and password to login to the system. Click forget password link if password had been forgotten. This link will send a password reset notification via user’s email. User can also register for new account by entering username, email and password.

3.2.2.1 Rules

Table 3.1 shows the conditions required for automatic irrigation to occur. These conditions are needed for optimum plant growth. Only temperature and soil moisture are compared because temperature and humidity is inversely proportional and humidity does not play a big role in plant growth. Humidity is still needed for monitoring purpose. Water is not required even when the surrounding temperature for plants is more than range and but the soil moisture is in range. When the temperature is more than range and soil is dry, water is required. When the temperature is in range and the soil is dry, water is required. Water is not required when temperature is in range and soil moisture is in range. Table 3.1 shows summary of the conditions.

Table 3.1 Conditions for automatic irrigation

Temperature	Soil moisture	Water pump
More than range	In range	OFF
More than range	Dry	ON
In range	Dry	ON
In range	In range	OFF

3.3 DESIGN

Design includes flowchart, use case, context diagram, activity diagram, data flow diagram, entity relationship diagram, data dictionary, circuit design, design of full system and interface design of web application of Household Plant Monitoring System.

3.3.1 Flowchart for Household Plant Monitoring System

Figure 3.8 is the flowchart for Household Plant Monitoring System. This figure shows the flow from selecting the type of plant, the sensors read data, the conditions to validate the data and then following by the actions.

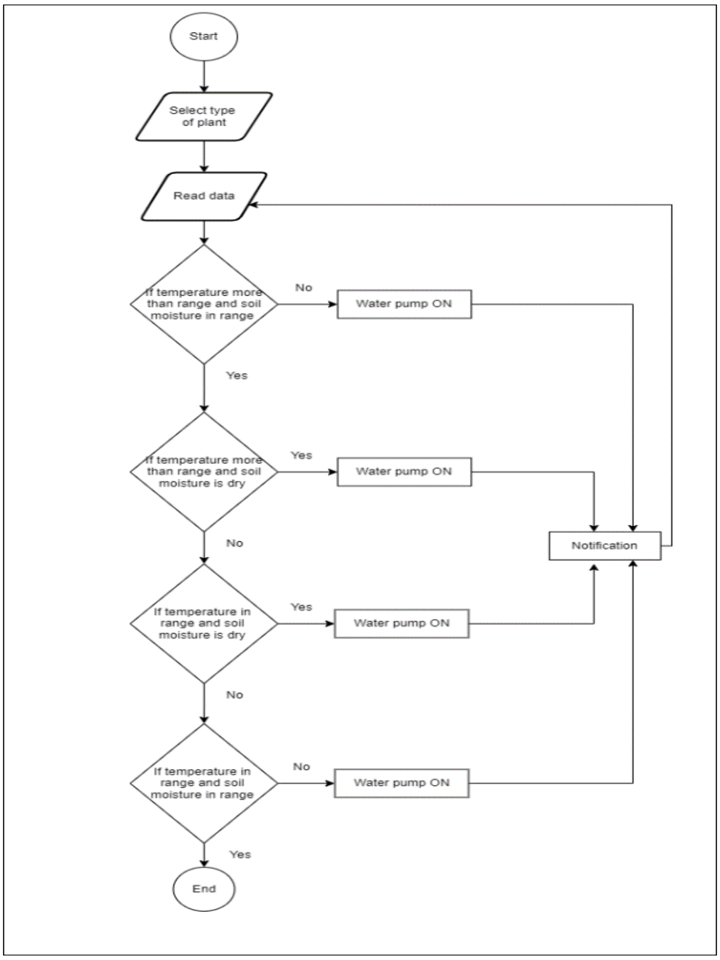

Start
Select type of plant
Read data
If temperature more than range and soil moisture in range
No
Water pump ON
Yes
If temperature more than range and soil moisture is dry
Yes
Water pump ON
No
Notification
If temperature in range and soil moisture is dry
Yes
Water pump ON
No
If temperature in range and soil moisture in range
No
Water pump ON
Yes
End

Figure 3.8 Flowchart for Household Plant Monitoring System

3.3.2 Use Case Diagram for Household Plant Monitoring System

Figure 3.9 is the use case diagram for Household Plant Monitoring System. There are three modules in the use case which are manage user, manage data and manage notification.

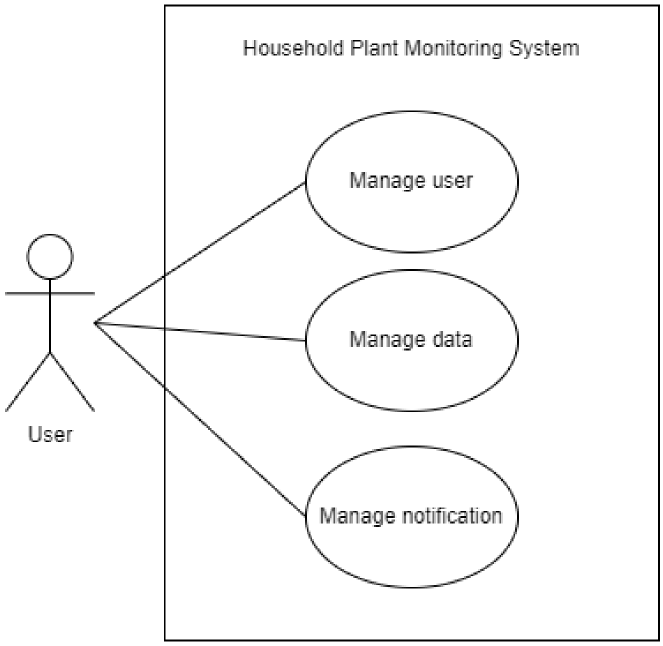


Figure 3.9 Use case diagram of Household Plant Monitoring System

In the module manage user, user can login into to system using username and password. New users can sign up to create new account which requires username, email and password. User can update username, password and email. In addition, password reset feature is available.

User can input the suitable data like temperature and soil moisture for each plant, in the module manage data. The input data can also be edited. User able to delete unwanted data that will be displayed on a table, monitor real-time data displayed on the web application and also view the data stored in database.

User will receive notification to smartphone via email when water pump is turned on, in the module manage notification.

3.3.3 Context diagram for Household Plant Monitoring System

Figure 3.10 is the context diagram which shows the relationship of Household Plant Monitoring System with external factors like sensors and user.

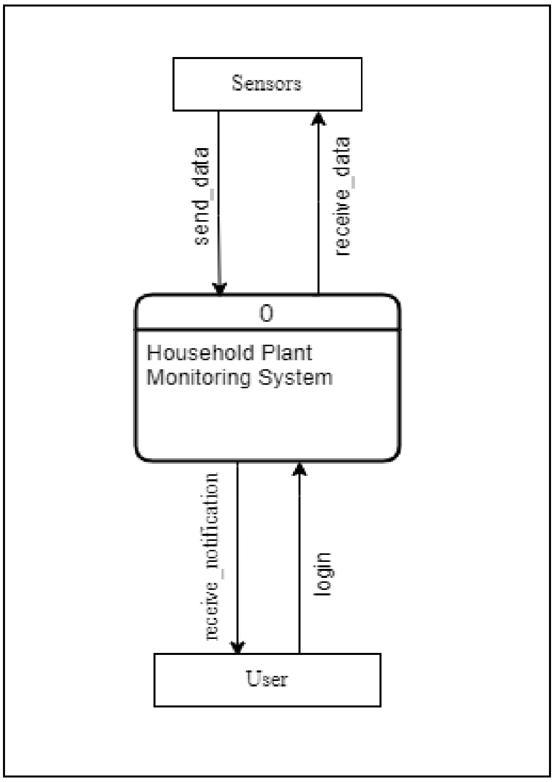


Figure 3.10 Context diagram of Household Plant Monitoring System

3.3.4 Activity diagram for Household Plant Monitoring System

Figure 3.11 is the activity diagram for Household Plant Monitoring System. The activities begin from login, user set data, microcontroller collect reading. Two activities occur at one time which is microcontroller validates data and also sends data to database. Validating data to determine irrigation process using the conditions. Web application gets and displays data from database. Actions come after going through the conditions.

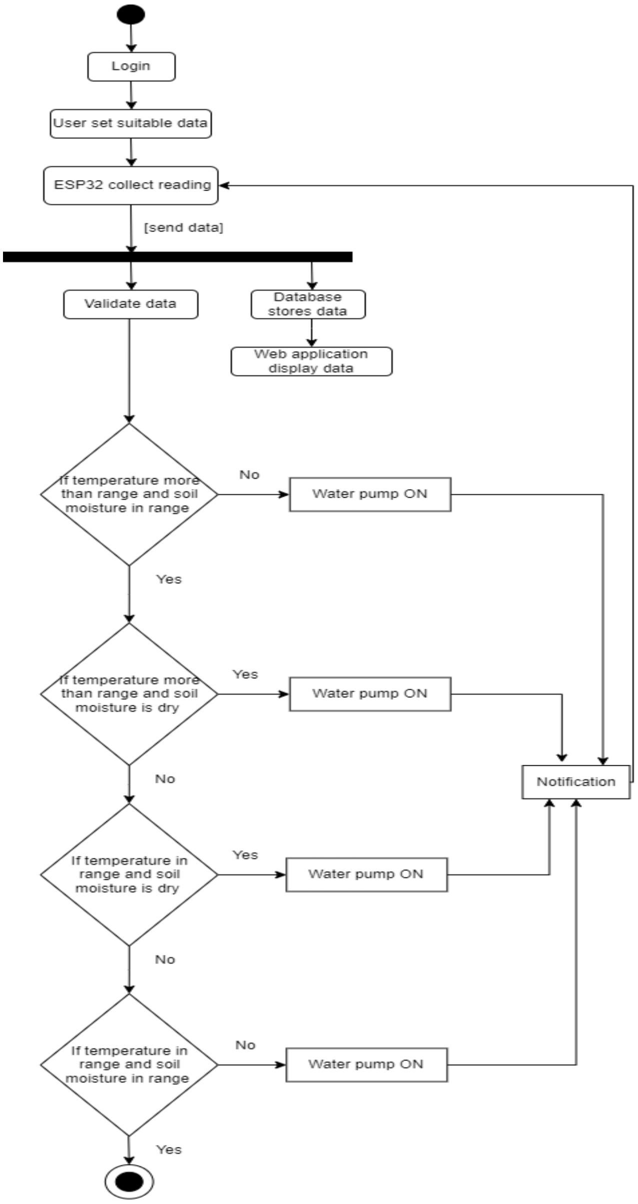

Login
User set suitable data
ESP32 collect reading
[send data]
Validate data
Database stores data
Web application display data
If temperature more than range and soil moisture in range
No
Water pump ON
Yes
If temperature more than range and soil moisture is dry
Yes
Water pump ON
No
Notification
If temperature in range and soil moisture is dry
Yes
Water pump ON
No
If temperature in range and soil moisture in range
No
Water pump ON
Yes

Figure 3.11 Activity diagram for Household Plant Monitoring System

3.3.5 Data Flow Diagram for Household Plant Monitoring System

Figure 3.12 shows the flow of data in Household Plant Monitoring System. Data inputs and outputs between user and entities, data storage in database.

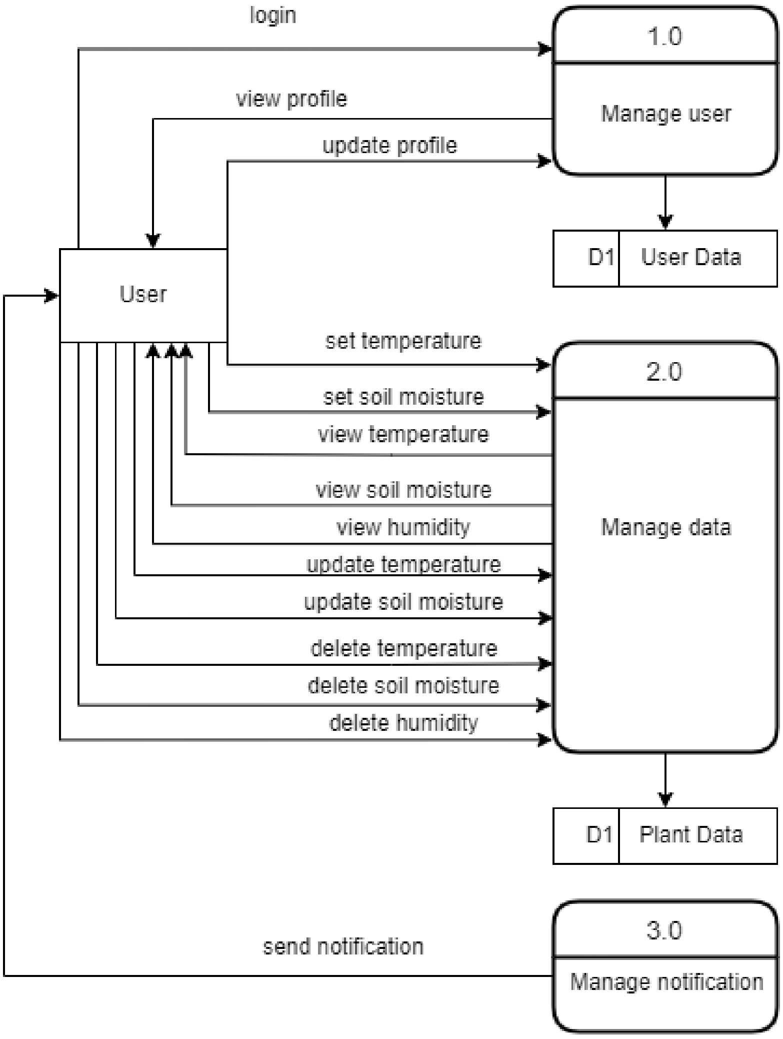


Figure 3.12 Data Flow Diagram of Household Plant Monitoring System

3.3.6 Entity Relationship Diagram for Household Plant Monitoring System

Figure 3.13 shows the relationship between the entities in the system. There are eight entities which are users, plant, dht11, dht11_2, dht11_3, preset, preset_2 and preset_3. Users entity contains user information and plant entity contains plant information. dht11, dht11_2 and dht11_3 entities contain the sensors readings, pump state and date for plant A, B and C respectively. preset, preset_2 and preset_3 entities contain temperature and soil moisture threshold for plant A, B and C respectively.

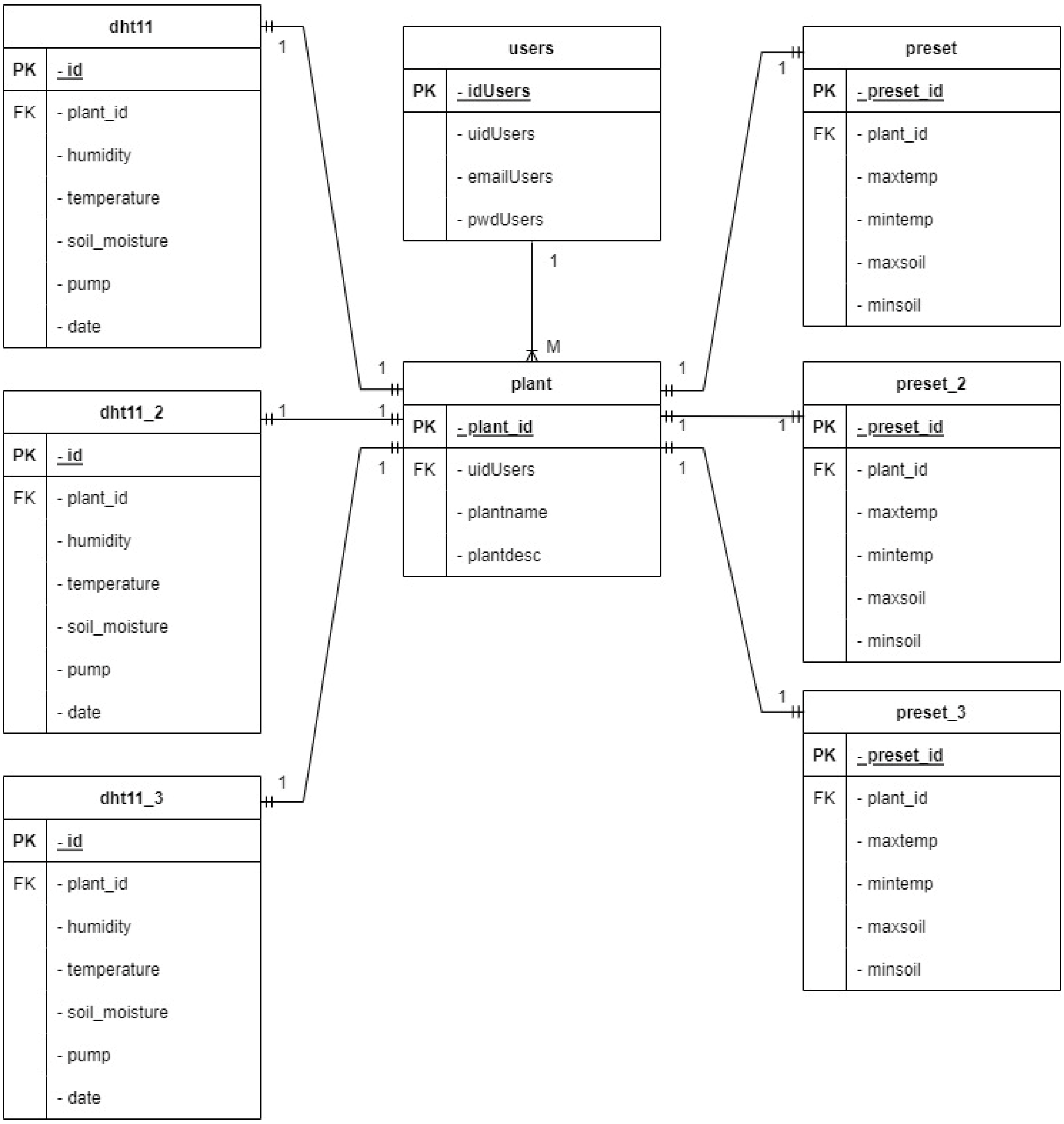


Figure 3.13 Entity Relationship Diagram for Household Plant Monitoring System

3.3.7 Data Dictionary

Data dictionary shows the entities and their attributes used to store in database. Table for data dictionary contains five columns which are entity, data type, description, example and constraint.

Table 3.2 shows the data dictionary table for users. This table consists entities which are idUsers, uidUsers, emailUsers and pwdUsers together with data type, description, example and constraint.

Table 3.2 Data dictionary for users

Entity	Data Type	Description	Example	Constraint
idUsers	int(11)	User Id	1	PK
uidUsers	varchar(100)	Username	Suugen raj	
emailUsers	varchar(100)	User's email	suugenraj.venugopal@gmail.com	
pwdUsers	varchar(100)	User password	$2y$10$Rn3Aa	

Table 3.3 shows the data dictionary table for plant. This table consists entities which are plant_id, uidUsers, plantname and plantdesc together with data type, description, example and constraint.

Table 3.3 Data dictionary for plant

Entity	Data Type	Description	Example	Constraint
plant_id	int(11)	Plant id	1	PK
uidUsers	varchar(100)	Username	raj	FK
plantname	varchar(100)	Name of plant	Chili plant	
plantdesc	varchar(4000)	Description of plant	Chili plant can grow in hot weather.	

Table 3.4 shows the data dictionary table for dht11. This table consists entities which are id, plant_id, soil_moisture, temperature, humidity, pump and date together with data type, description, example and constraint.

Table 3.4 Data dictionary for dht11

Entity	Data Type	Description	Example	Constraint
id	int(11)	Reading id	12	PK
plant_id	int(11)	Plant id	1	FK
soil_moisture	int(11)	Soil moisture reading	674	
temperature	float	Temperature reading	33.60	
humidity	float	Humidity reading	59	
pump	int(11)	Water pump state	1	
date	timestamp	Date and time	2023-01-18 18:14:23	

Table 3.5 shows the data dictionary table for dht11_2. This table consists entities which are id, plant_id, soil_moisture, temperature, humidity, pump and date together with data type, description, example and constraint.

Table 3.5 Data dictionary for dht11_2

Entity	Data Type	Description	Example	Constraint
id	int(10)	Reading id	23	PK
plant_id	int(11)	Plant id	2	FK
soil_moisture	int(100)	Soil moisture reading	1023	
temperature	float(100)	Temperature reading	29.8	
humidity	float(100)	Humidity reading	68	
pump	int(11)	Water pump state	1	
date	timestamp	Date and time	2023-01-18 12:32:48	

Table 3.6 shows the data dictionary table for dht11_3. This table consists entities which are id, plant_id, soil_moisture, temperature, humidity, pump and date together with data type, description, example and constraint.

Table 3.6 Data dictionary for dht11_3

Entity	Data Type	Description	Example	Constraint
id	int(10)	Reading id	15	PK
plant_id	int(11)	Plant id	3	FK
soil_moisture	int(100)	Soil moisture reading	2312	
temperature	float(100)	Temperature reading	31	
humidity	float(100)	Humidity reading	71	
pump	int(11)	Water pump state	0	
date	timestamp	Date and time	2023-01-18 05:44:28	

Table 3.7 shows the data dictionary table for preset. This table consists entities which are preset_id, plant_id, maxtemp, mintemp, maxsoil and minsoil together with data type, description, example and constraint.

Table 3.7 Data dictionary for preset

Entity	Data Type	Description	Example	Constraint
preset_id	int(10)	Preset id	1	PK
plant_id	int(11)	Plant id	1	FK
maxtemp	varchar(11)	Maximum temperature	32	
mintemp	varchar(11)	Minimum temperature	18	
maxsoil	varchar(11)	Maximum soil moisture	1200	
minsoil	varchar(11)	Minimum soil moisture	600	

Table 3.8 shows the data dictionary table for preset_2. This table consists entities which are preset_id, plant_id, maxtemp, mintemp, maxsoil and minsoil together with data type, description, example and constraint.

Table 3.8 Data dictionary for preset_2

Entity	Data Type	Description	Example	Constraint
preset_id	int(10)	Preset id	2	PK
plant_id	int(11)	Plant id	2	FK
maxtemp	varchar(11)	Maximum temperature	35	
mintemp	varchar(11)	Minimum temperature	19	
maxsoil	varchar(11)	Maximum soil moisture	1400	
minsoil	varchar(11)	Minimum soil moisture	600	

Table 3.9 shows the data dictionary table for preset_3. This table consists entities which are preset_id, plant_id, maxtemp, mintemp, maxsoil and minsoil together with data type, description, example and constraint.

Table 3.9 Data dictionary for preset_3

Entity	Data Type	Description	Example	Constraint
preset_id	int(10)	Preset id	3	PK
plant_id	int(11)	Plant id	3	FK
maxtemp	varchar(11)	Maximum temperature	30	
mintemp	varchar(11)	Minimum temperature	20	
maxsoil	varchar(11)	Maximum soil moisture	1300	
minsoil	varchar(11)	Minimum soil moisture	600	

3.3.8 Circuit

Figure 3.14 shows the circuit connection for Household Plant Monitoring System. Circuit connection consists of microcontroller which is ESP32, sensors, relay module, water pump, breadboard and jumper wires.

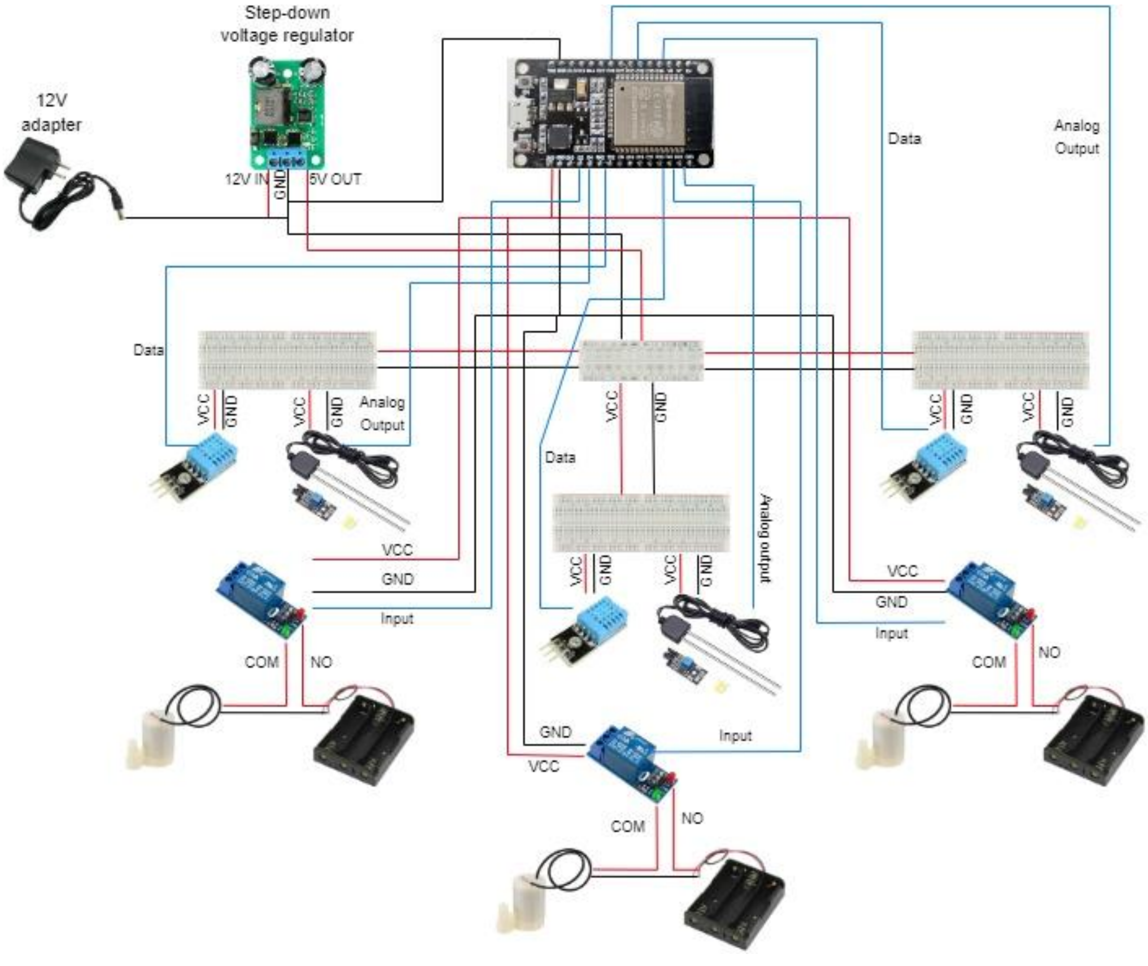


Figure 3.14 Circuit of Household Plant Monitoring System

3.3.9 Architecture design

Figure 3.15 shows the architecture design of Household Plant Monitoring System. Architecture design consists of microcontroller which is ESP32, sensors, relay module, water pump, breadboard, jumper wires, database and web application.

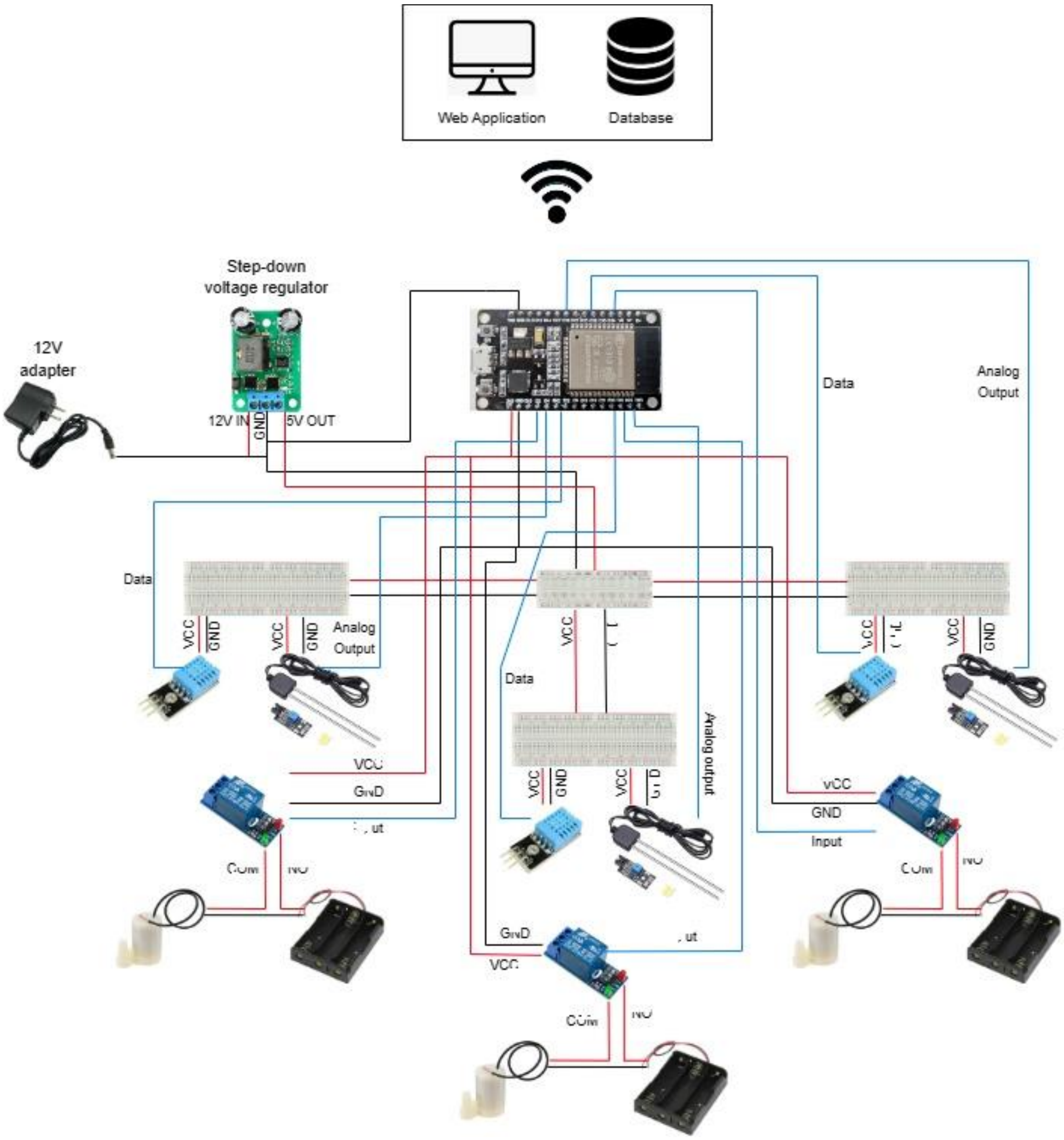


Figure 3.15 Architecture design of Household Plant Monitoring System

3.3.10 Interface design of web application

Figure 3.16 is the login page interface. User need enter username and password to login. User can click signup to create new profile. There is an option for users who forgot password.

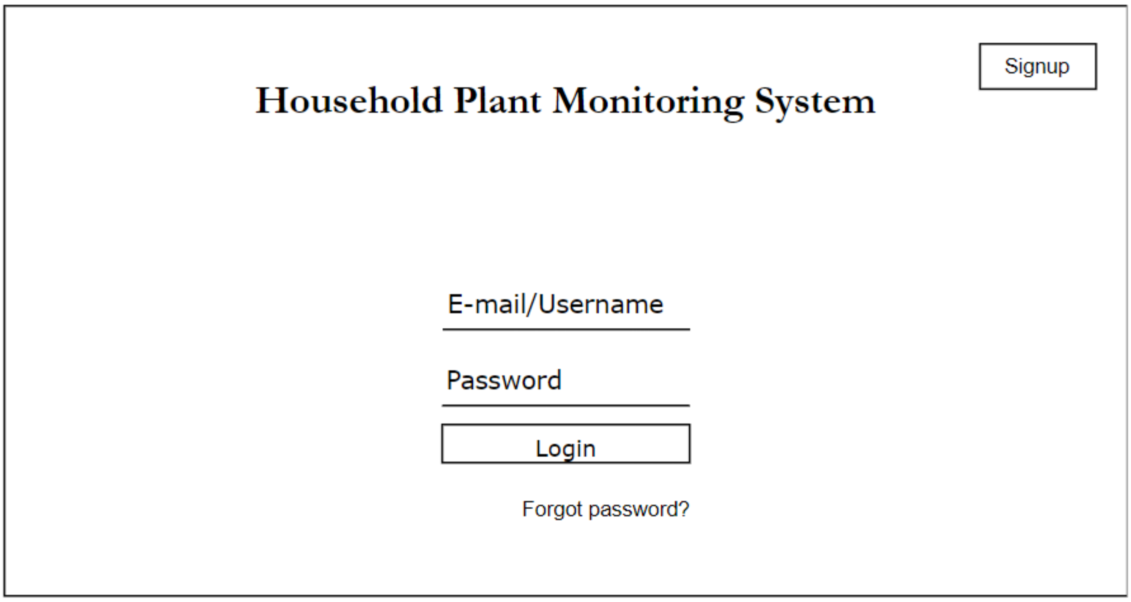


Figure 3.16 Login page interface

Figure 3.17 is the signup interface. To create account, user need to enter username, password and email address.

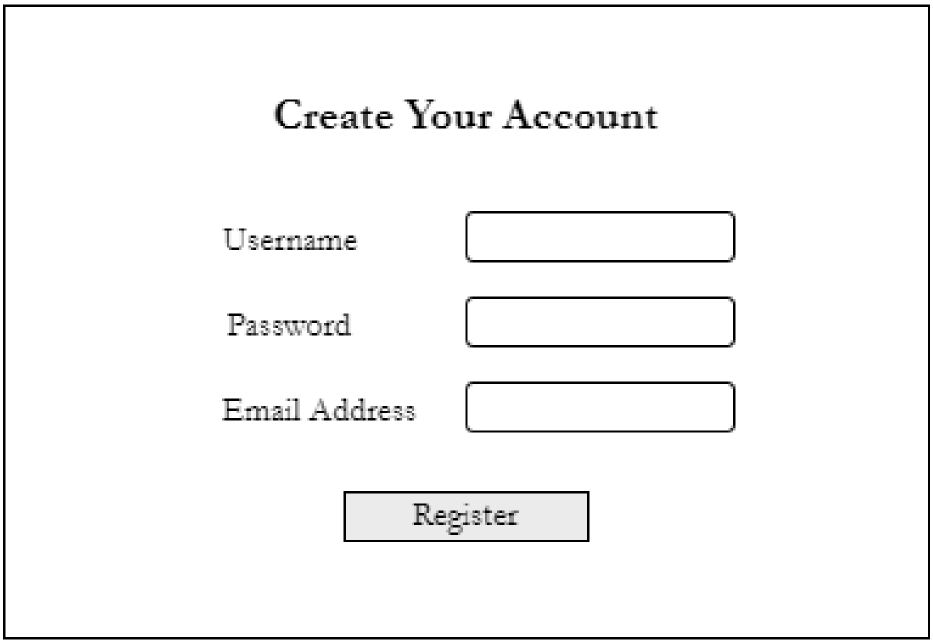
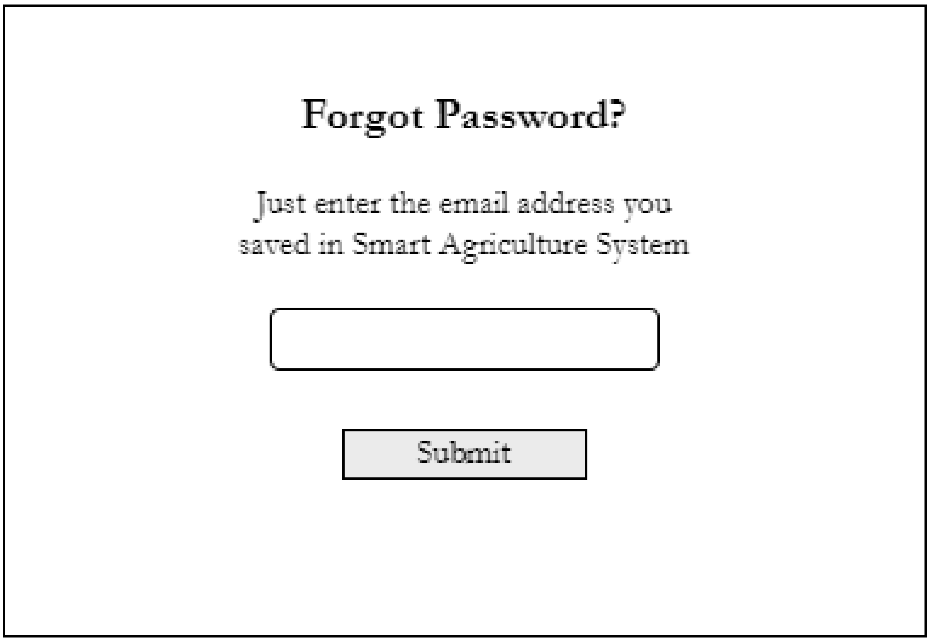


Figure 3.17 Signup page interface

Figure 3.18 is the forgot password page interface. User need to enter email address used to register the system. Then it will be redirected to another page to change new password.



Forgot Password?

Just enter the email address you saved in Smart Agriculture System

Figure 3.18 Forgot password page interface

Figure 3.19 shows reset password interface. User need to enter new password and then enter the password again for confirmation.

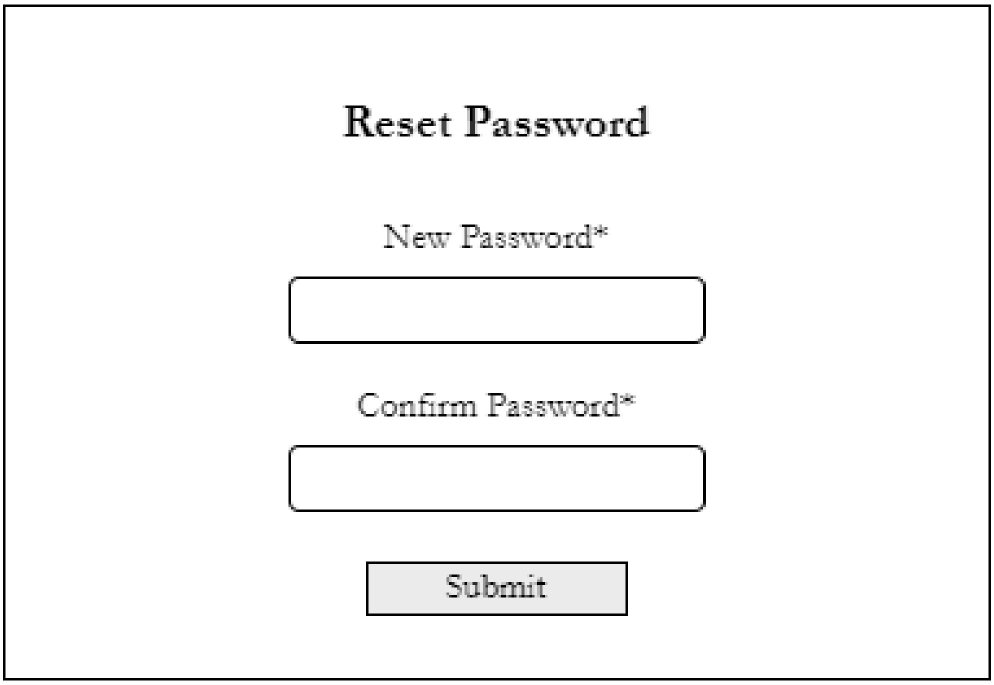
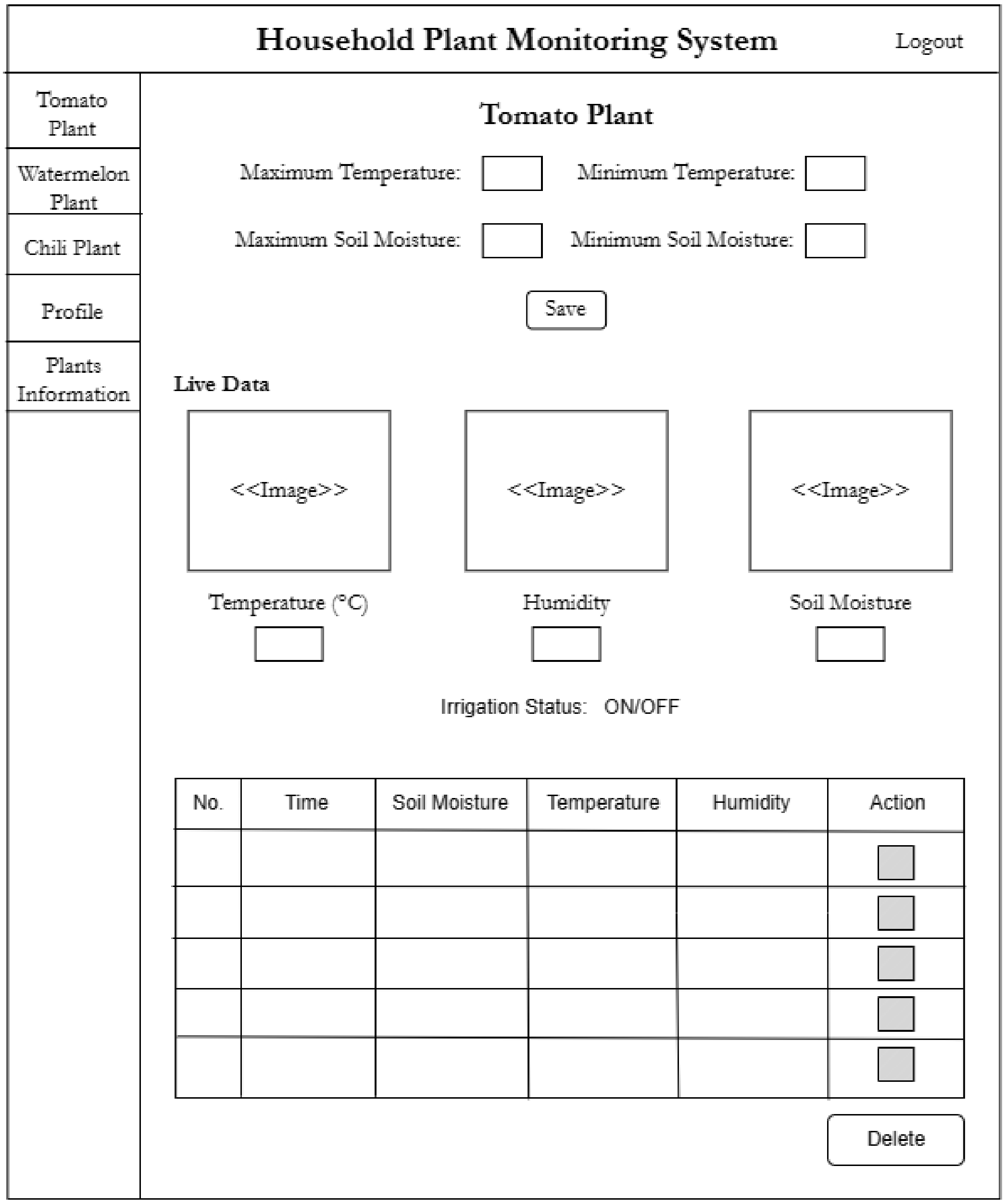


Figure 3.19 Reset password interface

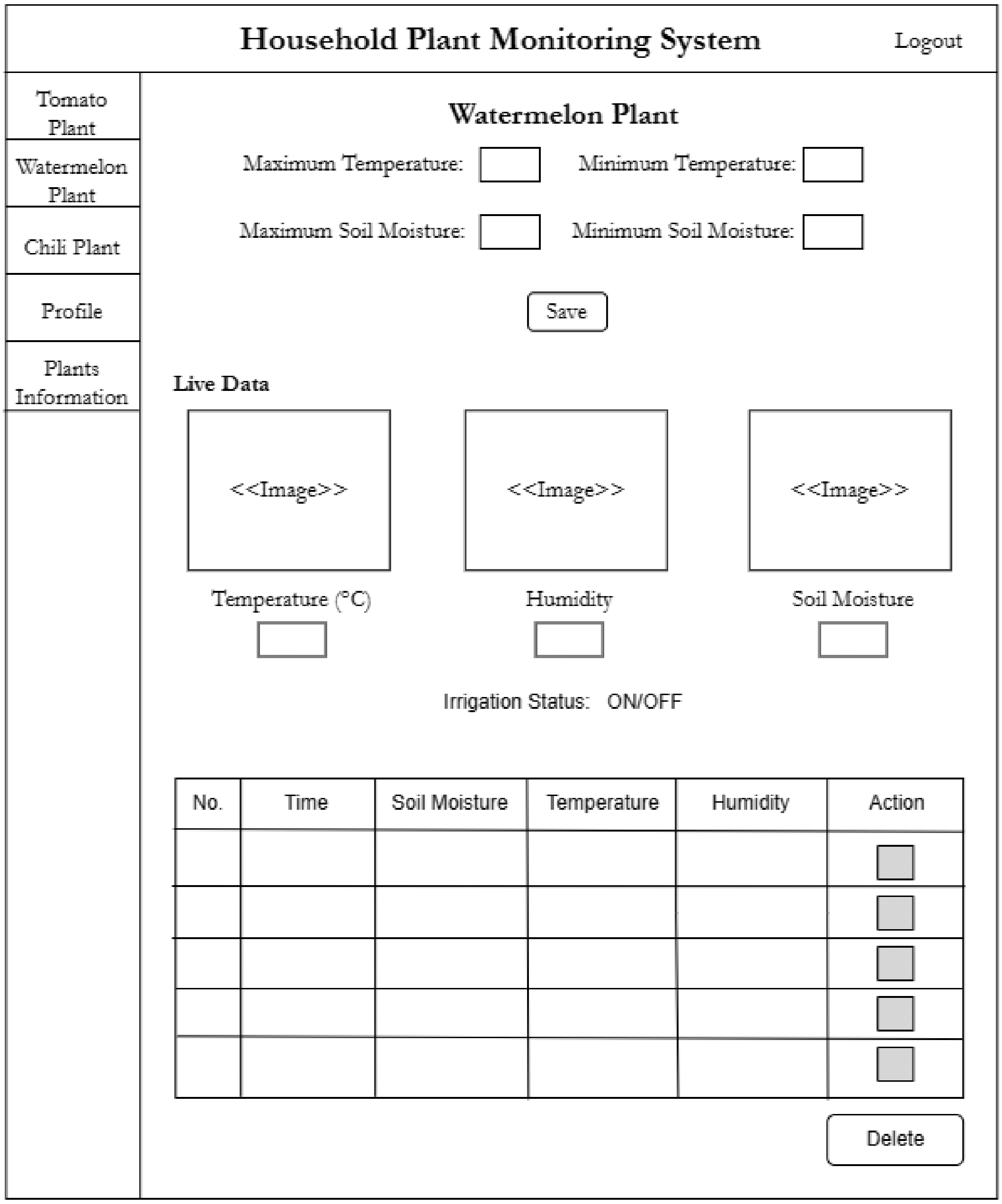
Figure 3.20 shows interface of Plant A. On the left side bar, user can choose plant A view the interface. On the top, user can input the suitable data for the plant where edit and save the data can be done. Below that, once the system starts to operate, live data such as temperature, humidity and soil moisture will be displayed. Irrigation status will be shown ON if the water pump is ON, if not irrigation status will be OFF. All the data retrieved will be stored on the table below according to the time interval set.



No.	Time	Soil Moisture	Temperature	Humidity	Action
					☐
					☐
					☐
					☐
					☐

Figure 3.20 Interface of Plant A

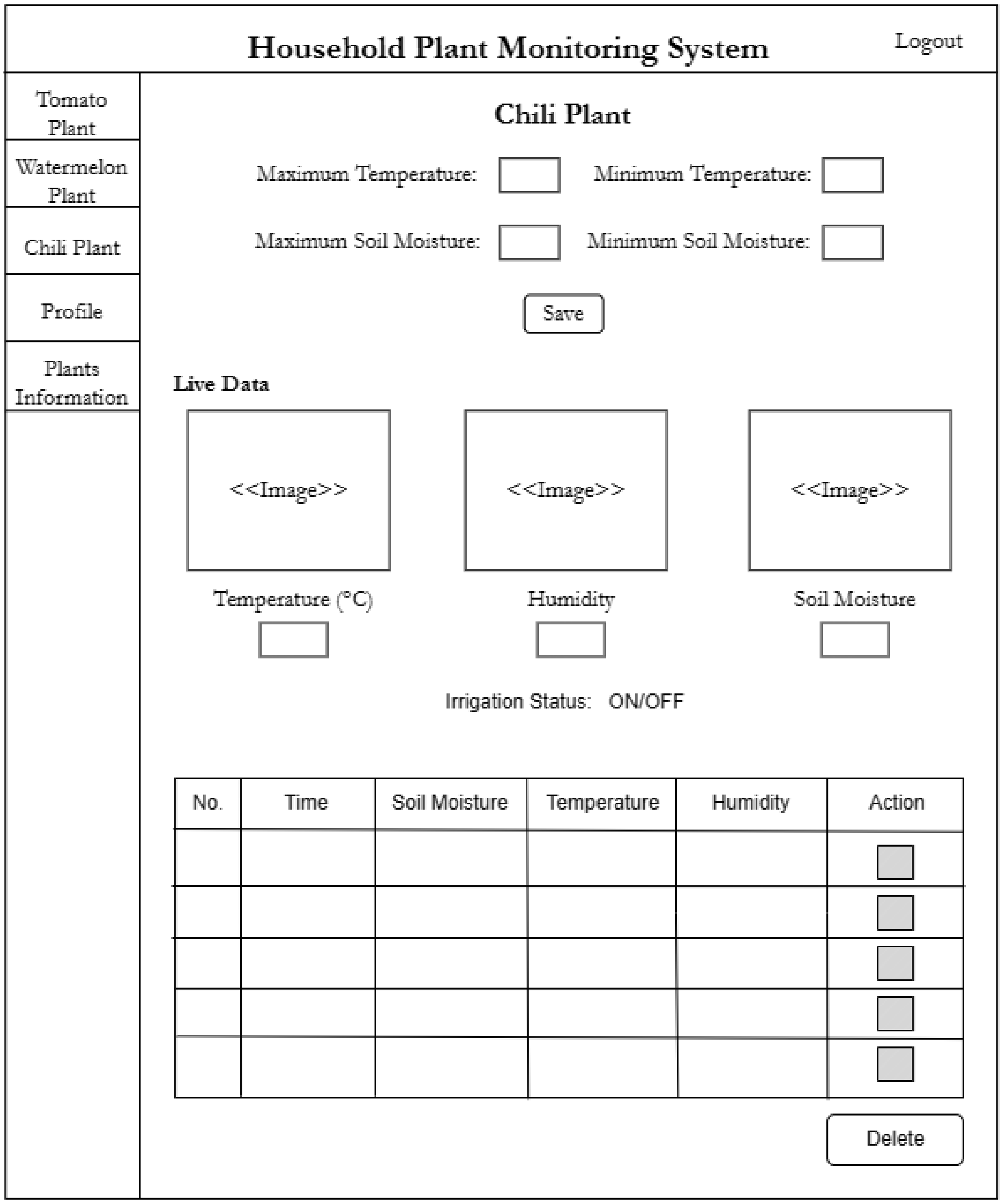
Figure 3.21 shows interface of Plant B. On the left side bar, user can choose plant B view the interface. On the top, user can input the suitable data for the plant where edit and save the data can be done. Below that, once the system starts to operate, live data such as temperature, humidity and soil moisture will be displayed. Irrigation status will be shown ON if the water pump is ON, if not irrigation status will be OFF. All the data retrieved will be stored on the table below according to the time interval set.



No.	Time	Soil Moisture	Temperature	Humidity	Action
					□
					□
					□
					□
					□

Figure 3.21 Interface of Plant B

Figure 3.22 shows interface of Plant C. On the left side bar, user can choose plant C view the interface. On the top, user can input the suitable data for the plant where edit and save the data can be done. Below that, once the system starts to operate, live data such as temperature, humidity and soil moisture will be displayed. Irrigation status will be shown ON if the water pump is ON, if not irrigation status will be OFF. All the data retrieved will be stored on the table below according to the time interval set.



Maximum Temperature: □ Minimum Temperature: □

Maximum Soil Moisture: □ Minimum Soil Moisture: □

Temperature (°C) □

Humidity □

Soil Moisture □

Irrigation Status: ON/OFF

No.	Time	Soil Moisture	Temperature	Humidity	Action
					□
					□
					□
					□
					□

Figure 3.22 Interface of Plant C

Figure 3.23 shows interface of profile. User can edit and save name and email.

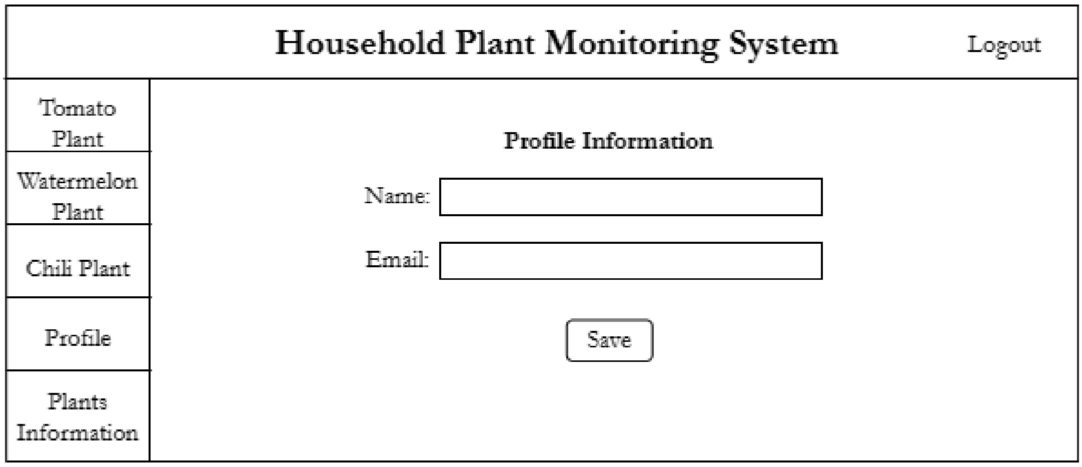


Figure 3.23 Interface of profile

Figure 3.24 shows interface of plants information. Name and description of each plant will be displayed on this interface.

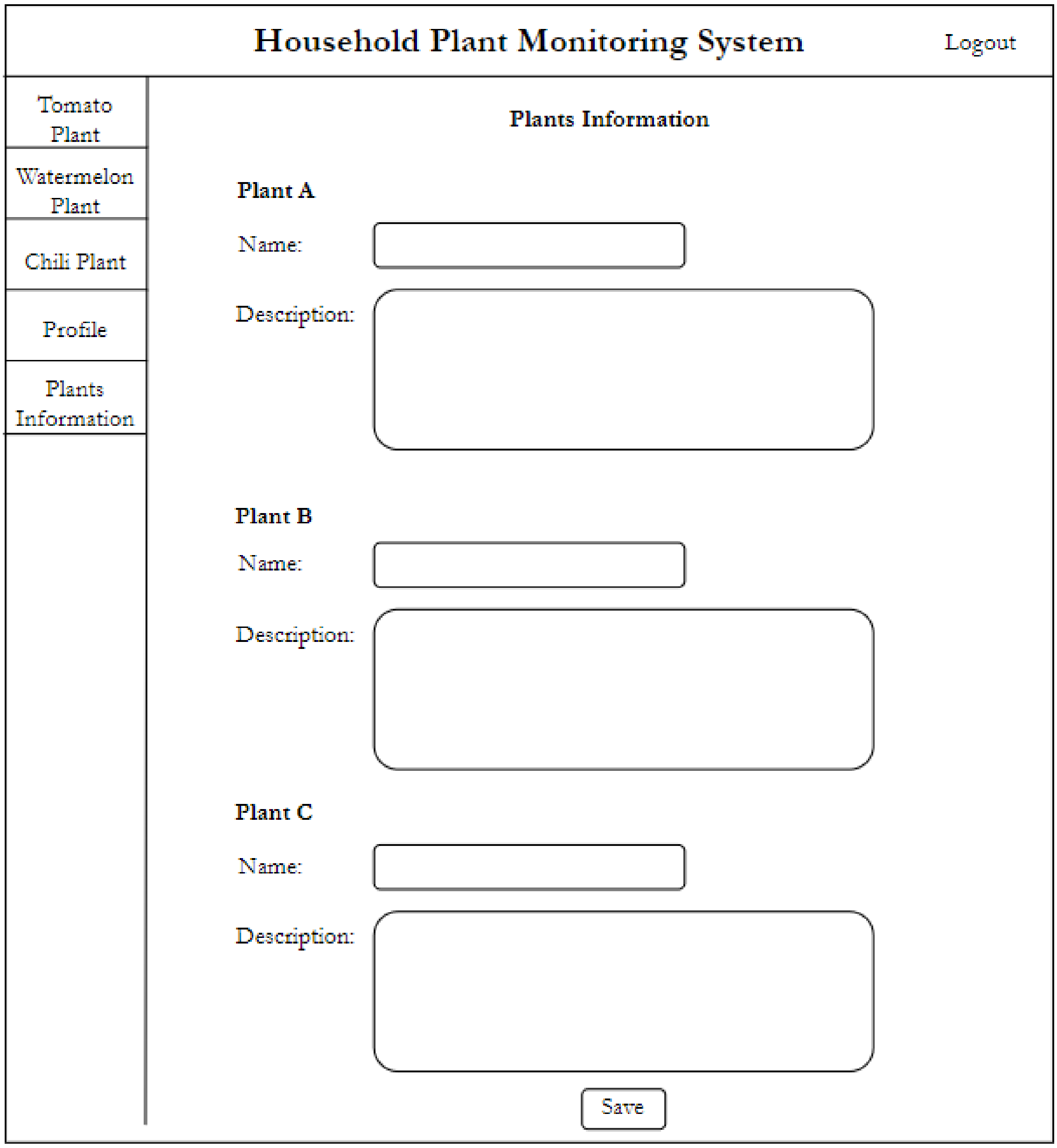


Figure 3.24 Interface of plants information

3.4 DEVELOPMENT

The development phase consists of four types of development which includes circuit development, web application interface development, database development and functional development.

3.4.1 Circuit development

Figure 3.25 shows the circuit development of Household Plant Monitoring System. Microcontroller is connected via jumper wires, breadboard with soil moisture sensor and DHT11 (Temperature & Humidity) sensor.

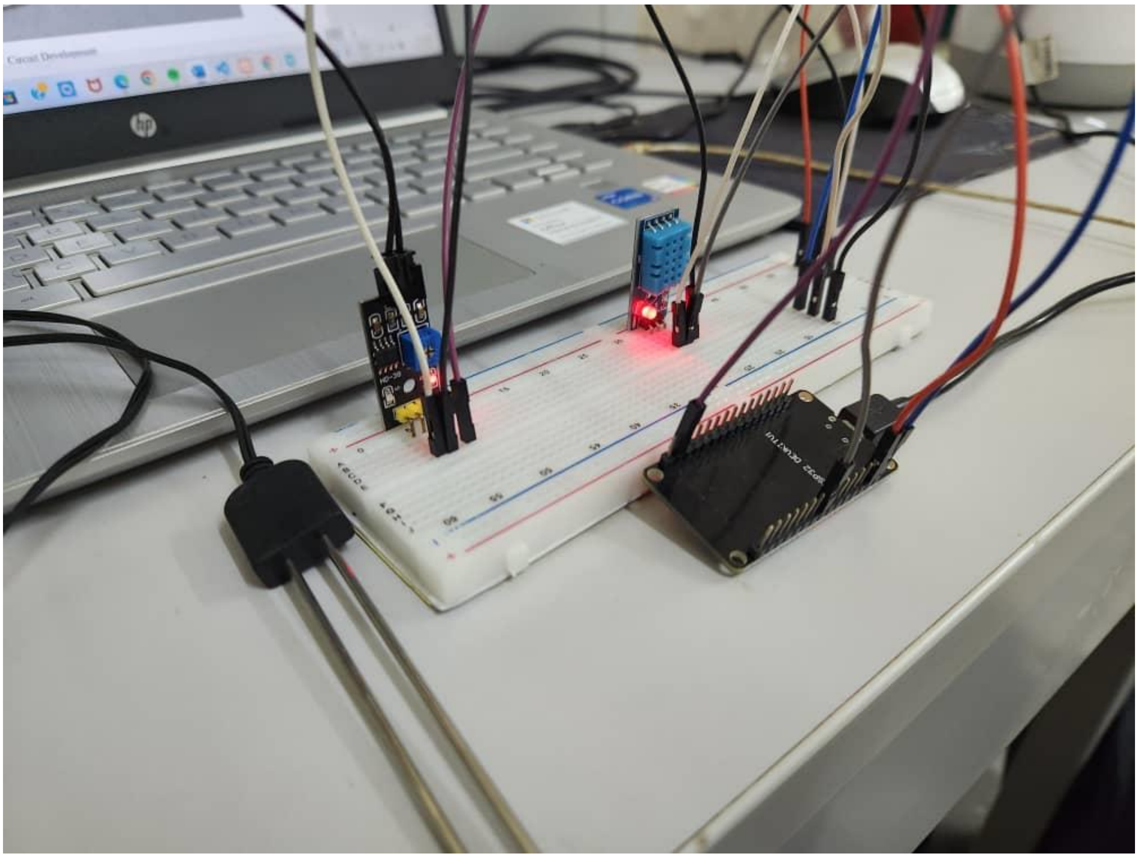


Figure 3.25 Circuit Development

3.4.2 Web Application Interface development

The interfaces for the web application designed is developed using Visual Studio Code software. The programming languages that is used are PHP, JavaScript, HTML and CSS.

3.4.3 Database development

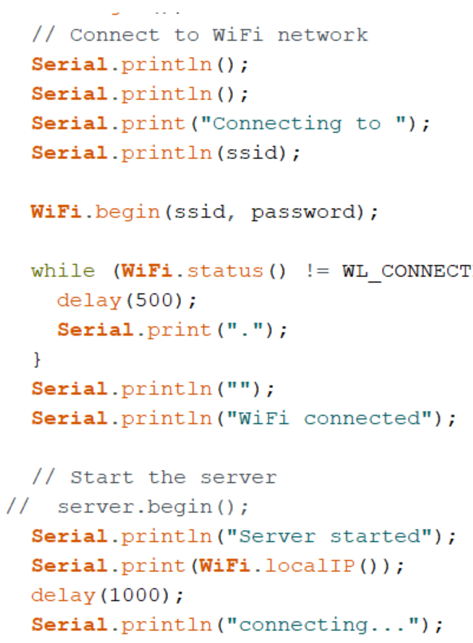
The database that is used for this project is phpMyAdmin. Code to retrieve data from Arduino microcontroller and then storing in database is written in Visual Studio Code software.

3.4.4 Functional development

In functional development, codes written from arduino and visualstudio code are shown. These important codes are shown together with description.

3.4.4.1 ESP32 Wi-Fi connection code

Figure 3.26 shows the code for ESP 32 Wi-Fi connection. Baud rate is set to 115200 and WiFi.begin will try to connect to the Wi-Fi credentials provided.

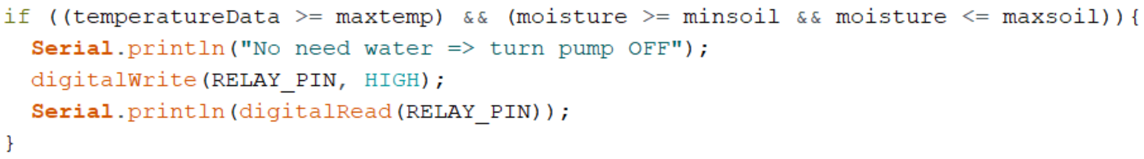


```
  // Connect to WiFi network
  Serial.println();
  Serial.println();
  Serial.print("Connecting to ");
  Serial.println(ssid);

  WiFi.begin(ssid, password);

  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("");
  Serial.println("WiFi connected");

  // Start the server
//  server.begin();
  Serial.println("Server started");
  Serial.print(WiFi.localIP());
  delay(1000);
  Serial.println("connecting...");
```

Figure 3.26 ESP 32 Wi-Fi connection

3.4.4.2 Code to set conditions

Figure 3.27 shows the temperature and soil moisture conditions set to water the plants. If the temperature is above the value set and soil moisture is in range of the value set, the plant doesn’t need water thus relay module will stay HIGH where no current will pass by to the water pump.

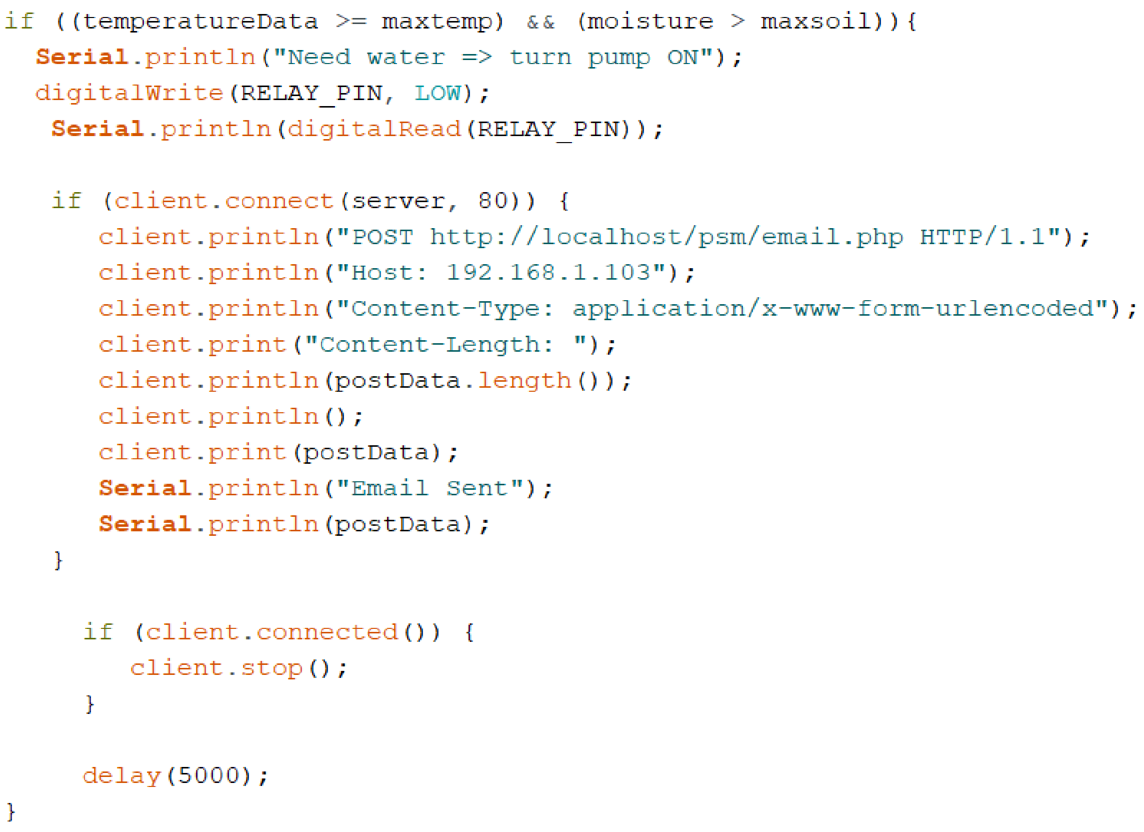


```
if ((temperatureData >= maxtemp) && (moisture >= minsoil && moisture <= maxsoil)){
  Serial.println("No need water => turn pump OFF");
  digitalWrite(RELAY_PIN, HIGH);
  Serial.println(digitalRead(RELAY_PIN));
}
```

Figure 3.27 Conditions for watering the plants

3.4.4.3 Email function code

Figure 3.28 shows the email function code. Email function works only when the relay module is in LOW state where it allows current to pass through to the water pump to water the plants. To send email, server’s IP address and port number is needed. It will then read and run email.php script to send email.

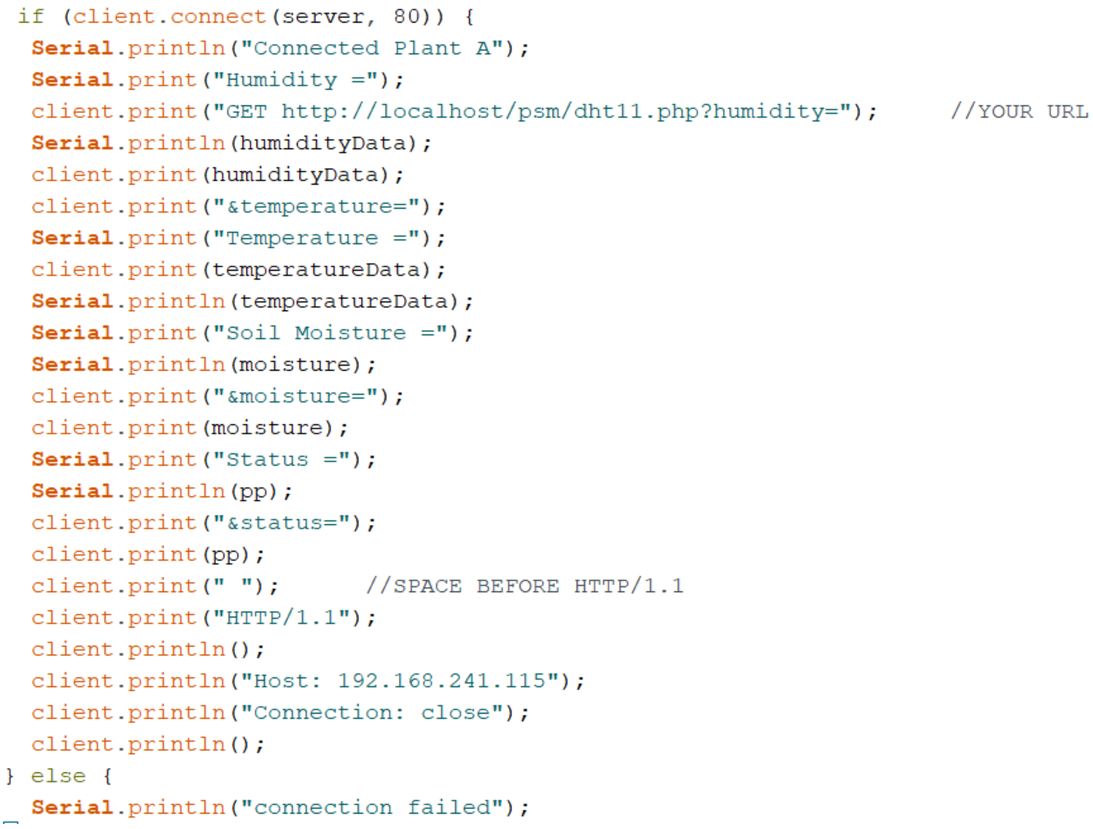


```
if ((temperatureData >= maxtemp) && (moisture > maxsoil)){
  Serial.println("Need water => turn pump ON");
  digitalWrite(RELAY_PIN, LOW);
   Serial.println(digitalRead(RELAY_PIN));

   if (client.connect(server, 80)) {
      client.println("POST http://localhost/psm/email.php HTTP/1.1");
      client.println("Host: 192.168.1.103");
      client.println("Content-Type: application/x-www-form-urlencoded");
      client.print("Content-Length: ");
      client.println(postData.length());
      client.println();
      client.print(postData);
      Serial.println("Email Sent");
      Serial.println(postData);
   }

     if (client.connected()) {
        client.stop();
     }

     delay(5000);
}
```

Figure 3.28 Email function

3.4.4.4 Code to send sensor data to database

Figure 3.29 shows the code to send sensor data to database. The code below will get the sensor data and will make a HTTP request to dht11.php page where the page contains code to get sensor data from arduino ide to store it in database. The pd variable below will read the relay module's state whether HIGH (0) or LOW (1). To send the data to database, server's IP address and port number is needed.

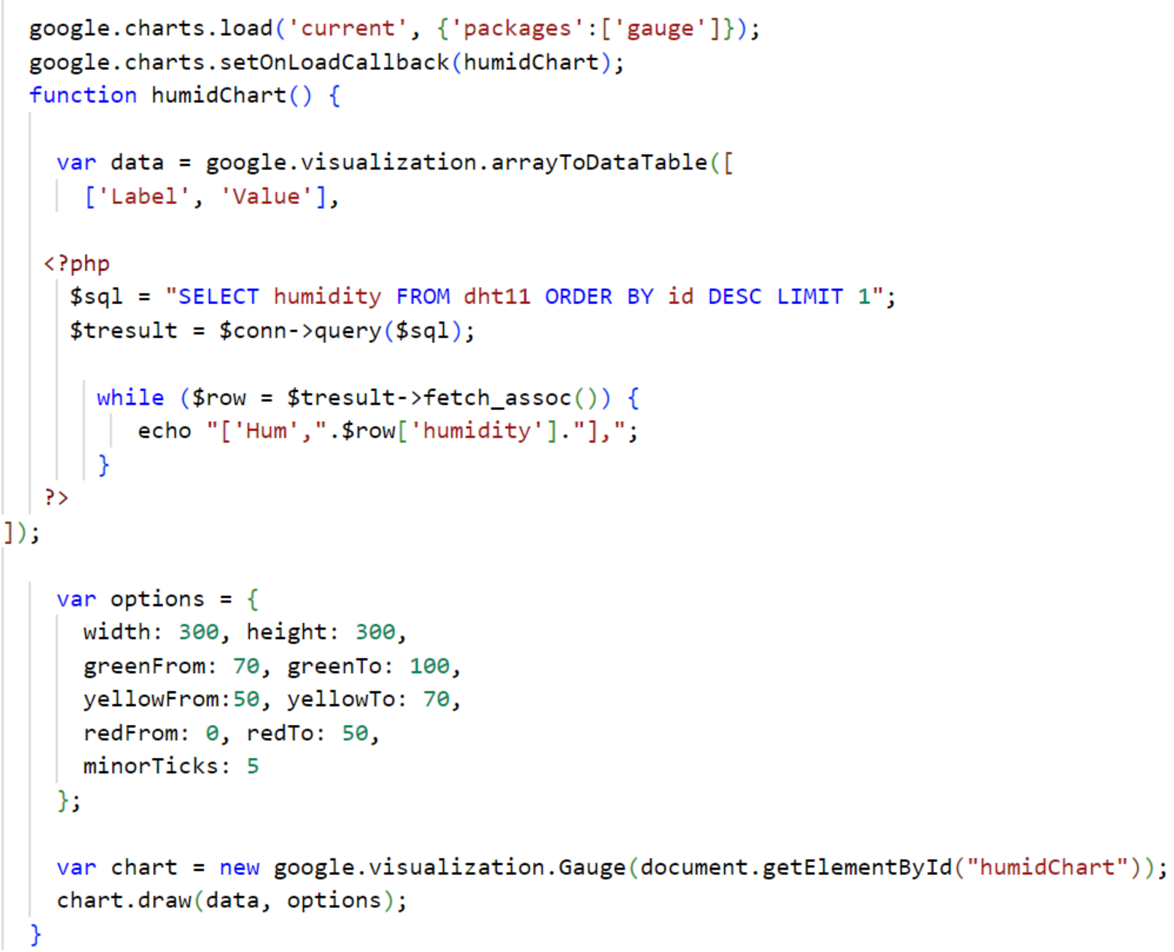


```
 if (client.connect(server, 80)) {
  Serial.println("Connected Plant A");
  Serial.print("Humidity =");
  client.print("GET http://localhost/psm/dht11.php?humidity=");     //YOUR URL
  Serial.println(humidityData);
  client.print(humidityData);
  client.print("&temperature=");
  Serial.print("Temperature =");
  client.print(temperatureData);
  Serial.println(temperatureData);
  Serial.print("Soil Moisture =");
  Serial.println(moisture);
  client.print("&moisture=");
  client.print(moisture);
  Serial.print("Status =");
  Serial.println(pp);
  client.print("&status=");
  client.print(pp);
  client.print(" ");      //SPACE BEFORE HTTP/1.1
  client.print("HTTP/1.1");
  client.println();
  client.println("Host: 192.168.241.115");
  client.println("Connection: close");
  client.println();
} else {
  Serial.println("connection failed");
```

Figure 3.29 Sending sensor data to database

3.4.4.5 Create gauges for data display

Figure 3.30 shows the code to create gauges to display temperature, humidity and soil moisture. The data are fetched from database using sql statement to display the readings.

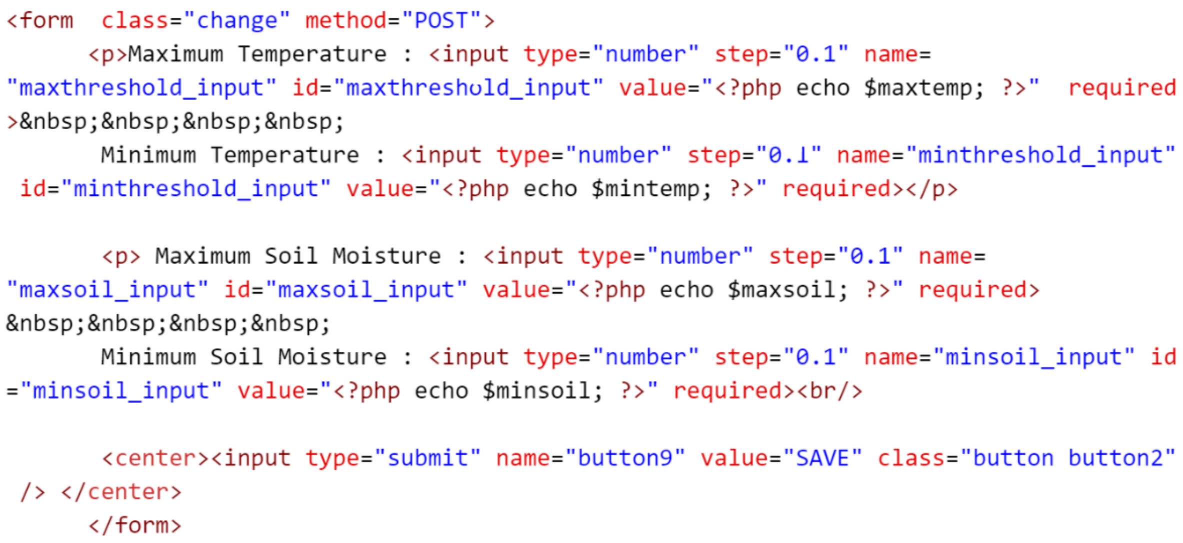


```
  google.charts.load('current', {'packages':['gauge']});
  google.charts.setOnLoadCallback(humidChart);
  function humidChart() {

    var data = google.visualization.arrayToDataTable([
      ['Label', 'Value'],

   <?php
     $sql = "SELECT humidity FROM dht11 ORDER BY id DESC LIMIT 1";
     $tresult = $conn->query($sql);

       while ($row = $tresult->fetch_assoc()) {
          echo "['Hum',".$row['humidity']."],";
       }
   ?>
]);

    var options = {
      width: 300, height: 300,
      greenFrom: 70, greenTo: 100,
      yellowFrom:50, yellowTo: 70,
      redFrom: 0, redTo: 50,
      minorTicks: 5
    };

    var chart = new google.visualization.Gauge(document.getElementById("humidChart"));
    chart.draw(data, options);
  }
```

Figure 3.30 Create gauge

3.4.4.6 Set temperature and soil moisture threshold

Figure 3.31 shows the code to create input field for temperature and soil moisture threshold. Values inserted are then stored in the database when saved.

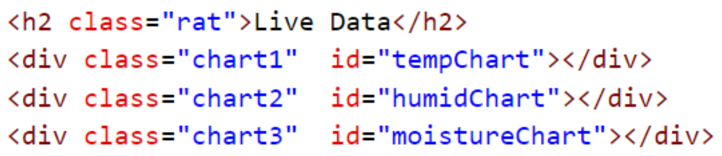


```
<form  class="change" method="POST">
      <p>Maximum Temperature : <input type="number" step="0.1" name=
"maxthreshold_input" id="maxthreshold_input" value="<?php echo $maxtemp; ?>"  required
>&nbsp;&nbsp;&nbsp;&nbsp;
       Minimum Temperature : <input type="number" step="0.1" name="minthreshold_input"
 id="minthreshold_input" value="<?php echo $mintemp; ?>" required></p>

       <p> Maximum Soil Moisture : <input type="number" step="0.1" name=
"maxsoil_input" id="maxsoil_input" value="<?php echo $maxsoil; ?>" required>
&nbsp;&nbsp;&nbsp;&nbsp;
       Minimum Soil Moisture : <input type="number" step="0.1" name="minsoil_input" id
="minsoil_input" value="<?php echo $minsoil; ?>" required><br/>

       <center><input type="submit" name="button9" value="SAVE" class="button button2"
 /> </center>
      </form>
```

Figure 3.31 Input temperature and soil moisture threshold

3.4.4.7 Display gauges for data display

Figure 3.32 shows the code to display the gauges created as shown in Figure 3.30.

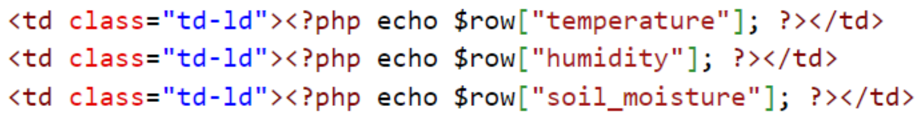


```
<h2 class="rat">Live Data</h2>
<div class="chart1"  id="tempChart"></div>
<div class="chart2"  id="humidChart"></div>
<div class="chart3"  id="moistureChart"></div>
```

Figure 3.32 Display gauge

3.4.4.8 Display live data

Figure 3.33 shows the code to display temperature, humidity and soil moisture readings below the gauges. These readings are fetched from the database.

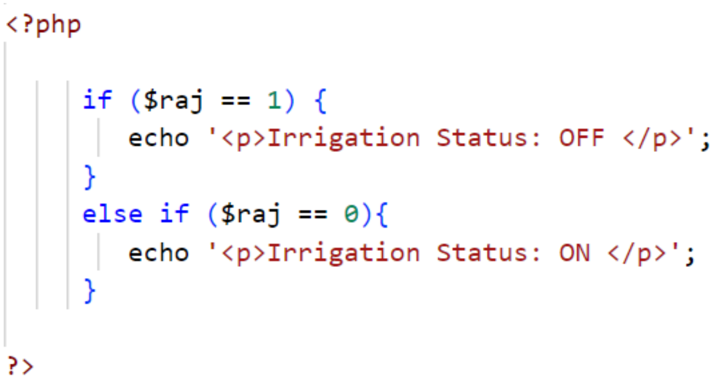


```
<td class="td-ld"><?php echo $row["temperature"]; ?></td>
<td class="td-ld"><?php echo $row["humidity"]; ?></td>
<td class="td-ld"><?php echo $row["soil_moisture"]; ?></td>
```

Figure 3.33 Display live data

3.4.4.9 Display irrigation status

Figure 3.34 shows the code to display irrigation status of water pump. The 1 and 0 values are fetched from the database.

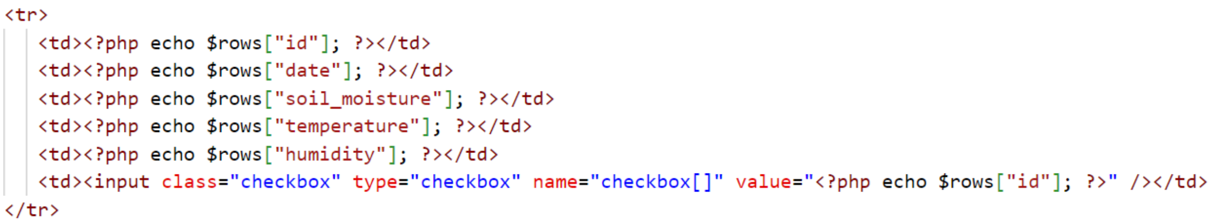


```
<?php

    if ($raj == 1) {
        echo '<p>Irrigation Status: OFF </p>';
    }
    else if ($raj == 0){
        echo '<p>Irrigation Status: ON </p>';
    }

?>
```

Figure 3.34 Display irrigation status

3.4.4.10 Display data in table

Figure 3.35 shows the code to display temperature, humidity and soil moisture readings in a table together with id and timestamp. There is also an option to delete any unwanted row by clicking the checkbox and delete button at the bottom.



```
<tr>
    <td><?php echo $rows["id"]; ?></td>
    <td><?php echo $rows["date"]; ?></td>
    <td><?php echo $rows["soil_moisture"]; ?></td>
    <td><?php echo $rows["temperature"]; ?></td>
    <td><?php echo $rows["humidity"]; ?></td>
    <td><input class="checkbox" type="checkbox" name="checkbox[]" value="<?php echo $rows["id"]; ?>" /></td>
</tr>
```

Figure 3.35 Display data in table

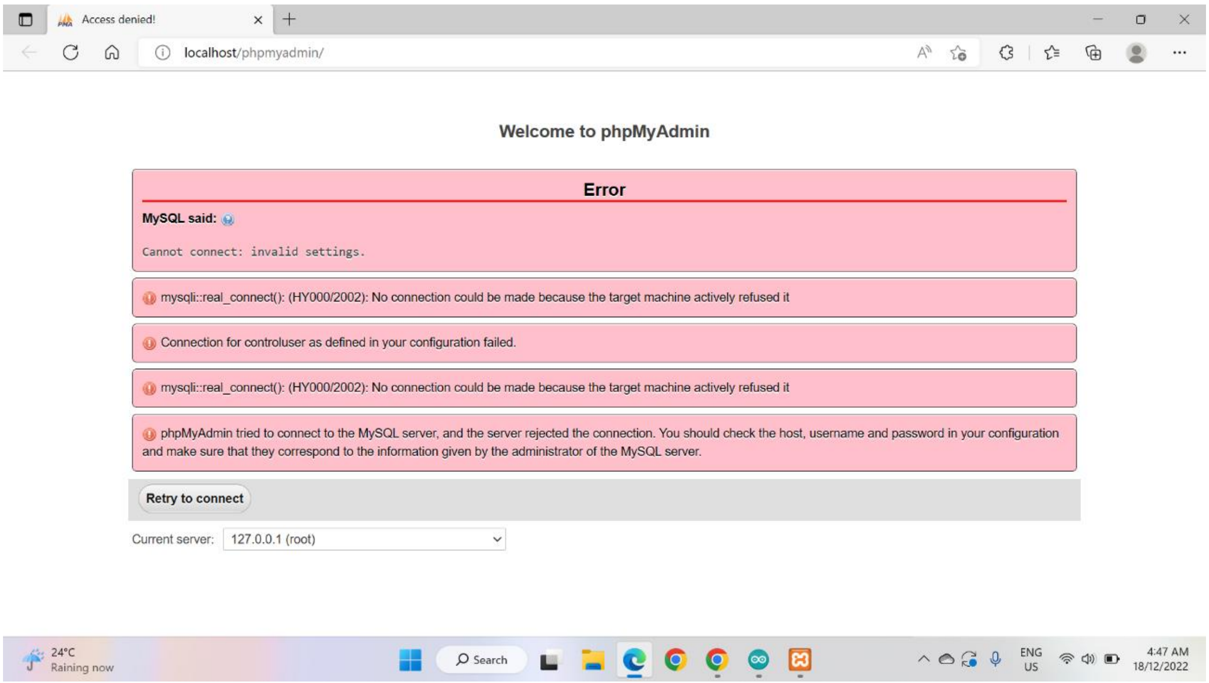
3.5 TESTING

Testing is divided into two which is development testing and final product testing. Development testing is where the system is tested each time during the development process whereas product testing is the testing of final product when the system deployed and entire project is completed.

3.5.1 Development Testing

There were some problems and errors faced during the development of this project. Errors and the way to solve it will be explained together with screeshots.

Figure 3.36 shows the error appeared when opening phpMyAdmin. Could not start both Apache and MySQL. This problem was solved by uninstalling and then reinstalling back xampp server.



Welcome to phpMyAdmin

Error

MySQL said:

```
Cannot connect: invalid settings.
```

mysqli::real_connect(): (HY000/2002): No connection could be made because the target machine actively refused it

Connection for controluser as defined in your configuration failed.

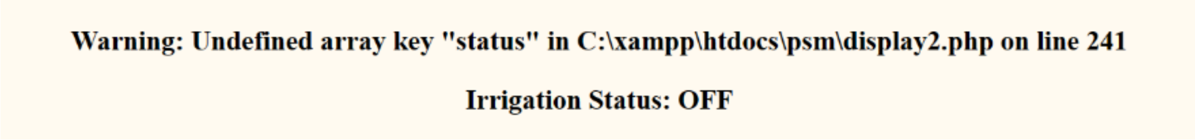
mysqli::real_connect(): (HY000/2002): No connection could be made because the target machine actively refused it

phpMyAdmin tried to connect to the MySQL server, and the server rejected the connection. You should check the host, username and password in your configuration and make sure that they correspond to the information given by the administrator of the MySQL server.

Current server: 127.0.0.1 (root)

Figure 3.36 Error opening phpMyAdmin

Figure 3.37 shows warning for undefined array key. This is because of the wrong declaration of variable. It was solved by declaring the variable inside the while loop.



```
Warning: Undefined array key "status" in C:\xampp\htdocs\psm\display2.php on line 241

Irrigation Status: OFF
```

Figure 3.37	Undefined array key error

3.5.2 Final product testing

After integrating and testing all the functionality, the system now will be deployed in real environment. The system will be tested for about five days time. The data gathered will be recorded every day. After the five days testing period, the recorded data will be analyzed to determine the correct rate of the system. The correct rate formula that will be used is as shown in Equation 3.**1**.

$$Correct\ rate = \frac{C}{A} \times 100$$
$$C - Number\ of\ correct\ data\ gathered$$
$$A - Total\ data\ gathered$$

Equation 3.1

3.6 GANTT CHART

Figure 3.38 shows the gannt chart for HPMS. In the gantt chart, all the phases from start till end of the project are shown in a timeline with duration.

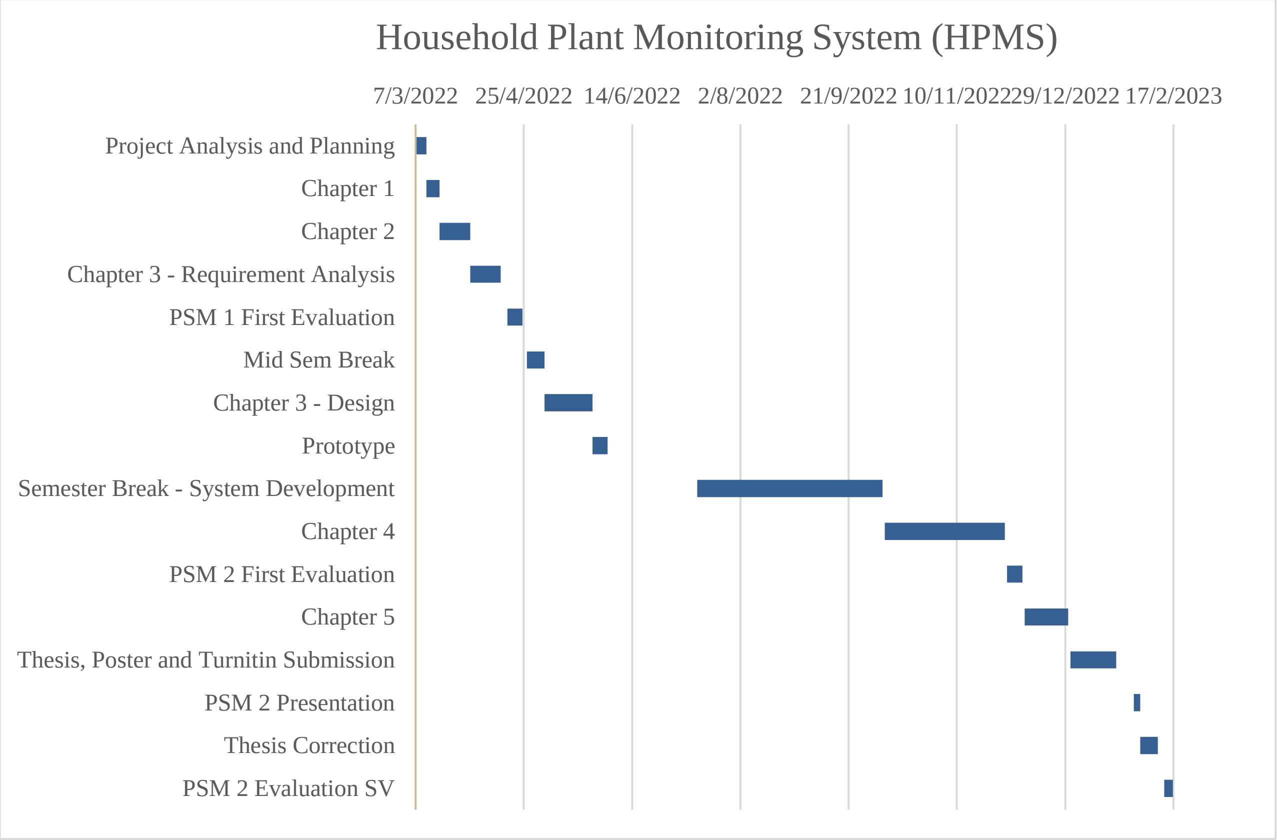


Figure 3.38 Gantt Chart for Household Plant Monitoring System

CHAPTER 4

RESULTS AND DISCUSSION

4.1 INTRODUCTION

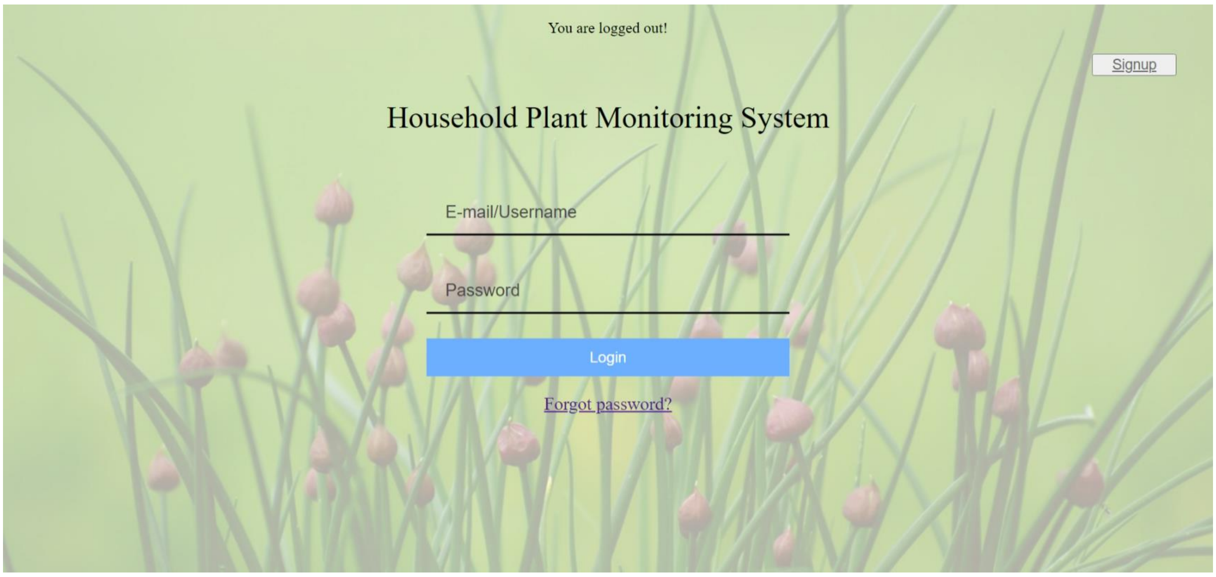
With the goals that were previously established in Chapter 1 as a guide, the results and discussion of the Household Plant Monitoring System are presented in this chapter. The operational prototype's screenshots of the implementation are shown, and this chapter will go through each interface's functionality. Additionally, the project's testing results will be discussed in the testing part. The Visual Studio Code, Arduino ide development platform and MySQL database are used to create the Household Plant Monitoring System.

4.2 RESULT OF HPMS DEVELOPMENT APPLICATION

Household Plant Monitoring System implementation will cover the steps in building and implementing various aspects of the project. A brief description of each user interface followed by screenshots.

4.2.1 Development result

Figure 4.1 shows login page of Household Plant Monitoring System. User can login into the system by using email/username and password.



You are logged out!

Household Plant Monitoring System

Figure 4.1 Login page of Household Plant Monitoring System

Figure 4.2 shows signup page of Household Plant Monitoring System. New user can create account by inputting username, password, repeat password and email address.

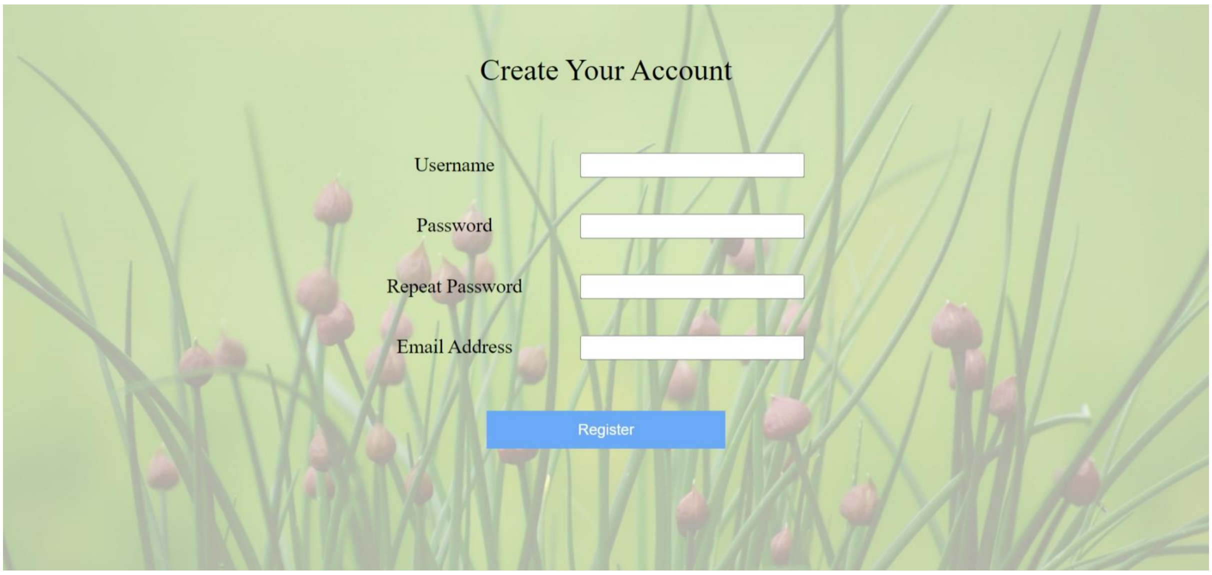
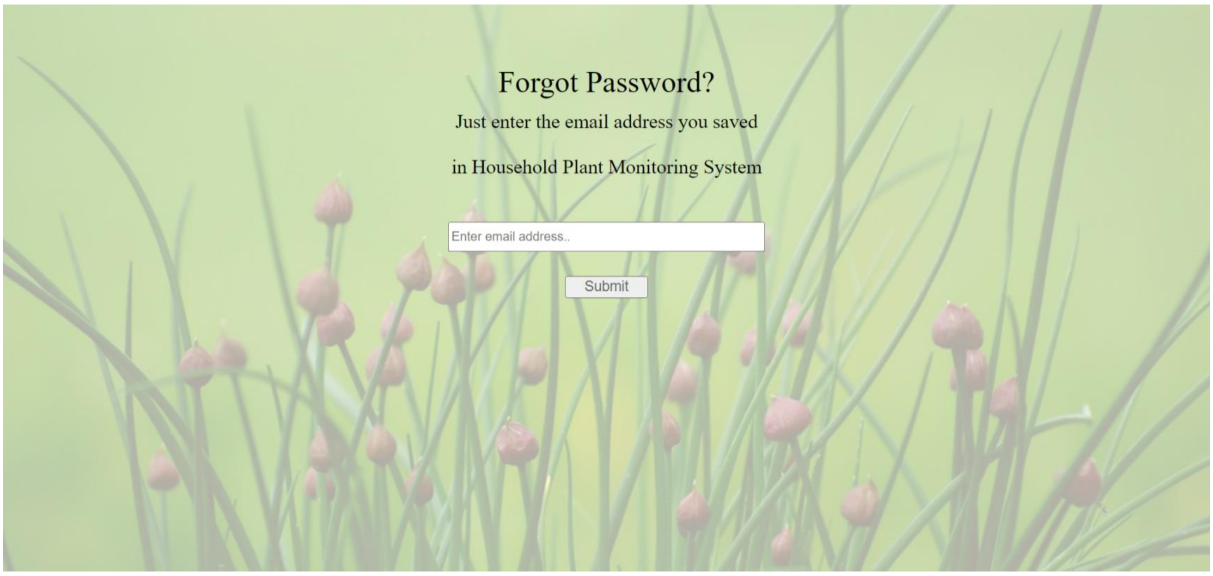

Create Your Account

Figure 4.2 Signup page of Household Plant Monitoring System

Figure 4.3 shows forgot password page of Household Plant Monitoring System. User need to enter email address that have been saved in their profile. A password reset link will be sent to the email address as shown in Figure 4.4.



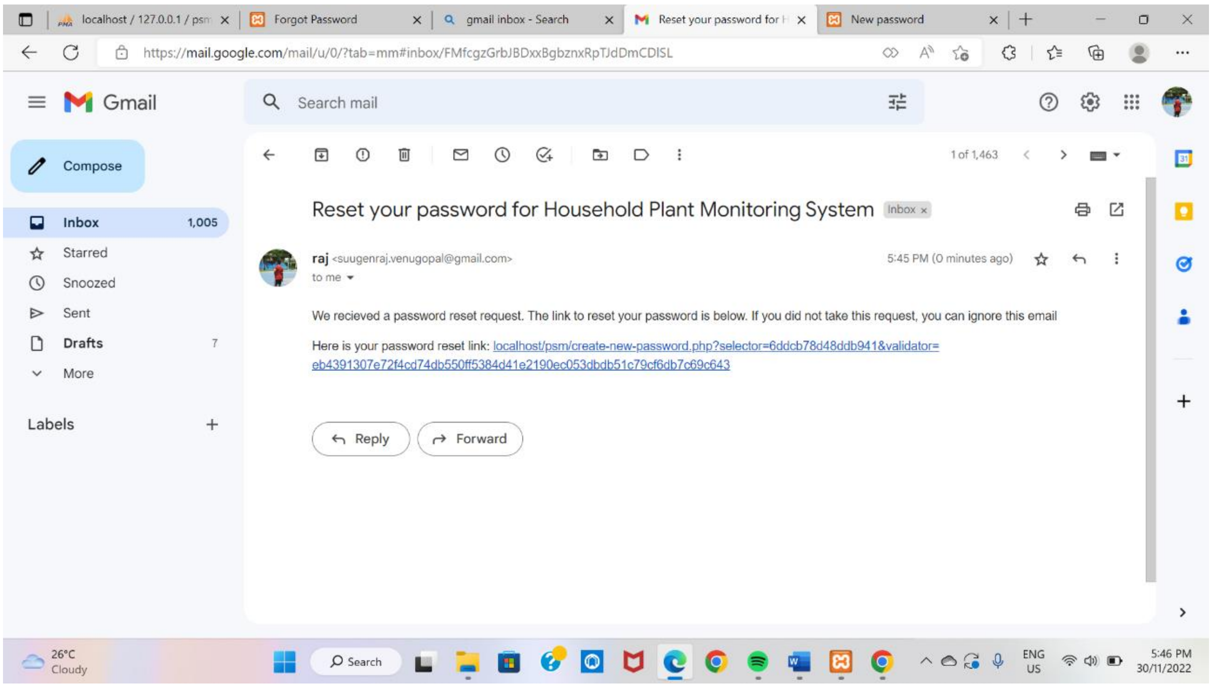
Forgot Password?

Just enter the email address you saved

in Household Plant Monitoring System

Figure 4.3 Forgot password page of Household Plant Monitoring System

Figure 4.4 shows gmail page with link for password reset. Once the link is clicked, it will redirect to reset password page as shown in Figure 4.**5**.



Reset your password for Household Plant Monitoring System

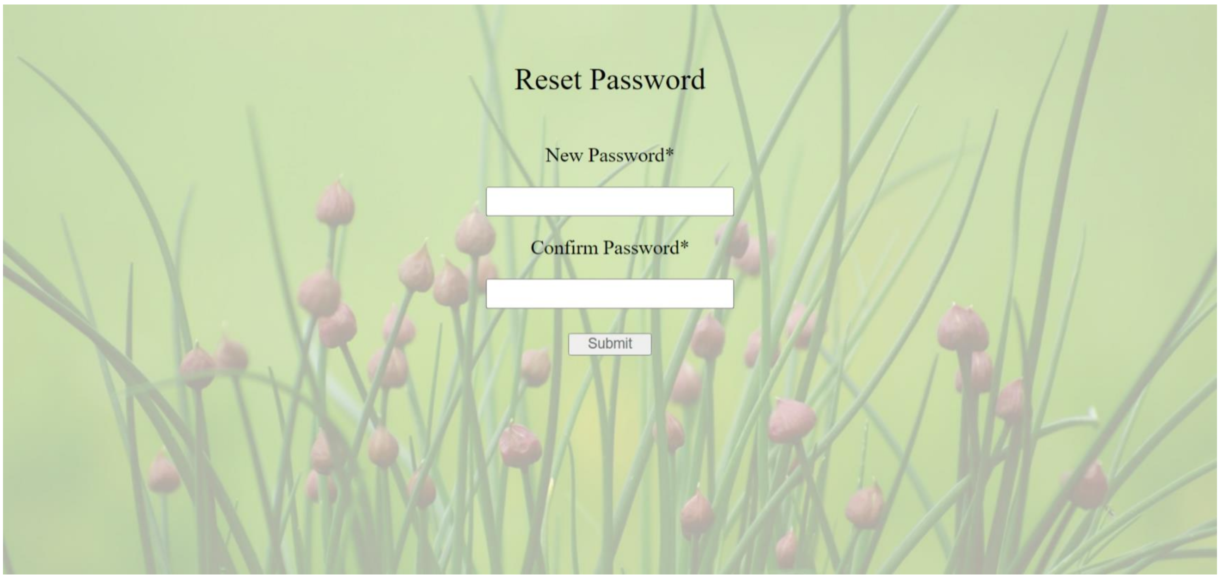
raj <suugenraj.venugopal@gmail.com>
to me
5:45 PM (0 minutes ago)

We recieved a password reset request. The link to reset your password is below. If you did not take this request, you can ignore this email

Here is your password reset link: localhost/psm/create-new-password.php?selector=6ddcb78d48ddb941&validator=eb4391307e72f4cd74db550ff5384d41e2190ec053dbdb51c79cf6db7c69c643

Figure 4.4 Gmail page with link for password reset

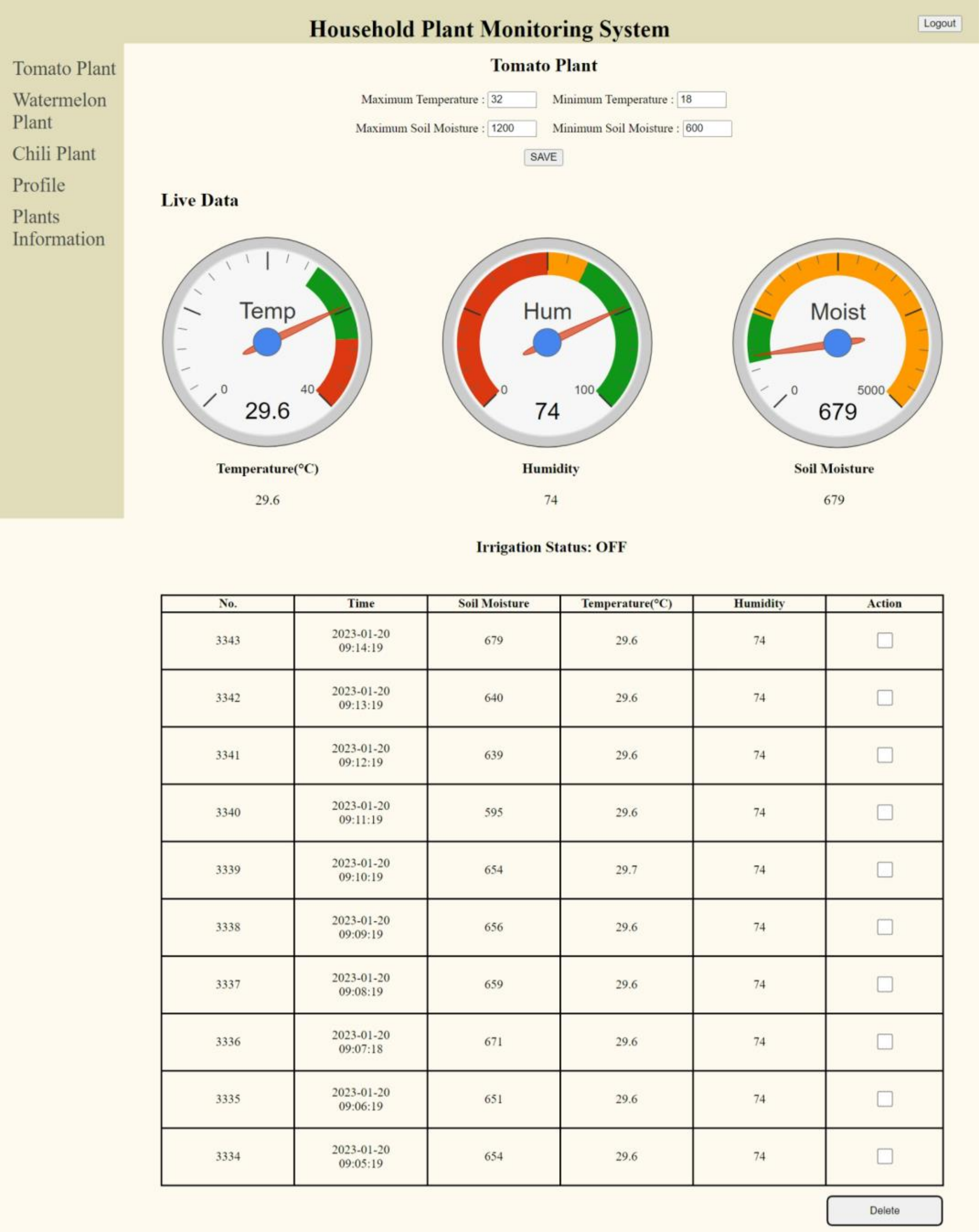
Figure 4.5 shows reset password page of Household Plant Monitoring System. User can enter a new password and repeat the password. This will then update the old password in user table in database according to user’s email.



Reset Password

Figure 4.5 Reset password page of Household Plant Monitoring System

Figure 4.6 shows Plant A page of Household Plant Monitoring System. On the left, there is sidebar where user can choose any tab to display. On top, user can set ideal temperature and soil moisture threshold value for tomato plant. Live data of temperature, humidity and soil moisture of tomato plant is displayed in gauges. Below that there is water pump irrigation status of tomato plant. Lastly, table at the bottom contains information retrieved from database such as number, time, soil moisture, temperature and humidity. There is also checkbox option for each row to delete any unwanted data.



Household Plant Monitoring System

Tomato Plant

Maximum Temperature : 32 Minimum Temperature : 18

Maximum Soil Moisture : 1200 Minimum Soil Moisture : 600

Live Data

Temperature(°C)
29.6

Humidity
74

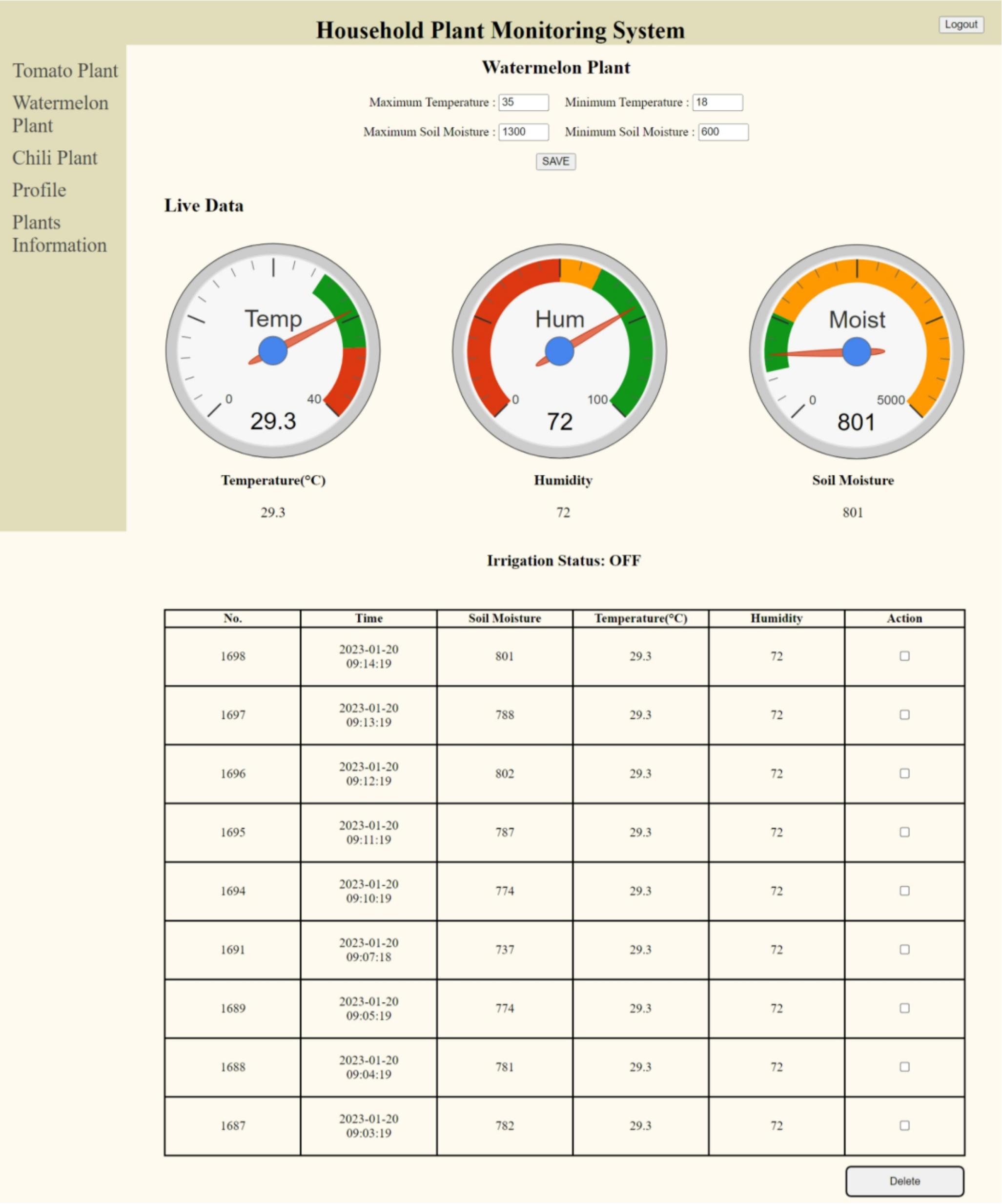
Soil Moisture
679

Irrigation Status: OFF

No.	Time	Soil Moisture	Temperature(°C)	Humidity	Action
3343	2023-01-20 09:14:19	679	29.6	74	□
3342	2023-01-20 09:13:19	640	29.6	74	□
3341	2023-01-20 09:12:19	639	29.6	74	□
3340	2023-01-20 09:11:19	595	29.6	74	□
3339	2023-01-20 09:10:19	654	29.7	74	□
3338	2023-01-20 09:09:19	656	29.6	74	□
3337	2023-01-20 09:08:19	659	29.6	74	□
3336	2023-01-20 09:07:18	671	29.6	74	□
3335	2023-01-20 09:06:19	651	29.6	74	□
3334	2023-01-20 09:05:19	654	29.6	74	□

Figure 4.6 Plant A page of Household Plant Monitoring System

Figure 4.7 shows Plant B page of Household Plant Monitoring System. On the left, there is sidebar where user can choose any tab to display. On top, user can set ideal temperature and soil moisture threshold value for watermelon plant. Live data of temperature, humidity and soil moisture of watermelon plant is displayed in gauges. Below that there is water pump irrigation status of watermelon plant. Lastly, table at the bottom contains information retrieved from database such as number, time, soil moisture, temperature and humidity. There is also checkbox option for each row to delete any unwanted data.



Household Plant Monitoring System

Watermelon Plant

Maximum Temperature : 35 Minimum Temperature : 18

Maximum Soil Moisture : 1300 Minimum Soil Moisture : 600

Live Data

Temperature(°C) 29.3

Humidity 72

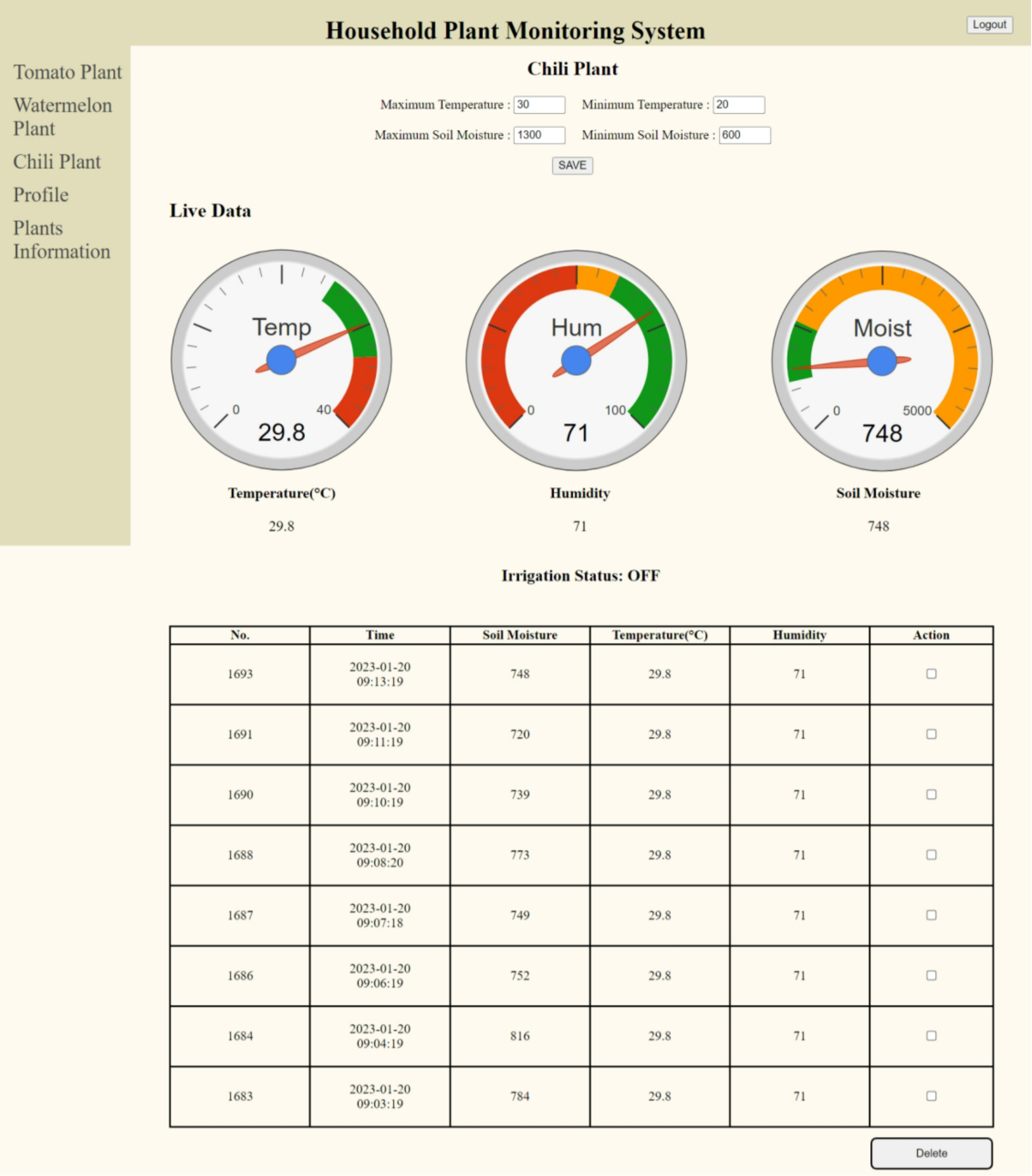
Soil Moisture 801

Irrigation Status: OFF

No.	Time	Soil Moisture	Temperature(°C)	Humidity	Action
1698	2023-01-20 09:14:19	801	29.3	72	□
1697	2023-01-20 09:13:19	788	29.3	72	□
1696	2023-01-20 09:12:19	802	29.3	72	□
1695	2023-01-20 09:11:19	787	29.3	72	□
1694	2023-01-20 09:10:19	774	29.3	72	□
1691	2023-01-20 09:07:18	737	29.3	72	□
1689	2023-01-20 09:05:19	774	29.3	72	□
1688	2023-01-20 09:04:19	781	29.3	72	□
1687	2023-01-20 09:03:19	782	29.3	72	□

Figure 4.7 Plant B page of Household Plant Monitoring System

Figure 4.8 shows Plant C page of Household Plant Monitoring System. On the left, there is sidebar where user can choose any tab to display. On top, user can set ideal temperature and soil moisture threshold value for chili plant. Live data of temperature, humidity and soil moisture of chili plant is displayed in gauges. Below that there is water pump irrigation status of chili plant. Lastly, table at the bottom contains information retrieved from database such as number, time, soil moisture, temperature and humidity. There is also checkbox option for each row to delete any unwanted data.



Household Plant Monitoring System

Logout

Tomato Plant

Watermelon Plant

Chili Plant

Profile

Plants Information

Chili Plant

Maximum Temperature : 30 Minimum Temperature : 20

Maximum Soil Moisture : 1300 Minimum Soil Moisture : 600

SAVE

Live Data

Temperature(°C) 29.8

Humidity 71

Soil Moisture 748

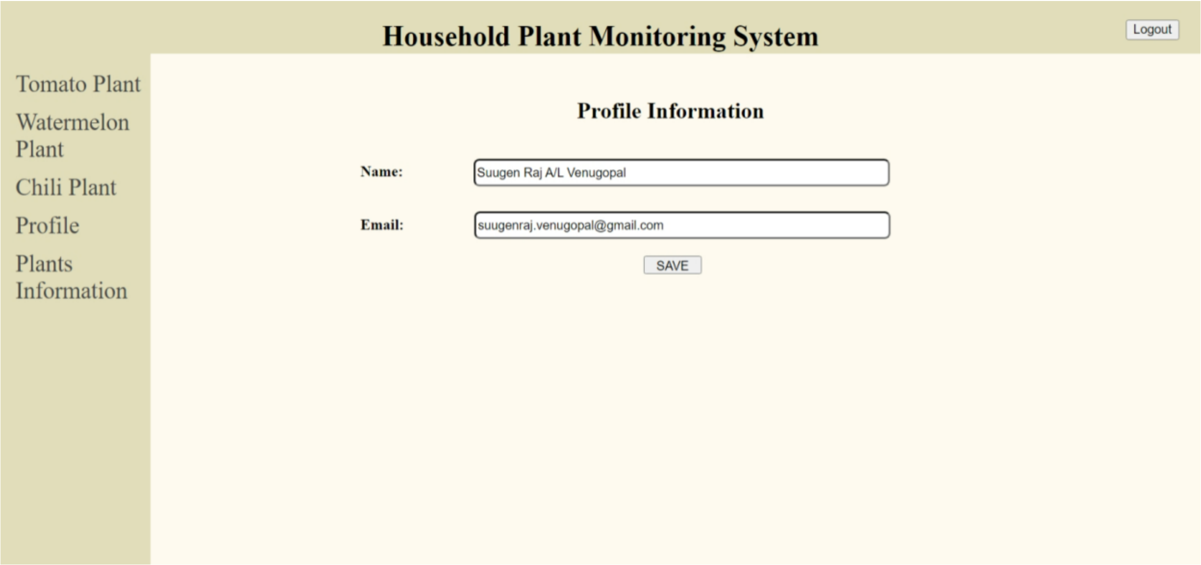
Irrigation Status: OFF

No.	Time	Soil Moisture	Temperature(°C)	Humidity	Action
1693	2023-01-20 09:13:19	748	29.8	71	□
1691	2023-01-20 09:11:19	720	29.8	71	□
1690	2023-01-20 09:10:19	739	29.8	71	□
1688	2023-01-20 09:08:20	773	29.8	71	□
1687	2023-01-20 09:07:18	749	29.8	71	□
1686	2023-01-20 09:06:19	752	29.8	71	□
1684	2023-01-20 09:04:19	816	29.8	71	□
1683	2023-01-20 09:03:19	784	29.8	71	□

Delete

Figure 4.8 Plant C page of Household Plant Monitoring System

Figure 4.9 shows profile information page of Household Plant Monitoring System. User can edit and save their name and email address.



Household Plant Monitoring System

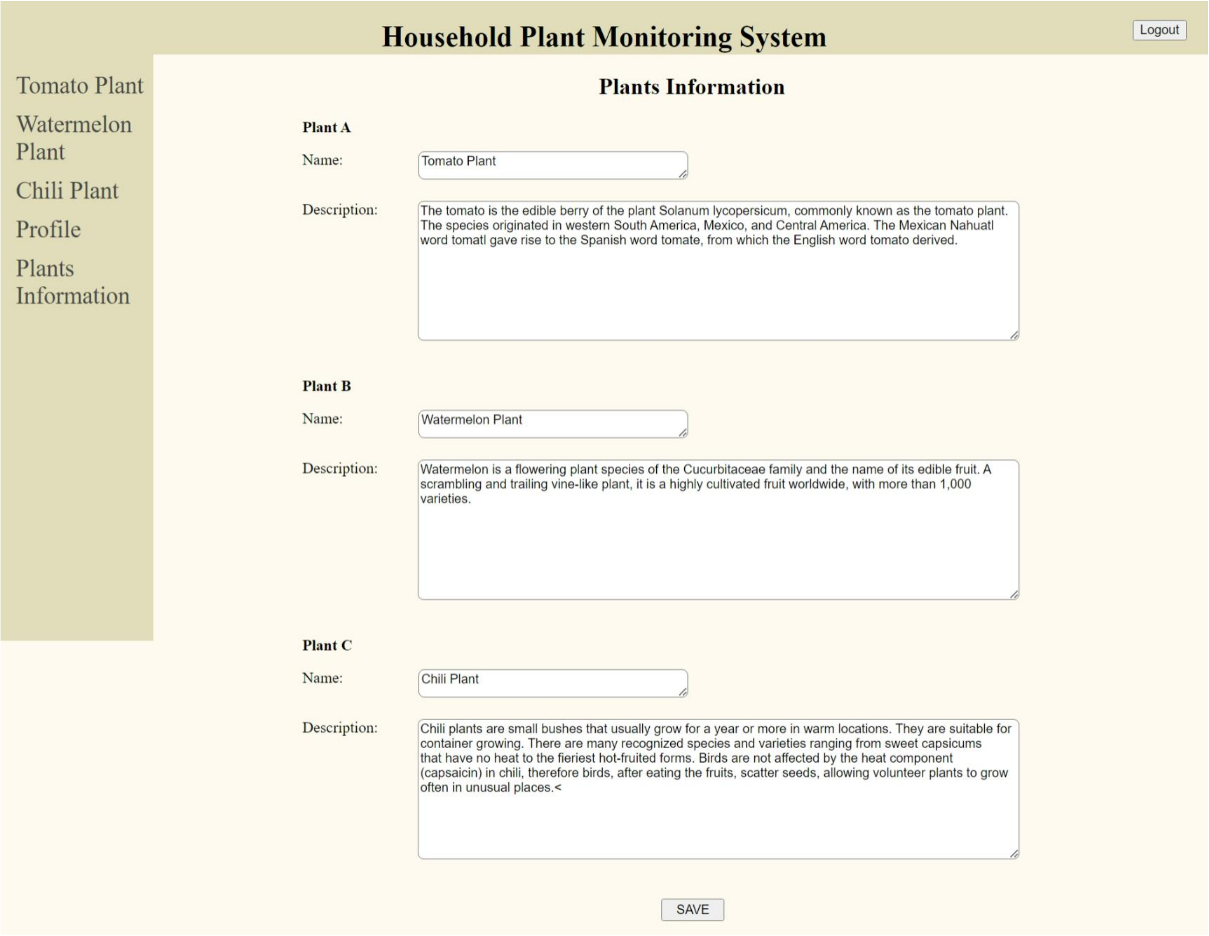
Profile Information

Name: Suugen Raj A/L Venugopal

Email: suugenraj.venugopal@gmail.com

Figure 4.9 Profile information page of Household Plant Monitoring System

Figure 4.10 shows plants information page of Household Plant Monitoring System. User can edit and save plants name and description for own reference.



Household Plant Monitoring System

Plants Information

Plant A

Name: Tomato Plant

Description: The tomato is the edible berry of the plant Solanum lycopersicum, commonly known as the tomato plant. The species originated in western South America, Mexico, and Central America. The Mexican Nahuatl word tomatl gave rise to the Spanish word tomate, from which the English word tomato derived.

Plant B

Name: Watermelon Plant

Description: Watermelon is a flowering plant species of the Cucurbitaceae family and the name of its edible fruit. A scrambling and trailing vine-like plant, it is a highly cultivated fruit worldwide, with more than 1,000 varieties.

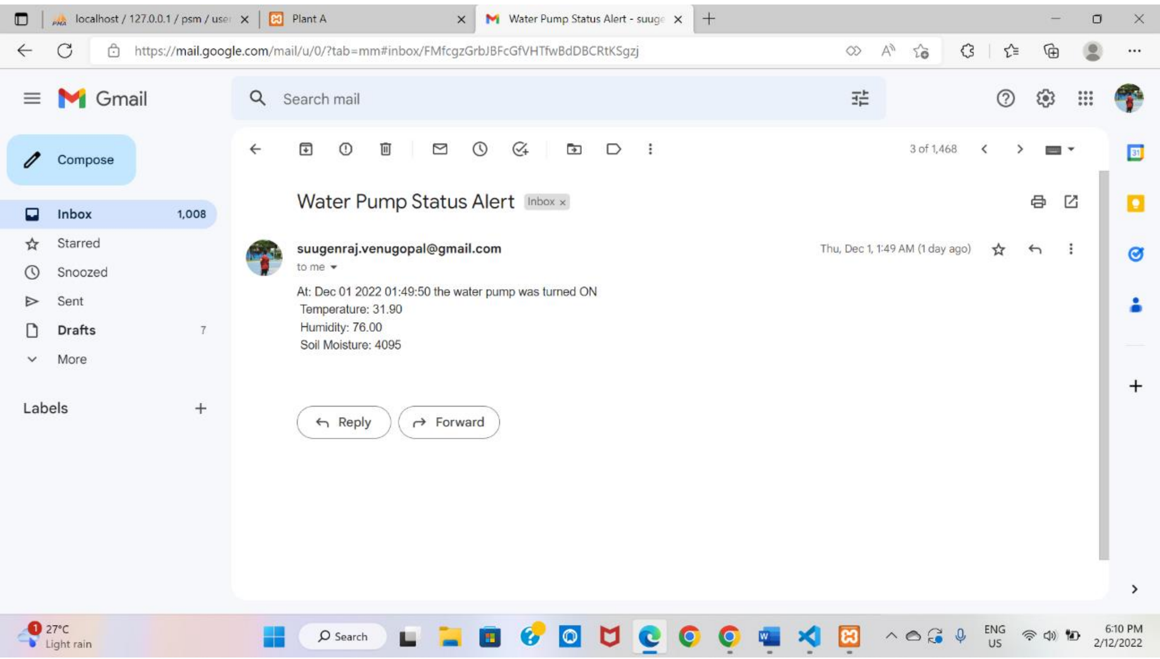
Plant C

Name: Chili Plant

Description: Chili plants are small bushes that usually grow for a year or more in warm locations. They are suitable for container growing. There are many recognized species and varieties ranging from sweet capsicums that have no heat to the fieriest hot-fruited forms. Birds are not affected by the heat component (capsaicin) in chili, therefore birds, after eating the fruits, scatter seeds, allowing volunteer plants to grow often in unusual places.<

Figure 4.10 Plants information page of Household Plant Monitoring System

Figure 4.11 shows email alert. Whenever the water pump is turned ON, an email will be sent to the user. Email alert will contain the timestamp, together with temperature, humidity and soil moisture data when water pump is turned ON.



Water Pump Status Alert

suugenraj.venugopal@gmail.com

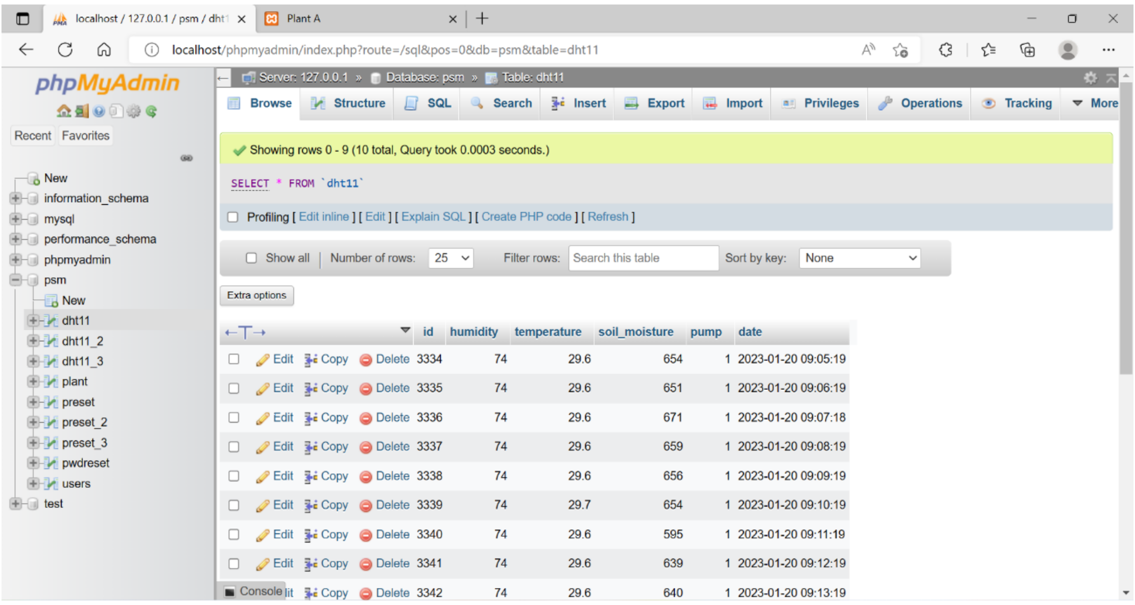
to me

Thu, Dec 1, 1:49 AM (1 day ago)

At: Dec 01 2022 01:49:50 the water pump was turned ON
Temperature: 31.90
Humidity: 76.00
Soil Moisture: 4095

Figure 4.11 Email Alert

Figure 4.12 shows database table of sensors data. Humidity, temperature, soil moisture and water pump status data are stored together with id and timestamp. These data are retrieved from Arduino ide when the serial monitor is opened and starts to gather readings from sensors.



✓ Showing rows 0 - 9 (10 total, Query took 0.0003 seconds.)

```
SELECT * FROM `dht11`
```

□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

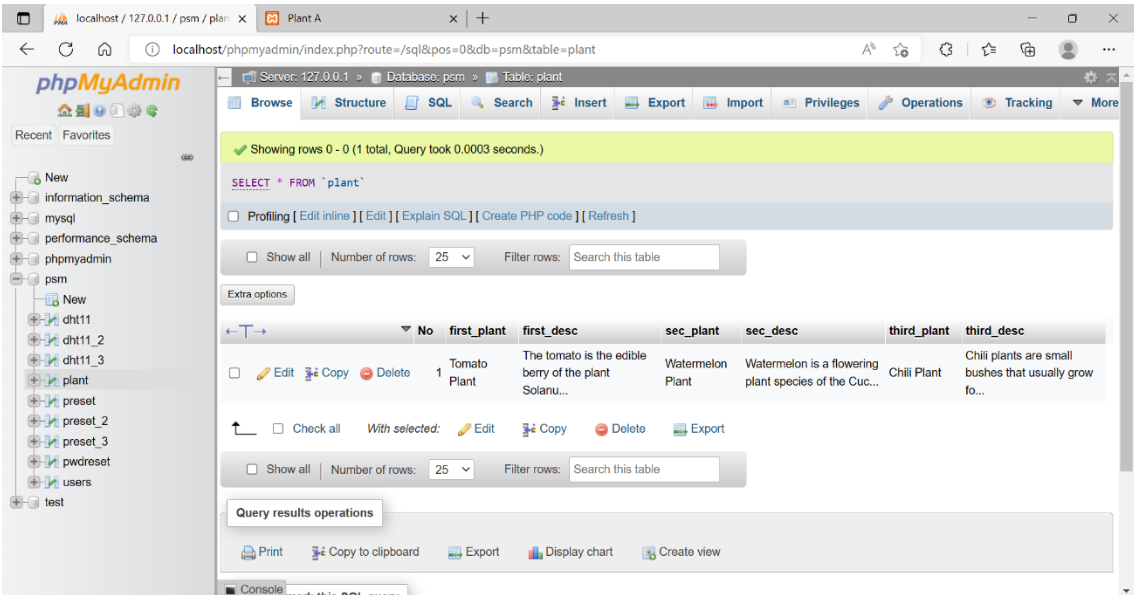
□ Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

Extra options

		id	humidity	temperature	soil_moisture	pump	date
□	Edit Copy Delete	3334	74	29.6	654	1	2023-01-20 09:05:19
□	Edit Copy Delete	3335	74	29.6	651	1	2023-01-20 09:06:19
□	Edit Copy Delete	3336	74	29.6	671	1	2023-01-20 09:07:18
□	Edit Copy Delete	3337	74	29.6	659	1	2023-01-20 09:08:19
□	Edit Copy Delete	3338	74	29.6	656	1	2023-01-20 09:09:19
□	Edit Copy Delete	3339	74	29.7	654	1	2023-01-20 09:10:19
□	Edit Copy Delete	3340	74	29.6	595	1	2023-01-20 09:11:19
□	Edit Copy Delete	3341	74	29.6	639	1	2023-01-20 09:12:19
	lit Copy Delete	3342	74	29.6	640	1	2023-01-20 09:13:19

Figure 4.12 Database table of sensors data

Figure 4.13 shows database table of plants information. This information can be edited and saved by user for their own reference in the web page Plants Information. Example of information are plants name and the description of each plant.



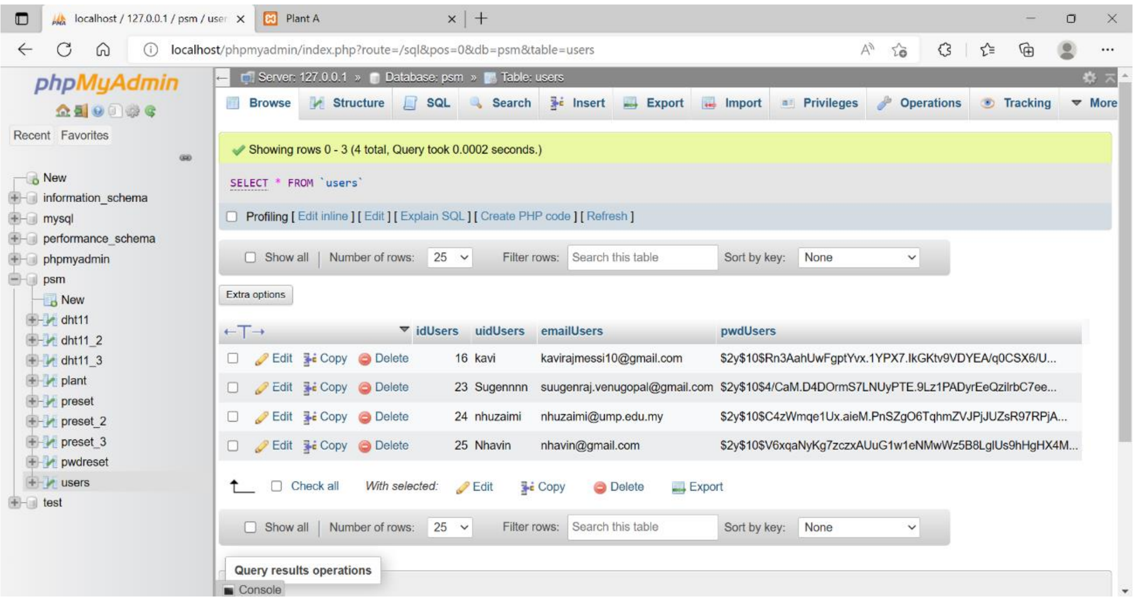
Showing rows 0 - 0 (1 total, Query took 0.0003 seconds.)

```
SELECT * FROM `plant`
```

	No	first_plant	first_desc	sec_plant	sec_desc	third_plant	third_desc
□ Edit Copy Delete	1	Tomato Plant	The tomato is the edible berry of the plant Solanu...	Watermelon Plant	Watermelon is a flowering plant species of the Cuc...	Chili Plant	Chili plants are small bushes that usually grow fo...

Figure 4.13 Database table of plants information

Figure 4.14 shows database table of user information. There are two more columns which are username (uidUsers) and password (pwdUsers). Username, password and email address information are stored when a user create new account. Passwords are encrypted to provide privacy for users. User can update their username and email in the profile tab of HPMS web page.



Showing rows 0 - 3 (4 total, Query took 0.0002 seconds.)

```
SELECT * FROM `users`
```

idUsers	uidUsers	emailUsers	pwdUsers
16	kavi	kavirajmessi10@gmail.com	$2y$10$Rn3AahUwFgptYvx.1YPX7.IkGKtv9VDYEA/q0CSX6/U...
23	Sugennnn	suugenraj.venugopal@gmail.com	$2y$10$4/CaM.D4DOrmS7LNUyPTE.9Lz1PADyrEeQzilrbC7ee...
24	nhuzaimi	nhuzaimi@ump.edu.my	$2y$10$C4zWmqe1Ux.aieM.PnSZgO6TqhmZVJPjJUZsR97RPjA...
25	Nhavin	nhavin@gmail.com	$2y$10$V6xqaNyKg7zczxAUuG1w1eNMwWz5B8LglUs9hHgHX4M...

Figure 4.14 Database table of user information

4.2.2 Household Plant Monitoring System (HPMS) device development

Figure 4.15 shows the complete device development of HPMS. All the components used are shown and labelled. Each plant has its own set of components like DHT11 sensor, soil moisture sensor, relay module, water pump, battery holder and breadboard. All these are connected to ESP32 microcontroller and external power supply.

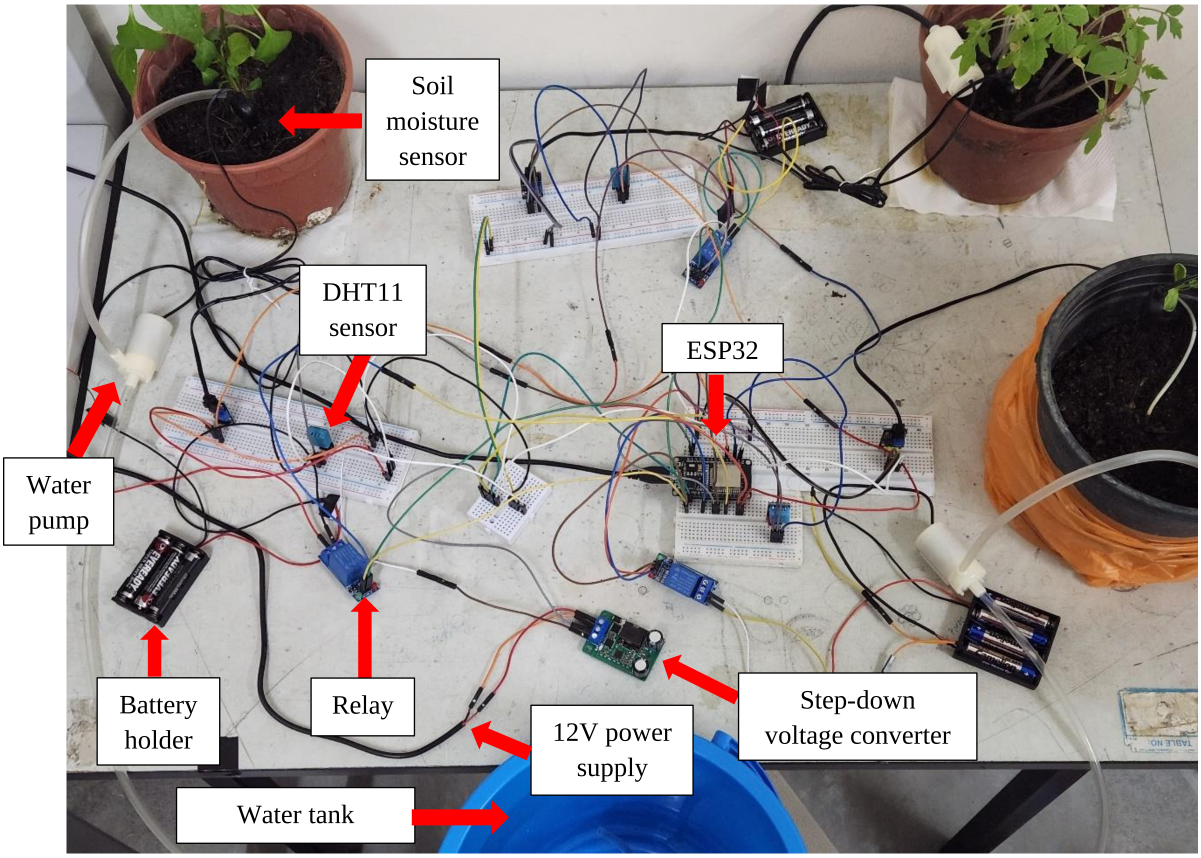


Figure 4.15 Complete HPMS device development

4.3 RESULT OF HPMS VALIDATION

Household Plant Monitoring System has been validated for five days by carrying out one testing each day. This validation is to ensure the efficiency of the system. The testing is carried out once a day for around one hour, where data were gathered for every one minute. All the three plants are tested together simultaneously but using their own set of components assigned. After five days of testing, the results gathered are validated using the formula.

4.3.1 Tomato plant testing and validation result

Table 4.1 shows the number of correct data gathered and total number of data gathered during each testing for tomato plant.

Table 4.1 Tomato plant testing result

Testing	Number of correct data gathered	Total data gathered
1	37	37
2	35	49
3	56	59
4	61	61
5	49	57

Table 4.2 shows the output obtained and the the effiency calculation for each testing of tomato plant using the Equation 3.1.

Table 4.2 Tomato plant validation result

Testing	Discussion	Efficiency
1	Water pump remained off as the temperature and soil moisture were in range.	100%
2	Water pump remained off as the temperature and soil moisture were in range.	71%
3	Water pump remained off as the temperature and soil moisture were in range.	94%
4	Water pump turned on for 5 seconds when soil moisture went above the range.	100%
5	Water pump remained off as the temperature and soil moisture were in range.	85%

4.3.2 Watermelon plant testing and validation result

Table 4.3 shows the number of correct data gathered and total number of data gathered during each testing for watermelon plant.

Table 4.3 Watermelon plant testing result

Testing	Number of correct data gathered	Total data gathered
1	36	49
2	41	56
3	50	60
4	42	61
5	39	52

Table 4.4 shows the output obtained and the the effiency calculation for each testing of watermelon plant using the Equation 3.1.

Table 4.4 Watermelon plant validation result

Testing	Discussion	Efficiency
1	Water pump turned on for 5 seconds when soil moisture went above the range.	73%
2	Water pump remained off as the temperature and soil moisture were in range.	73%
3	Water pump turned on for 5 seconds when soil moisture went above the range.	85%
4	Water pump remained off as the temperature and soil moisture were in range.	68%
5	Water pump remained off as the temperature and soil moisture were in range.	75%

4.3.3 Chili plant testing and validation result

Table 4.5 shows the number of correct data gathered and total number of data gathered during each testing for chili plant.

Table 4.5 Chili plant testing result

Testing	Number of correct data gathered	Total data gathered
1	42	48
2	46	54
3	49	61
4	50	60
5	49	58

Table 4.6 shows the output obtained and the the effiency calculation for each testing of chili plant using the Equation 3.1.

Table 4.6 Chili plant validation result

Testing	Discussion	Efficiency
1	Water pump turned on for 5 seconds when soil moisture went above the range.	87%
2	Water pump remained off as the temperature and soil moisture were in range.	85%
3	Water pump remained off as the temperature and soil moisture were in range.	80%
4	Water pump turned on for 5 seconds when soil moisture went above the range.	83%
5	Water pump remained off as the temperature and soil moisture were in range.	84%

4.3.4 System efficiency comparison of three plants

Figure 4.16 shows the system’s efficiency comparison for all three plants. The efficiency rate calculated for each plant are taken from Table 4.2, Table 4.4, Table 4.6 and converted into a graph. On the y-axis is the percentage (%) and on the x-axis is the testing (days).

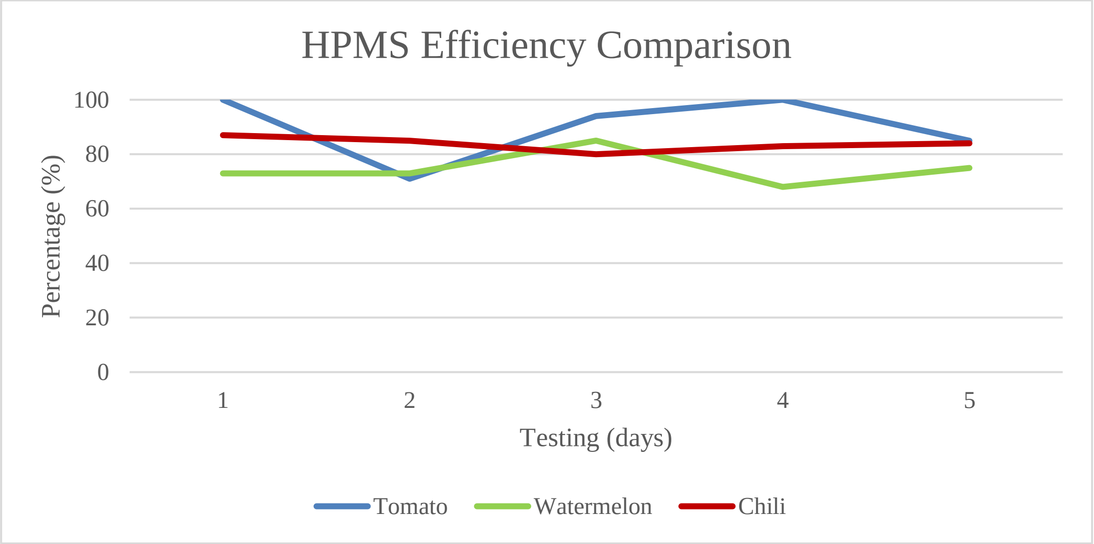


Figure 4.16 HPMS Efficiency Comparison

CHAPTER 5

CONCLUSION

Household Plant Monitoring System (HPMS) can gather data of soil moisture, temperature and humidity and display on the web application. At the same time, the data gets stored in the database. The system operates automatically for the irrigation process. User will be sent notifications via email about the irrigation status. The main and common problem is the lack of plant monitoring by users especially when they're away. Besides, most of the existing systems are not able to monitor more than one plant on the same system and watering the plants manually is not very systematic as excess water might be supplied. HPMS is able to solve all of the problems mentioned above.

One of the features of the system included is that the threshold value for the soil moisture and temperature can be set. Users will also be able to see the irrigation status of the system. For this project, the SDLC model used is incremental model. In the first increment, component connection will be done. In the second increment, setting up of database and then in the third increment, setting up of the web application.

In designing HPMS, it is achieved by designing the component connection and web page interface design as shown in chapter 3. For the development of HPMS, entire system which includes the component connection, database and web page interface as shown in chapter 4 was completely developed. To validate the system, the efficiency was calculated based on the readings taken. The average system efficiency for tomato plant is 90%, watermelon plant 74% and chili plant 83%. Data got displayed on the web application, got stored in database and system irrigated water automatically. This concludes that all the objectives listed for this project have been achieved successfully.

Some of the components will not work properly at times is the limitation for this project. Sometimes, the sensor will not collect any reading.In the future, better components can be used with an extra budget so that it will work properly at all times, which will increase its efficiency. With an improved efficiency, this system can be implemented for bigger plantation projects rather than using it only for house plants.

REFERENCES

[1] A. Ashraf Ahmad Shaharudin, “Protecting the Agriculture Sector During the Covid-19 Crisis.” [Online]. Available: www.KRInstitute.org.

[2] A. Zulaikha Mohd Zaki *et al.*, “Journal of Sustainable Natural Resources Building a Smart Gardening System and Plant Monitoring Using IoT,” *J. Sustain. Nat. Resour.*, vol. 2, no. 1, pp. 1–6, 2021, [Online]. Available: http://publisher.uthm.edu.my/ojs/index.php/jsunrhttp://publisher.uthm.edu.my/jsunr

[3] M. Ariff Bin Azlan, “SMART IRRIGATION SYSTEM,” 2018.

[4] C. S. Agriculture, “Small scale agriculture Labour-saving techniques Advantages of small scale agriculture”.

[5] D. H. Galhena, R. Freed, and K. M. Maredia, “Promising Aproach,” *BioMed Cent.*, pp. 1–13, 2013.

[6] A. Chadha, S. K. Florentine, B. S. Chauhan, B. Long, and M. Jayasundera, “Influence of soil moisture regimes on growth, photosynthetic capacity, leaf biochemistry and reproductive capabilities of the invasive agronomic weed; Lactuca serriola,” *PLoS One*, vol. 14, no. 6, Oct. 2018, doi: 10.1371/journal.pone.0218191.

[7] D. E. D. G. Pollo, “Automatic Watering Device for Tomato Using Soil Moisture Sensor,” *First Int. Conf. Exhib. Sci. Technol.*, no. 25th-27th October, pp. 152–160, 2018.

[8] R. R. Shamshiri, J. W. Jones, K. R. Thorp, D. Ahmad, H. C. Man, and S. Taheri, “Review of optimum temperature, humidity, and vapour pressure deficit for microclimate evaluation and control in greenhouse cultivation of tomato: A review,” *Int. Agrophysics*, vol. 32, no. 2, pp. 287–302, 2018, doi: 10.1515/intag-2017-0005.

[9] N. America, N. Zealand, and A. S. Service, “Watermelon Production This publication was developed by the Small-scale and Part-time Farming Project at Penn State with support from the,” 2009.

[10] A. Hidayat, Nasrullah, D. Putra, and Ramiati, “Temperature and soil control design with fuzzy method in greenhouse for cabe seeding,” *Int. J. Informatics*

Vis., vol. 3, no. 3, pp. 243–246, 2019, doi: 10.30630/joiv.3.3.168.

[11] M. G. Usman, M. Y. Rafii, M. R. Ismail, M. A. Malek, and M. Abdul Latif, "Heritability and genetic advance among chili pepper genotypes for heat tolerance and morphophysiological characteristics," *Sci. World J.*, vol. 2014, 2014, doi: 10.1155/2014/308042.

[12] S. Aisyah *et al.*, "Remote Watering System for Home Garden," 2015.

[13] M. Manoj, V. Sai, K. Subba Rao, N. V. Krishna, and M. V. Lakshmi, "Iot Based Smart Agriculture," 2018.

[14] "Smart Irrigation System," 2017.

[15] Y. Fahim and T. Sarkar, "A Project Report On IoT based SMART FARMING SYSTEM CERTIFICATE OF APPROVAL Countersigned by IoT based Smart Farming System View project," 2018. [Online]. Available: https://www.researchgate.net/publication/334131097

[16] C. K. Chian, U. Tunku, and A. Rahman, "INTERNET OF THINGS FOR INDOOR FARMING A REPORT SUBMITTED TO," 2018.

[17] P. R. Satao, S. A. Chavan, R. B. Kale, R. S. Patil, P. R. Chaugule, and, "IOT BASED AUTOMATIC IRRIGATION SYSTEM", doi: 10.51397/OAIJSE05.2021.0056.

[18] R. A. Rozario P and A. BanuS, "Tomen:A Plant monitoring and smart gardening system using IoT." [Online]. Available: http://www.ijpam.eu

[19] S. Sudarshan Gowda, D. R. Students, and J. H. Kumar B Asst Professor, "Automated Plant Watering System." [Online]. Available: www.ijert.org

[20] K. KANNAN*, N. S. KUMAR, E. LOGITH, R. M. KUMAR, and O. S. PRAKASH, "An Internet of Things Based Smart Irrigation using Solenoid Valve," *Int. J. Recent Technol. Eng.*, vol. 9, no. 1, pp. 2018–2023, May 2020, doi: 10.35940/ijrte.A1979.059120.

[21] Amna Zulqadar, "SDLC Waterfall Model: The 6 phases you need to know about," *Rezaid*, pp. 4–11, 2019, [Online]. Available: https://rezaid.co.uk/sdlc-waterfall-model/

[22] A. Dziuba, “Agile Software Development Lifecycle Phases Explained,” *Relevant*, pp. 1–24, 2021, [Online]. Available: https://relevant.software/blog/agile-software-development-lifecycle-phases-explained/

[23] Lakshay Sharma, “What is Incremental Model or Iterative Model,” pp. 1–8, 2016, [Online]. Available: https://www.toolsqa.com/software-testing/sdlc-incremental-iterative-model/

APPENDIX A

USER MANUAL

TABLE OF CONTENTS

COMPONENTS IN HPMS

Table 1.1 shows the entire list of components used in HPMS together with quantity of each component.

Table 1.1 List of components included in HPMS

Components	Quantity
ESP32 Microcontroller	1
DHT11 Sensor	3
Soil Moisture Sensor	3
Relay Module	3
Water pump	3
Male to Male Jumper Wires	1 set
Male to Female Jumper Wires	1 set
Large breadboard	3
Battery holder	3
Tube	3
Step-Down Voltage Regulator	1

COMPONENT CONNECTION HPMS

Figure 1.1 shows component connection of HPMS. The labelled numbers are elaborated step by step to give detail and clear explanation to connect the components.

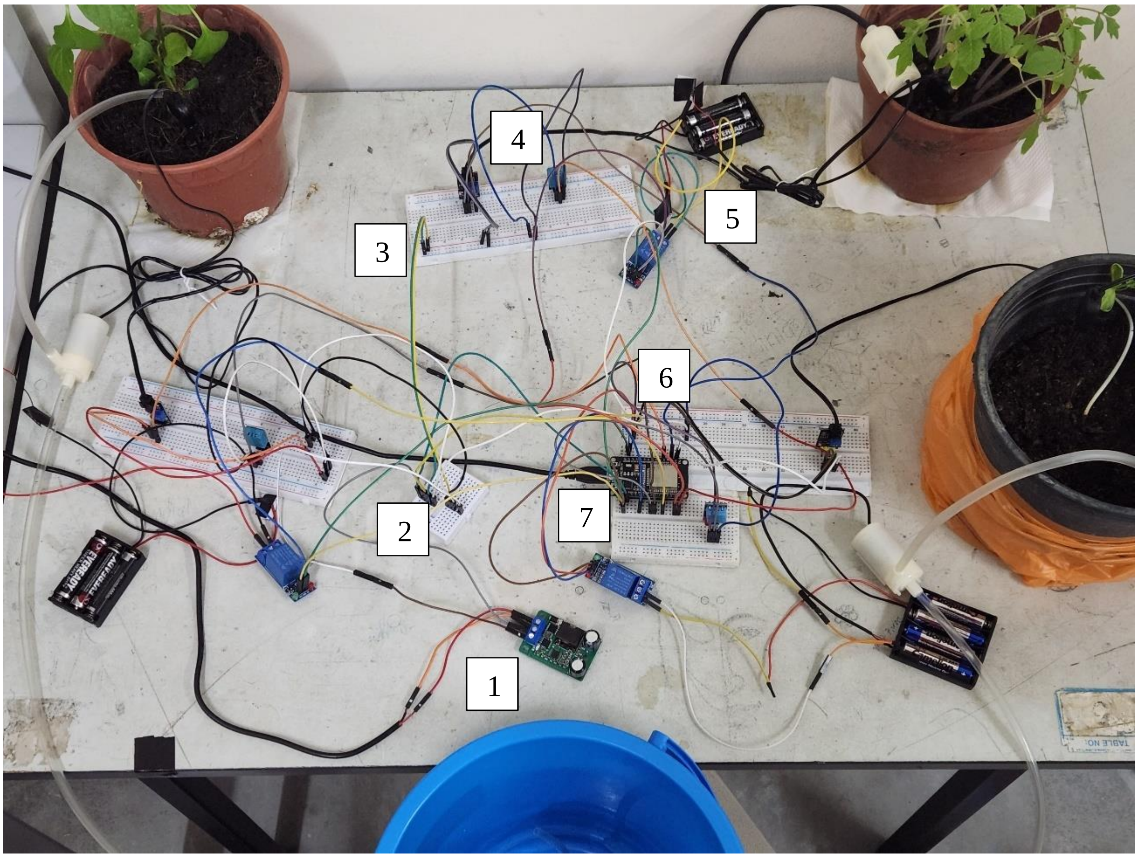


Figure 1.1 Labelled component connection of HPMS

1. Take 12V adapter, connect the red wire to the IN port and the black wire to the GND port of step-down voltage converter.

2. Take a jumper wire and connect one end of it to the OUT port of step-down voltage converter. Connect the other end to positive rail in a small breadboard. Use another jumper wire and connect the GND port of step-down voltage converter with negative rail in the small breadboard. Connect ESP32 GND pin to the negative rail in the small breadboard.

3. Take two jumper wires, connect one from positive rail of small breadboard to positive rail of large breadboard. Connect the other one from negative rail of small breadboard to negative rail of large breadboard. Repeat these steps for the other two large breadboards.

4. Insert soil moisture sensor and DHT11 sensor to the large breadboard. Connect the positive and negative pins of the sensors to positive and negative rails in the same breadboard. Repeat these steps for the other two large breadboards.

5. Connect the VCC and GND pin of relay module to 3.3V and GND pin of ESP32 respectively. Then, connect NO port of relay to the red wire of battery holder and COM port to the red wire of water pump. Connect the black wire of water pump and battery holder together. Repeat these steps to the other two relay modules.

6. Now connect the data pin of DHT11 sensor and analog output pin of soil moisture sensor to input pins of ESP32. Connect the IN pin of relay module to output pin of ESP32. Repeat these steps for the other sensors and relay module.

7. Lastly, power up the ESP32 microcontroller using USB cable by connecting to laptop's USB port. Switch on the 12V adapter. The component connection is complete if the sensors light up.

HPMS SYSTEM USAGE

1. After completing the component connection, user can open the Arduino code.

2. User must make sure the connected Wi-Fi had been declared in the Arduino code.

8. Go to tools and make sure that the ‘DOIT ESP32 DEVKIT V1’ board and port selected is correct.

9. Click on “Verify” icon in Arduino ide to see if there’s any error in the code. User can see “Done compiling” at the bottom right once it’s done.

10. Now, click on “Upload” icon in Arduino ide to upload the code to ESP32. User can see “Done uploading” at the bottom once it’s done.

11. Click the Serial Monitor icon on the top right to see the sensor readings.

12. Login to the HPMS webpage to see the current temperature, humidity and soil moisture readings on the gauges. Below the gauge there’s irrigation status of water pump. Lastly, user can see all the readings in a table according to the time interval set. On the sidebar, click on other plants to see their readings.