REAL TIME SYSTEM AND INTERNET OF THINGS FINAL PROJECT REPORT
DEPARTMENT OF ELECTRICAL ENGINEERING
UNIVERSITAS INDONESIA

Remote Agriculture System

GROUP B8

Muhammad Farrel Mirawan	**2106731554**
Jeremy Ganda Pandapotan	**2106731573**
Handaneswari Pramudhyta Imanda	**2106731346**
Sulthan Satrya Yudha Darmawan	**2106731560**

PREFACE

In recent years, there has been a growing interest in leveraging technology to monitor and control various aspects of our daily lives. One domain where this technological advancement is gaining prominence is in the realm of agriculture, presenting an opportunity to enhance and optimize crop yields consistently.

The ESP32 microcontroller, with its robust capabilities, emerges as a versatile platform for developing projects aimed at revolutionizing agricultural practices. This project, titled "Remote Agriculture System," focuses on facilitating remote monitoring and management of agricultural environments. By utilizing sensors like YL39 for soil moisture, DHT11 for temperature and humidity, and an LDR for light intensity, the system aims to provide comprehensive insights into the conditions affecting plant health.

The Remote Agriculture System employs multiple sensors to gauge crucial environmental parameters. The YL39 sensor measures soil moisture, the DHT11 monitors air temperature and humidity, while the LDR assesses light levels. These data points collectively contribute to a holistic understanding of the plant's well-being. Additionally, the system incorporates 2 LEDs as indicators, visually representing the environmental quality ranging from optimal to critical conditions. And 1 water pump to help the supply of water, in the case the plant soil is dry.

We extend our gratitude to our lecturer, F. Astha Ekadiyanto S.T, M.Sc., for invaluable guidance throughout this project. Special thanks also go to the Lab Assistant from the Digital Laboratory for providing us with the opportunity to apply our acquired knowledge in the realization of this impactful project.

Depok, December 08, 2023

Group B8

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

1.1 PROBLEM STATEMENT

In the world of farming, a big challenge is the lack of effective tools to keep a close eye on and manage critical things like soil moisture, temperature, humidity, and light for crops. These factors are essential for good crop growth, but farmers often lack quick and accurate information about them. This becomes more challenging with changing and unpredictable weather patterns.

Farmers currently struggle to know the right time to water crops, how to protect them from pests, and how to manage everything efficiently. The absence of a reliable monitoring system leads to the inefficient use of resources such as water and energy, reducing the effectiveness of farming practices. Additionally, modern farmers are increasingly asked to be environmentally friendly, using fewer resources and taking care of the environment. The existing tools are not well-equipped to meet these new demands.

Therefore, there is a genuine need for an advanced system that provides farmers with real-time information about their crops. This system should be user-friendly and assist farmers in making informed decisions to protect their crops and adopt sustainable practices. The proposed solution is the "Remote Agriculture System," utilizing a smart device called the ESP32 and various sensors to give farmers a comprehensive understanding of their crops, making farming more intelligent and resource-efficient.

1.2 PROPOSED SOLUTION

The proposed solution is the implementation of the "Remote Agriculture System" utilizing the ESP32 microcontroller and a suite of sensors. This innovative system aims to revolutionize agricultural practices by providing farmers with real-time data on key environmental factors affecting plant growth. The integration of YL39 soil moisture sensors, DHT11 temperature and humidity sensors, and an LDR light intensity sensor offers a comprehensive set of parameters for monitoring and managing crop conditions.

The Remote Agriculture System leverages the ESP32's robust capabilities to collect and transmit data wirelessly, enabling remote monitoring from any location with internet connectivity. The YL39 sensor measures soil moisture levels, allowing farmers to optimize irrigation schedules and prevent both under and over-watering. The DHT11 sensor provides information on air temperature and humidity, crucial for understanding the overall climate conditions influencing plant health. The LDR sensor contributes by assessing light levels, ensuring that plants receive the optimal amount of sunlight.

To enhance user-friendliness and immediacy of information, the system incorporates visual indicators with 2 LEDs, representing environmental quality from optimal to critical conditions. Additionally, a water pump is integrated into the system to automatically supply water when soil moisture levels indicate dry conditions.

The "Remote Agriculture System" thus presents a comprehensive and technologically advanced solution to empower farmers with real-time insights, enabling them to make informed decisions, optimize resource usage, and ultimately improve crop yield in an increasingly dynamic agricultural landscape.

1.3 ACCEPTANCE CRITERIA

The acceptance criteria of this project are as follows:

1. Objective 1: Data Acquisition and Monitoring
   The system must successfully retrieve real-time data, including temperature, humidity, soil moisture, and soil temperature, using DHT11, YL39 soil moisture sensor, and other specified sensors.
2. Objective 2: Environmental Monitoring
   The system must effectively monitor and report changes in temperature, humidity, soil moisture, and soil temperature in real time.
3. Objective 3: Sensor Integration
   The system must seamlessly integrate the specified sensors, including DHT11, YL39 soil moisture sensor, and LDR, to ensure comprehensive environmental monitoring.
4. Objective 4: Actuation System

The system must incorporate an actuation mechanism to automatically activate the water pump when soil moisture falls below a predefined threshold. The actuation system should respond promptly to changes in soil moisture levels and ensure proper irrigation.

5. Objective 5: Cloud Connectivity

   The system must establish a reliable connection to an IoT cloud platform such as Blynk for real-time data transmission and remote monitoring. Data sent to the cloud should include temperature, humidity, soil moisture, and soil temperature information in a structured and secure manner.

6. Objective 6: User Interface

   The system must provide a user-friendly interface for farmers to monitor environmental conditions, set threshold values, and configure monitoring parameters. The user interface should be accessible from both desktop and mobile devices.

1.4 ROLES AND RESPONSIBILITIES

The roles and responsibilities assigned to the group members are as follows:

Roles	Responsibilities	Person
Software Developer	Develop algorithms for data processing, actuation triggers, and communication with IoT cloud platforms.	Muhammad Farrel Mirawan, Handaneswari Pramudhyta Imanda, Sulthan Satrya Yudha Darmawan, Jeremy Ganda Pandapotan
Hardware Engineer	Integrate sensors (DHT11, YL39, LDR), water pump, and LED strip into a cohesive system.	Handaneswari Pramudhyta Imanda, Muhammad Farrel Mirawan,

Field Tester	Conduct field testing to validate the system's performance under varying environmental conditions. Gather feedback from testing regarding system usability, reliability, and effectiveness.	Handaneswari Pramudhyta Imanda, Jeremy Ganda Pandapotan, Sulthan Satrya Yudha Darmawan, Muhammad Farrel Mirawan,

Table 1. Roles and Responsibilities

1.5 TIMELINE AND MILESTONES

Number	Task Title	November											Desember									
		20	21	22	23	24	25	26	27	28	29	30	1	2	3	4	5	6	7	8	9	10
1	HARDWARE DESIGN																					
1.1	Planning and Preperation																					
1.2	Gathering Components																					
1.3	Component Testing																					
1.4	Assembly																					
2	SOFTWARE DEVELOPMENT																					
2.1	Making the flowchart																					
2.2	Coding for YL39																					
2.3	Coding for DHT11																					
2.4	Coding for LDR																					
2.5	Connecting to Blynk																					
2.6	Putting the code together																					
3	INTEGRATION AND TESTING OF HARDWARE AND SOFTWARE																					
3.1	Software and Hardware Integration																					
3.2	Trial and error																					
3.3	Raw testing																					
3.4	Troubleshooting																					
4	FINAL PRODUCT ASSEMBLY AND TESTING																					
4.1	Final Assembly																					
4.2	Final testing																					
4.3	Finished Product																					
4.4	Finished Report																					

Table 2. Gantt Chart

CHAPTER 2

IMPLEMENTATION

2.1 HARDWARE DESIGN AND SCHEMATIC

The Remote Agriculture System is an IoT project designed to enhance agricultural practices by providing real-time monitoring and automation using Blynk. The hardware design incorporates key components, including the ESP32 microcontroller for seamless connectivity, the YL39 Soil Moisture Sensor to gauge soil moisture levels, the DHT11 Temperature Sensor for monitoring ambient temperature, and an LDR (Light Dependent Resistor) for detecting ambient light conditions. Additionally, an LED is integrated to illuminate the agricultural area in low light conditions, and a water pump is employed to irrigate the soil when moisture levels are below the desired threshold. The schematic ensures efficient communication between the components, enabling the ESP32 to receive data from the sensors and trigger actions based on predefined conditions. If the LDR detects darkness, the LED will be activated to provide artificial light. Simultaneously, if the soil moisture sensor indicates dry conditions, the water pump will be activated to irrigate the soil, ensuring optimal conditions for plant growth.

So the components that will be used in this project are :

- 1 ESP32
- 1 YL39
- 1 DHT11
- 1 LDR
- 2 LED
- 1 Water Pump
- 20 Jumper Cable
- 1 Protoboard

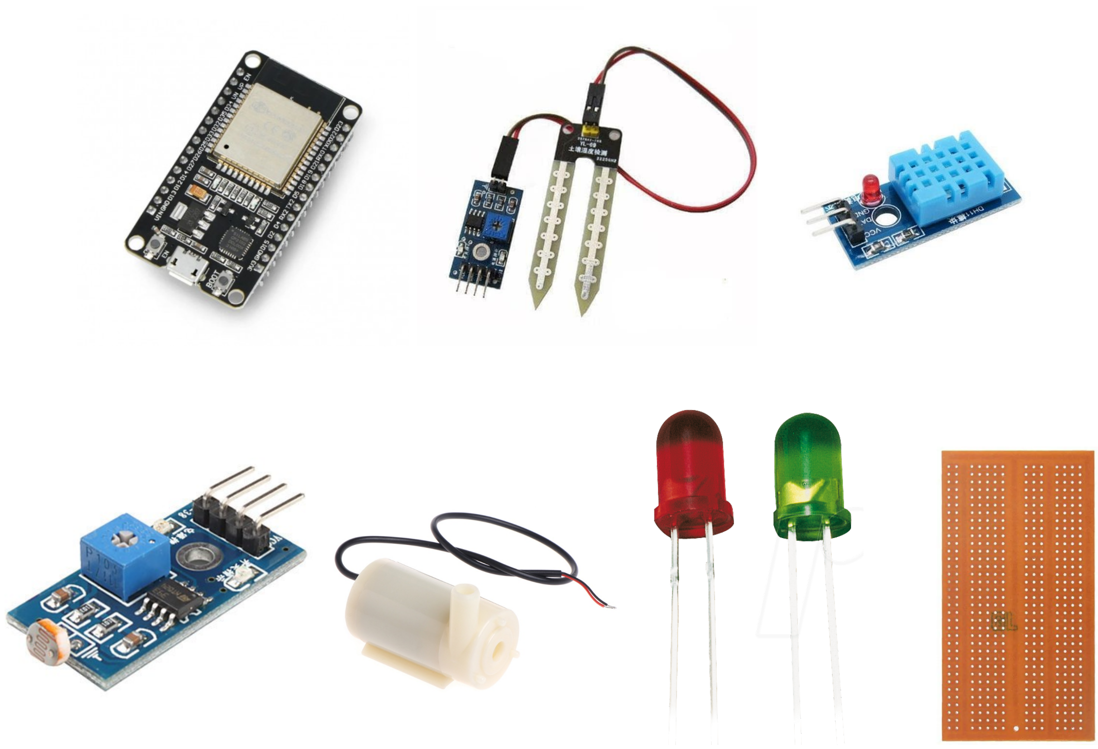


Fig 2.1. Components Used

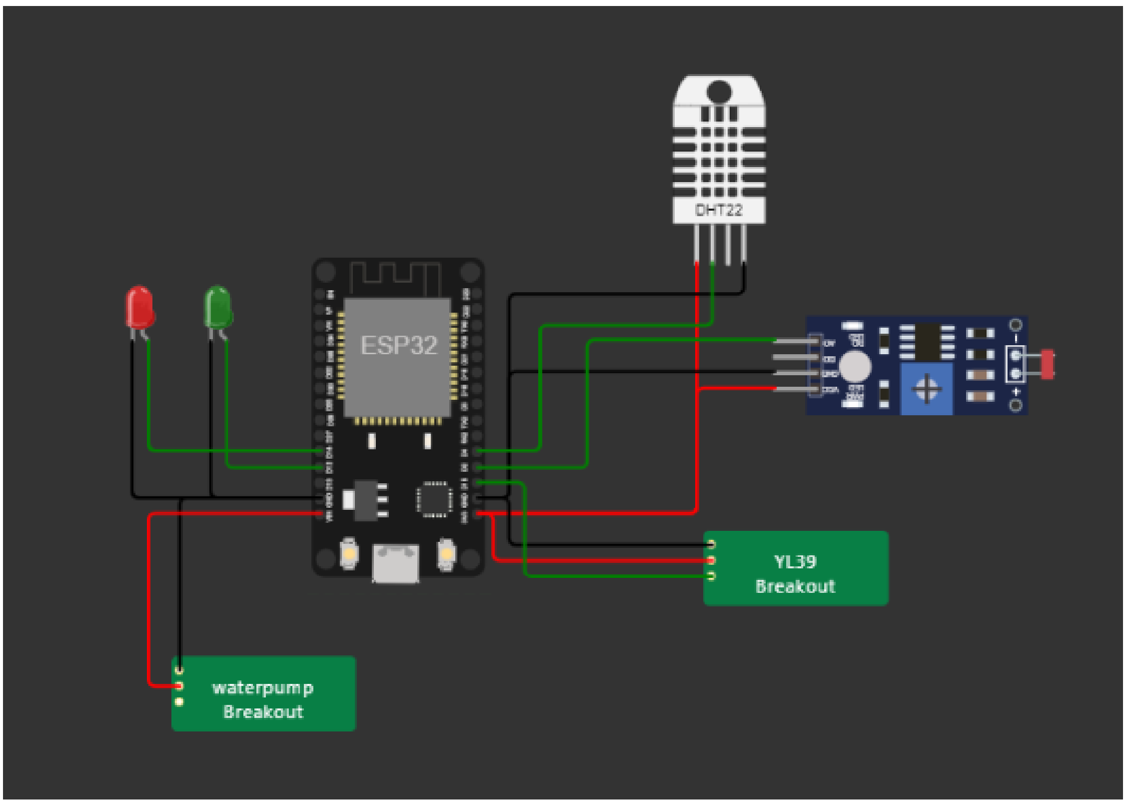
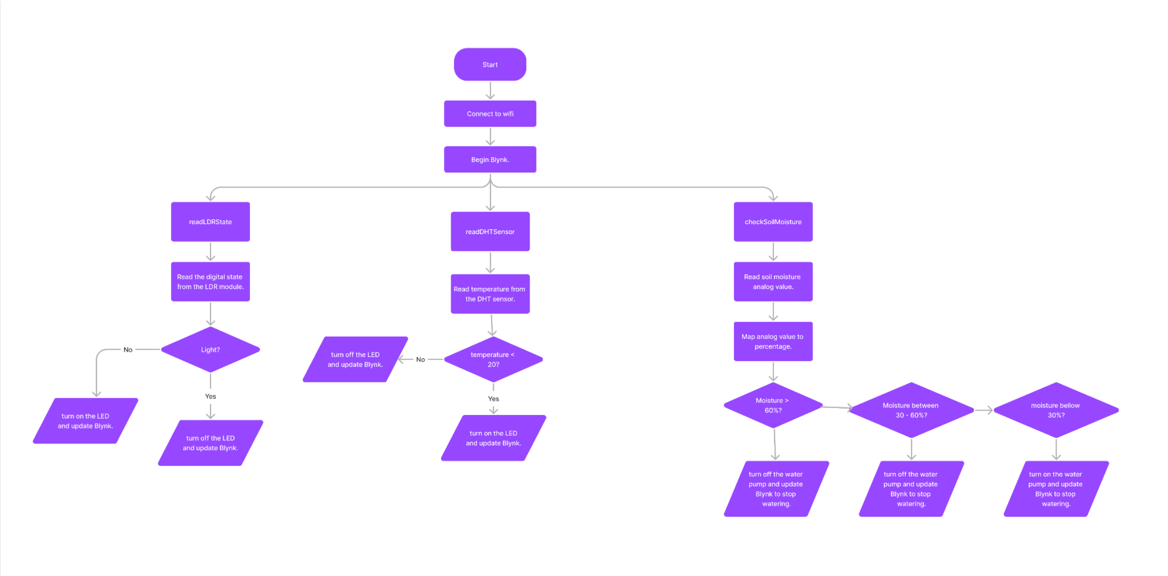


Fig 2.2. Hardware Schematic

2.2 SOFTWARE DEVELOPMENT

The presented code embodies the software development of a remote agriculture system, adept at monitoring and managing environmental conditions through the use of an ESP32 microcontroller. The integration with the Blynk platform is a pivotal aspect, enabling the creation of a remote user interface for real-time data visualization. The program establishes a Wi-Fi connection for seamless communication with the Blynk server. Employing a task-based architecture with FreeRTOS, the code efficiently manages concurrent operations. Three distinct tasks, namely readLDRState, readDHTSensor, and checkSoilMoisture, focus on monitoring light conditions, temperature, and soil moisture, respectively. Sensors such as the DHT22 for temperature and soil moisture sensors provide crucial environmental data. Actuators, in the form of LEDs, respond to environmental changes, with the LED_PIN adjusting based on ambient light, and the TEMP_LED_PIN responding to temperature thresholds. Additionally, two LEDs (SOIL_LED_PIN_1 and SOIL_LED_PIN_2) signify different soil moisture levels. Blynk.virtualWrite() statements facilitate the real-time update of virtual pins on the Blynk app, ensuring seamless data visualization. Intentional delays within the tasks control the frequency of sensor readings. Serial communication aids in debugging by providing insights into sensor readings and system states. The setup() function initializes crucial components, including serial communication, Wi-Fi connection, and Blynk integration, while GPIO pins are configured based on connected devices. The loop() function is deliberately kept minimal, as the tasks are designed to run concurrently, optimizing the system's responsiveness and overall performance. In essence, the software orchestrates a comprehensive ecosystem, integrating sensors, actuators, and remote visualization to facilitate effective remote agriculture management.



The flowchart illustrates the software execution of a remote agriculture system. The program initiates by connecting to a Wi-Fi network, establishing a crucial link for communication. Following successful connectivity, the software engages with the Blynk platform, facilitating remote monitoring and control. The parallel execution of tasks is a notable feature, where three tasks—readDHTSensor, readLDRState, and checkSoilMoisture—operate concurrently using FreeRTOS. The readDHTSensor task retrieves temperature data from the DHT22 sensor, sends it to the Blynk app, and controls a temperature LED based on thresholds. Simultaneously, the readLDRState task monitors ambient light conditions through a Light Dependent Resistor (LDR), adjusting an LED and reporting status to Blynk. Meanwhile, the checkSoilMoisture task reads data from a soil moisture sensor, adjusts two LEDs based on soil moisture levels, and communicates the status to Blynk. This parallelism enables real-time monitoring of temperature, light, and soil moisture, enhancing the efficiency of the remote agriculture system. The flowchart represents a continuous loop, ensuring ongoing monitoring and control in agricultural environments.

2.3 HARDWARE AND SOFTWARE INTEGRATION

In this project, the integration of hardware and software plays a crucial role in achieving a smart and responsive environmental monitoring system. On the hardware side, the project utilizes an ESP32 microcontroller as the central processing unit, capable of handling the diverse set of sensors and actuators. The hardware components include a

Light-Dependent Resistor (LDR) module for ambient light sensing, a YL69 humidity sensor for soil moisture measurement, and a DHT11 sensor for monitoring temperature. Additionally, two LEDs serve as indicators, providing visual feedback on the soil moisture conditions, and a water pump acts as an actuatable component for automated watering.

The software aspect of the project is implemented using the Arduino programming language, leveraging the capabilities of the ESP32 microcontroller. The code is structured into tasks that run in parallel, ensuring efficient and concurrent processing. The integration with the Blynk platform enhances the project's functionality by providing a user-friendly interface to remotely monitor the environmental parameters and control the system. Blynk allows real-time visualization of data through customizable widgets, offering a seamless bridge between the hardware and the user.

The interaction between hardware and software is orchestrated through tasks. For instance, the readLDRState function reads the state of the LDR module, adjusting the LED accordingly, and updating the Blynk interface. Similarly, readDHTSensor fetches temperature data, controls the temperature LED, and sends the information to Blynk. The checkSoilMoisture function processes the soil moisture data, dynamically managing the state of LEDs based on the moisture level and communicating the status to Blynk for user awareness.

This integration ensures that the hardware components respond intelligently to environmental changes and that users can conveniently monitor and control the system through the Blynk app. The ESP32 serves as the bridge between the physical world and the virtual interface, making the project a cohesive and effective hardware and software integration for environmental monitoring and automated plant care.

CHAPTER 3

TESTING AND EVALUATION

3.1 TESTING

The Remote Agriculture System project was developed using C++ programming language and the ESP32 microcontroller, utilizing Blynk for connectivity. The project aims to create an automated system for monitoring the health and environmental conditions of plants through sensors such as YL39 for soil moisture, DHT11 for temperature and humidity, and LDR for light intensity.

To ensure the reliability and functionality of the system, comprehensive testing was conducted at various stages of development. The following outlines the testing methodologies employed and the results obtained.

Unit Testing :

Unit testing was performed on individual components of the Remote Agriculture System to verify their proper functioning. Each module, including the moisture sensor (YL39), DHT sensor, and LDR sensor, was tested separately using appropriate test cases. This testing ensured that each component was calibrated correctly and generated accurate readings.

Integration Testing :

Integration testing aimed to assess the seamless integration of all individual components and modules of the Remote Agriculture System. The objective was to ensure that data flow and communication between various modules, including the DHT sensor, LDR sensor, and Blynk connectivity, were functioning as expected. Test cases were designed to simulate different scenarios, testing the system for stability, responsiveness, and data accuracy during these scenarios.

In this case, we tested the system by initially integrating only the DHT sensor and Blynk connectivity to confirm if these components could communicate and display the correct readings. Subsequently, we also tested the moisture sensor (YL39) and Blynk, along with just the LDR sensor and Blynk.

User Acceptance Testing :

User acceptance testing involves testing the device in an environment that could emulate real-world use cases, such as immersing the moisture sensor in water and simulating different temperature readings using the DHT sensor, as well as varying light levels with the LDR sensor. Feedback and observations were collected to assess the system's usability, intuitiveness, and overall experience. Any identified issues or suggestions for improvement were noted for further refinement of the Remote Agriculture System.

3.2 RESULT

All individual components pass the unit testing phase successfully, but at the beginning we encounter some trouble with the DHT11 sensor in which the sensor cannot pick up the temperature reading at all, and will only return 0 values. So we changed our DHT11 sensor to a new one and everything works well. YL39 moisture sensor is able to pick up accurate moisture level readings, the LDR can catch light reading pretty accurately. And each LED and water pump is working as expected as well.

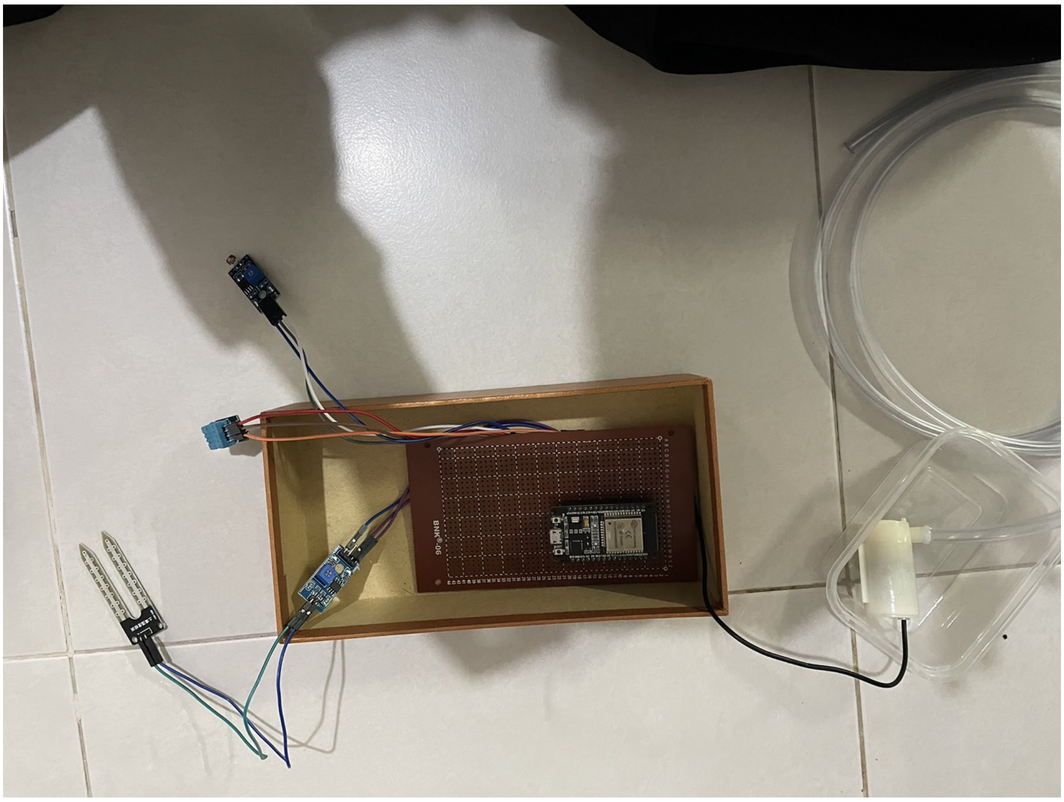


Fig 3.1 Unit Testing

The integration testing phase verified that the components of the Remote Agriculture System were effectively integrated. Data flow between modules was smooth, and the system provided accurate and real-time readings of environmental parameters for the plant such as its temperature, soil moisture, and the light reading.

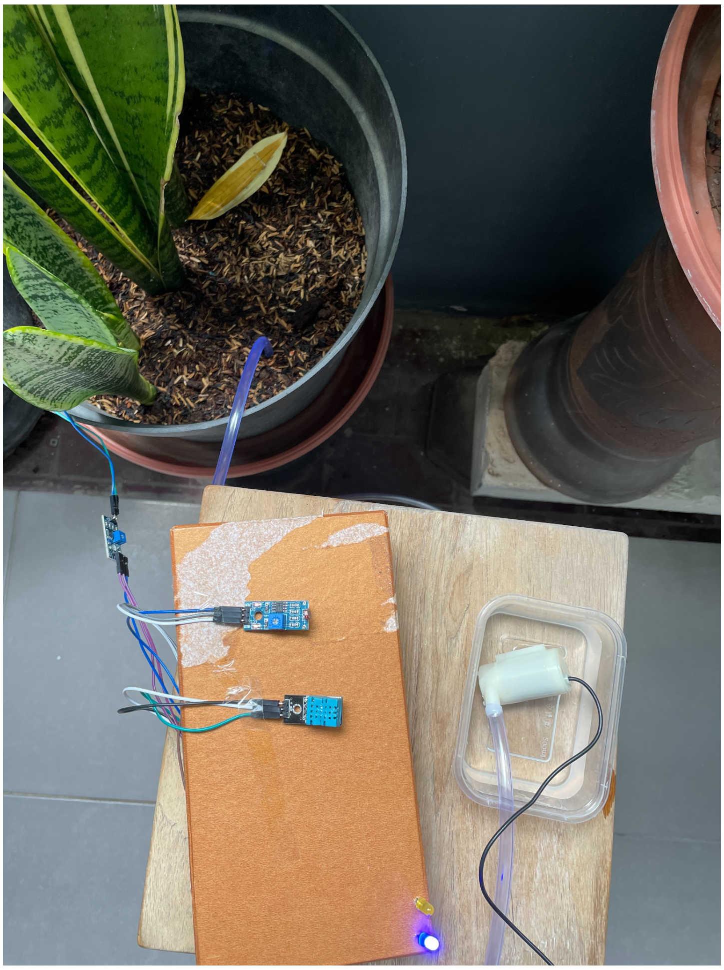


Fig 3.2 Integration Testing

During the user acceptance testing, the Remote Agriculture System exhibited successful performance when all the components were connected and operated together, simulating a real-world environment. The system effectively produced accurate readings, displaying the environmental temperature on Blynk. Additionally, the moisture level and light level were successfully presented on the display, as evidenced by the status indicators. This outcome affirms the system's capability to provide relevant and reliable data in diverse environmental conditions.

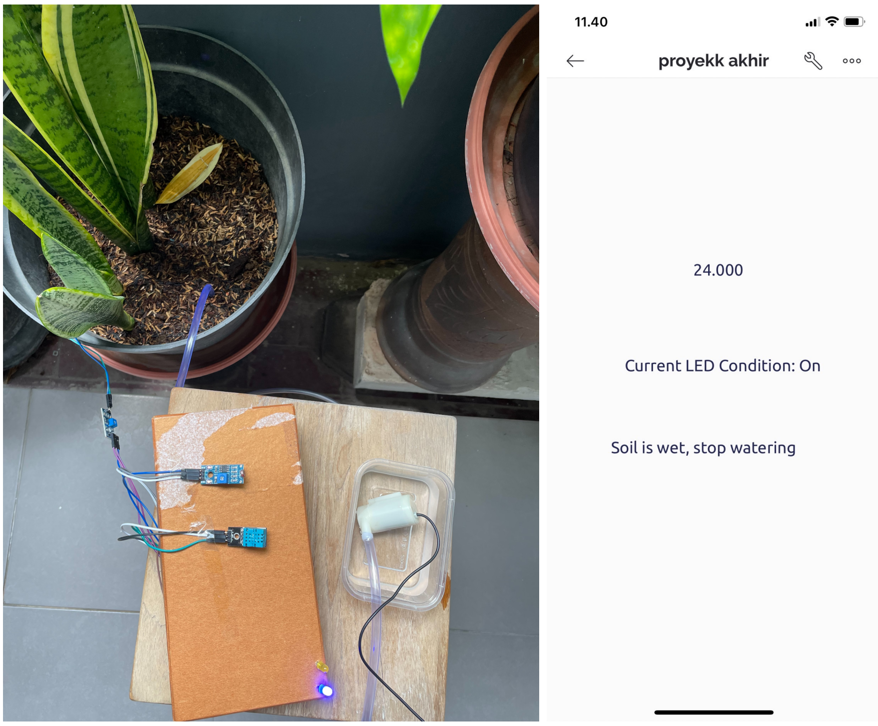


Fig 3.3 User Acceptance Testing

In summary, the testing process for the Remote Agriculture System included rigorous unit testing, integration testing, and user acceptance testing. The system demonstrated successful outcomes across all testing phases, showcasing precise readings, smooth integration of components, efficient performance, and favorable user feedback. These comprehensive tests validate the reliability and functionality of the Remote Agriculture System, establishing it as a valuable tool for both plant enthusiasts and professionals. The system's proven accuracy and seamless operation underscore its potential to contribute significantly to plant health monitoring and cultivation practices.

3.3 EVALUATION

The Remote Agriculture System is an IoT project that has demonstrated reliability by providing precise sensor readings, swift responses to environmental changes, and efficient energy usage. It effectively monitors essential factors like moisture levels, temperature, and light intensity, aligning well with its primary goal. User testing confirmed its accuracy and potential to improve farming practices by saving time and minimizing plant damage risks.

Despite its strengths, we identified areas for enhancement. A key improvement involves introducing manual controls through a dedicated app connected to Blynk. This would allow users to directly adjust watering schedules and control lighting, providing a more user-friendly experience.

In summary, the Remote Agriculture System is a commendable IoT project, showcasing strong performance and positive contributions to modernizing agriculture. The suggested improvements offer avenues for future development, ensuring the system continues to evolve to meet users' needs, offering greater control and flexibility in smart agriculture management.

CHAPTER 4

CONCLUSION

This IoT-based project represents a significant stride in the realm of smart agriculture, seamlessly integrating hardware components with sophisticated software development. By harnessing the power of the ESP32 microcontroller and an array of sensors, the system transforms traditional agriculture into a technologically advanced, remotely monitored endeavor. The utilization of Blynk as the IoT platform establishes a seamless connection, enabling users to access real-time data and control the system from any location with internet connectivity. The concurrent execution of tasks, made possible by FreeRTOS, underscores the project's commitment to efficiency and responsiveness. With a focus on optimizing resource usage, the system continuously monitors temperature, light conditions, and soil moisture, making informed decisions to enhance crop growth. This project serves as a testament to the transformative impact of IoT in agriculture, paving the way for sustainable, data-driven farming practices that can be remotely managed for increased productivity and resource efficiency.

REFERENCES

[1] A. Choudhary. " IoT based Smart Agriculture Monitoring System," circuitdigest.com, [Online]. Available : https://circuitdigest.com/microcontroller-projects/iot-based-smart-agriculture-moniotring-system Accessed : Nov 21, 2023

[2] Circuit Digest, "IoT based DIY Smart Agriculture Monitoring System (2022)," YouTube. Aug 25, 2021. Accessed : Nov 22, 2023. [YouTube Video]. Available : https://www.youtube.com/watch?v=gz3hDL2tgng&ab_channel=CircuitDigest

[3] "ESP32 Pinout Reference: Which GPIO pins should you use?," randomnerdtutorials.com, [Online]. Available : https://randomnerdtutorials.com/esp32-pinout-reference-gpios/ Accessed : Nov 25, 2023

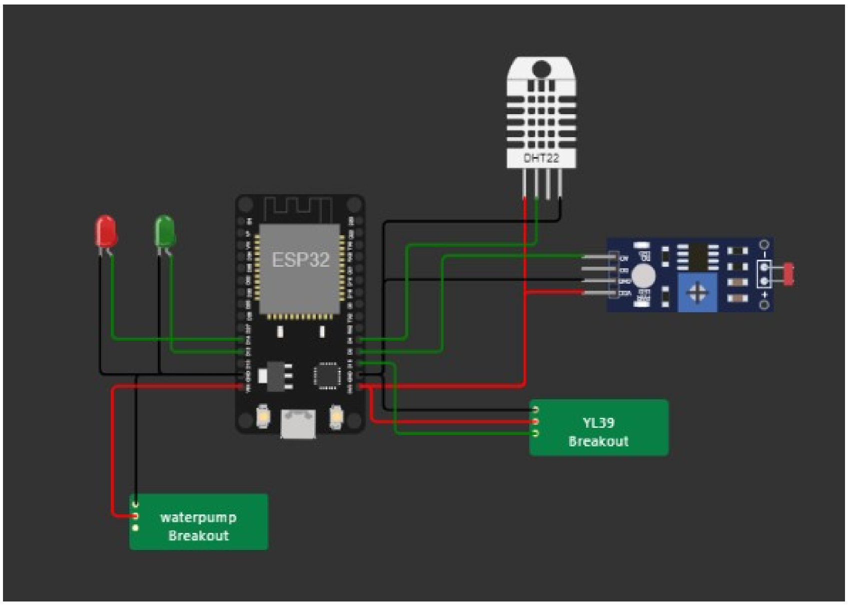
[4] "ESP32 - Soil Moisture Sensor," esp32io.com, [Online]. Available : https://esp32io.com/tutorials/esp32-soil-moisture-sensor Accessed : Nov 26, 2023

[5] "ESP32 with DHT11/DHT22 Temperature and Humidity Sensor using Arduino IDE" randomnerdtutorials.com, [Online]. Available : https://randomnerdtutorials.com/esp32-dht11-dht22-temperature-humidity-sensor-arduino-ide/ Accessed : Nov 26, 2023

[6] INOVATRIX, "How to Use ESP32 with LDR sensor and Arduino IDE," YouTube. Apr 21, 2023. Accessed : Nov 26, 2023. [YouTube Video]. Available : https://www.youtube.com/watch?v=Va-psZoTBkY&ab_channel=INOVATRIX

APPENDICES

Appendix A: Project Schematic



Appendix B: Documentation