Faculty of Computer Science and Information Technology

SMART AGRICULTURE WEATHER STATION POWERED BY IOT

NUR ADLIN DINA BINTI ROHAIZAT

BACHELOR OF SOFTWARE ENGINEERING WITH HONORS

2025

SMART AGRICULTURE WEATHER STATION POWERED BY IOT

NUR ADLIN DINA BINTI ROHAIZAT

This project is submitted in partial fulfilment of the requirements for the degree of

Bachelor of Software Engineering with Honours

Faculty of Computer Science and Information Technology

UNIVERSITI MALAYSIA SARAWAK

2025

SMART AGRICULTURE WEATHER STATION POWERED BY IOT

NUR ADLIN DINA BINTI ROHAIZAT

Projek ini merupakan salah satu keperluan untuk

Ijazah Sarjana Muda Kejuruteraan Perisian dengan Kepujian

Fakulti Sains Komputer dan Teknologi Maklumat

UNIVERSITI MALAYSIA SARAWAK

2025

Acknowledgements

Firstly, I would like to thank my supervisor Assoc. Prof. Dr. Noor Alamshah Bin Bolhassan for their invaluable guidance, support, and encouragement throughout the development of my Final Year Project. Their expertise and insights have been instrumental in the successful completion of this project. I am also thankful to University Malaysia Sarawak for providing the resources and facilities needed to conduct this project. The learning opportunities and support offered during this period have greatly enhanced my understanding and skills.

I would like to extend my heartfelt appreciation to my family, especially my strict mom but more lovely and soft person that I've ever known. She has been a true source of inspiration that encouraging me to complete my studies even when I didn't believe I could earn my degree at the age of 24. My mother has also supported me financially and in many other ways. She always prayed for me, listened to my worries and witnessed everything I went through throughout this degree journey. I just want to say that even though my results may not be excellent, I promise that one day I will become a child she can truly be proud of. If not now maybe one day.

Moreover, I would like to express my sincere gratitude to all my friends who have supported me since the first year. Even though we were all new at the time, they never hesitated to help and guide me, especially in completing assignments and navigating our studies together. Thank you for always being supportive, kind, and genuine throughout this journey. I truly hope that we will all graduate together and shine brightly in our own paths like stars in our degree journey. Good luck to all of you as you move into the next phase of your lives and careers and I believe you'll achieve great things and know that each of you will always be in my prayers.

Lastly, I want to sincerely thank myself for making it through this long and challenging journey. There were times when I felt like giving up, but I kept pushing forward even when the path was uncertain and the days felt heavy. I've learned to be patient with myself, to trust my growth and to keep showing up no matter how tough it got. This achievement is not just about earning a degree, but about proving to myself that I am capable, strong and worthy of success. As the Malay proverb goes, *"Kalau tidak dipecahkan ruyung, manakan dapat sagunya,"* meaning that without effort, nothing can be achieved. Today, I stand proud, knowing that every struggle was worth it and this is only the beginning.

Table of Content

List of Figures 9

List of Tables 11

Abstract 12

Abstrak 13

Chapter 1: Introduction 14

1.1 Introduction 14

1.2 Problem Statement 14

1.3 Scope 15

1.4 Aims and Objectives 16

1.5 Factors/Parameters for Weather Station 17

1.6 Brief Methodology 18

1.7 Significance of Project 19

1.8 Project Schedule 20

1.9 Expected Outcome 21

1.10 Summary 22

Chapter 2: Background Study 23

2.1 Introduction 23

2.2 Review on Existing Systems 23

2.3 Comparison between Existing Systems 38

2.4 Comparison between Proposed System and Existing Systems 39

2.5 Tools and Technology used in Development of the Proposed System 40

2.6 Summary 40

Chapter 3: Requirement Analysis and Design 41

3.1 Introduction 41

3.2 Requirement Analysis 42
3.2.1 Software Requirements 42
3.2.2 Hardware Requirements 43
3.3 System Design 44
3.3.1 Use Case Diagram 44
3.3.2 System Architecture 46
3.3.3 Data Flow Diagram 47
3.3.4 Entity-Relationship Diagram 48
3.3.5 Prototype Design 49
3.4 Summary 50
Chapter 4: Development 51
4.1 Introduction 51
4.2 Tools Deployment 51
4.3 Hardware Deployment 52
4.3.1 Hardware Setup Phase 52
4.4 Software Deployment 53
4.4.1 Arduino IDE 53
4.4.2 Arduino IoT Cloud 55
4.5 Summary 56
Chapter 5: Testing 57
5.1 Introduction 57
5.2 Functional Testing 57
5.2.1 Component Testing 57
5.2.2 System Testing 58
5.3 Non-Functional Testing 61

5.3.1 Usability Testing 62

5.4 Summary 62

Chapter 6: Conclusion and Future Works 63

6.1 Introduction 63

6.2 Objective Achievement 63

6.3 Project Limitations 63

6.4 Future Works 64

6.5 Conclusion 64

References 65

List of Figures

Figure 1.1: Shows the Waterfall Model 18
Figure 1.2: Shows the Project Schedule FYP 1 of submission Chapter 1, 2 and 3 20
Figure 1.3: Shows the Project Schedule FYP 2 of submission Chapter 4, 5, 6 and Final Report 20
Figure 2.1: Shows the Basic Components of IoT based Weather Station 24
Figure 2.2: Shows the Proposed Development Life Cycle of IoT based Weather Station 26
Figure 2.3: Shows the Block Diagram of Control System for Development of a Weather Station Using IoT 29
Figure 2.4: Shows the 3D Design of Weather Station along with actual on-site Installation of Structure 31
Figure 2.5: Shows the Meteorological Data a and b 32
Figure 2.6: Shows the Meteorological Data c and d 33
Figure 2.7: Shows the Meteorological Data e 33
Figure 2.8: Shows the Block Diagram of the Smart Farming System 36
Figure 2.9: Shows the Output Unit of the Smart Farming System 37
Figure 2.10: Shows the Smart Weather Monitoring System 37
Figure 2.11: Shows the Low-cost Wind Vane of the Smart Farming System 37
Figure 3.1: Shows the Use Case Diagram of Proposed System 44
Figure 3.2: Shows the Block Diagram of Proposed System 46
Figure 3.3: Shows the Data Flow Diagram of Proposed System 47
Figure 3.4: Shows the Entity-Relationship Diagram of Proposed System 48
Figure 3.5: Shows the Design Prototype of the Proposed System 49
Figure 4.1: Shows the Component of the Proposed System 52
Figure 4.2: Shows the Hardware Final Setup Proposed System 52
Figure 5.1: Shows the on Top of the Box that Complete Design of the Proposed System 57
Figure 5.2: Shows the Inside of the Box that Complete Design of the Proposed System 58
Figure 5.3: Arduino IDE page 59
Figure 5.4: Shows the Serial Monitor Sensor Reading 1 in Arduino IDE 59
Figure 5.5: Shows the Serial Monitor Sensor Reading 2 in Arduino IDE 59
Figure 5.6: Shows the Web Dashboard in Arduino IoT Cloud of Proposed System (Adlinweather) 60

Figure 5.7: Shows the Mobile Dashboard in Arduino IoT Cloud of Proposed System (Adlinweather) 60
Figure 5.8: Shows the Serial Monitor in Arduino IoT Cloud of the Proposed System (Adlinweather) 61

List of Tables

Table 1.1: Lists of Factors/Parameters for Weather Station 17
Table 2.1: Shows the Comparison between Existing Systems 38
Table 2.2: Shows the Comparison between Proposed System and Existing Systems..... 39
Table 3.1: Lists of the Software Requirements with Description 42
Table 3.2: Lists of the Hardware Requirements with Description..... 43
Table 3.3: Shows the Use Case Description 45

Abstract

The Smart Agriculture Weather Station Powered by IoT is an advanced technology project designed to support small-scale organic farmers by providing real-time monitoring of critical environmental parameters essential for crop growth and sustainability. These farmers often face challenges such as unpredictable weather conditions, lack of access to accurate data, and the prohibitive costs of advanced farming technologies. This project leverages the ESP8266 microcontroller, IoT sensors and the Arduino IoT Cloud to create an affordable and effective solution for monitoring parameters such as temperature, humidity, light intensity, and rainfall. The system collects and processes data that can be accessed remotely via a user-friendly web and mobile dashboards. By implementing a system that captures real-time data on key environmental factors can empower farmers to make more informed decisions that will lead to healthier crops, more efficient use of resources, and better productivity. The integration of IoT technology not only enhances the efficiency of resource usage, such as water and fertilizers, but also promotes sustainable farming practices by reducing resource waste, and improved crop health. This project prioritizes simplicity and affordability to ensure it is accessible to farmers with limited budgets and technical expertise. By bridging the gap between traditional farming methods and modern technological advancements, the system addresses the pressing need for accessible agricultural tools. Lastly, this project contributes to the productivity, sustainability, and adaptability of organic farming while promoting a more sustainable and self-reliant agricultural sector.

Abstrak

Projek Stesen Cuaca Pertanian Pintar Berasaskan IoT adalah satu teknologi canggih yang bertujuan untuk menyokong petani organik berskala kecil dengan menyediakan pemantauan masa nyata terhadap parameter persekitaran yang kritikal bagi pertumbuhan tanaman dan kemampanan. Petani-petani ini sering menghadapi cabaran seperti keadaan cuaca yang tidak menentu, kekurangan akses kepada data yang tepat, dan kos tinggi teknologi pertanian maju. Projek ini menggunakan mikropengawal ESP8266, sensor IoT dan Arduino IoT Cloud untuk mencipta penyelesaian yang mampu milik dan berkesan untuk memantau parameter seperti suhu, kelembapan, keamatan cahaya, dan hujan. Sistem ini mengumpul dan memproses data yang boleh diakses secara jarak jauh melalui papan pemuka web yang mesra pengguna. Dengan melaksanakan sistem yang merekod data masa nyata mengenai faktor persekitaran utama, petani dapat membuat keputusan yang lebih tepat, yang akan membawa kepada tanaman yang lebih sihat, dengan penggunaan sumber yang lebih cekap, dan peningkatan produktiviti. Integrasi teknologi IoT bukan sahaja meningkatkan kecekapan penggunaan sumber seperti air dan baja, tetapi juga mempromosikan amalan pertanian mampan dengan mengurangkan pembaziran sumber dan meningkatkan kesihatan tanaman. Projek ini mengutamakan aspek kesederhanaan dan mampu milik untuk memastikan ia dapat diakses oleh petani yang mempunyai bajet terhad dan pengetahuan teknikal yang minimum. Dengan merapatkan jurang antara kaedah pertanian tradisional dan kemajuan teknologi moden, sistem ini memenuhi keperluan mendesak alat pertanian yang mudah diakses. Akhir sekali, projek ini menyumbang kepada produktiviti, kemampanan, dan keupayaan adaptasi pertanian organik sambil mempromosikan sektor pertanian yang lebih mampan dan berdikari.

Chapter 1: Introduction

1.1 Introduction

Weather is related to the conditions of temperature, humidity and wind in a location over a specific time. In general, the weather is always changing. Sometimes there is a dry season, rain, until snowfalls. The weather is generally influenced by three elements, namely the sun, water, and wind (Eko Murdyantoro, 2019). Smart agriculture is usually used to denote the application of IoT solutions in agriculture. Farmers can improve almost every aspect of their work, from improving livestock to growing crops, by employing IoT sensors to collect environmental and machine metrics (Vikaspdia, n.d.). Small-scale and organic farmers often face challenges due to unpredictable weather that negatively impact crop yield. These farmers face difficulties in tracking important factors such as temperature, humidity, light intensity and rainfall which can lead to water wastage, nutrient depletion, and reduced crop quality. Small-scale organic farmers cannot afford modern to use smart farming solutions due to their high cost and technology. Maintaining ideal growing conditions is challenging due to the lack of simple technology which in turn affects crop yield and the sustainability of farming methods. Due to this problem, the Smart Agriculture Weather Station system is designed to support small-scale organic farmers by providing real-time environmental monitoring through a powered IoT weather station that uses the ESP8266 microcontroller. The system continuously collects data important to maintain the optimal growing environment through sensors that measure temperature, humidity, light intensity, and rainfall. The ESP8266 processes and sends this data with a Wi-Fi or cellular networks to the Arduino IoT Cloud, where it can be accessed remotely through a user-friendly web and mobile dashboards. This allows farmers to monitor conditions and adjustments from anywhere.

1.2 Problem Statement

Uncertain weather monitoring for small-scale organic farming is one of the problem statements in this project. Controlling plants can be extremely difficult for small-scale organic farmers because of weather conditions are unpredictable, especially considering climate change and region weather changes. Small farms usually rely on general weather reports that might not accurately represent local conditions in comparison to large farms that frequently have advanced weather data and determining tools. Organic farmers must manage their environment carefully because they avoid chemicals like fertilizers and pesticides that might effects of bad weather. For example, a

sudden drop in temperature or unexpected rain can damage organic produce and cause damage to sensitive crops, disturb the soil balance, or increase the growth of fungi. Small-scale farmers are often unprepared for these changes in due to the lack of a reliable local weather tracking systems and lead to reduced crop yields and increased costs to repair damage.

Other than that, because of the lack of real-time data for optimizing plant growth that manage environmental factors like temperature and humidity have importance for effective plant growth. Many organic farming methods place an intense focus on carefully managing these factors to maintain the natural health of the soil and plants. In the other hand, sudden changes in humidity of the environment may make plants more exposed to plant health. So, this should be discussed right away to prevent it. By implementing a system that captures real-time data on key environmental factors can empower farmers to make more informed decisions that will lead to healthier crops, more efficient use of resources, and better productivity.

Due to limited accessibility to smart farming solutions, many advanced agricultural solutions that include sensors and automatic irrigation systems that are designed for large-scale commercial farms and require high installation and maintenance costs that make them difficult to access for small-scale or private organic farmers. By providing small-scale farmers the resources and information they need to improve their crop management and make better decisions without requiring difficult technical training or larger costs, the development of an affordable and easy to use weather station system can close this gap. This accessibility will contribute to more stable productive organic farms, benefiting farmers and communities that depend on sustainable agricultural produce.

1.3 Scope

Even though this system later cannot manage the entire field management, this weather station will provide important information that farmers need to optimize crop conditions. For example, although this system can provide early warnings to farmers about changes in environmental factors such as temperature, humidity, light intensity, and rainfall, it is not equipped with additional functions such as automatic controlling bugs, ongoing fertilization, or a thorough analysis of soil nutrients. Due to the higher costs and technical requirements, this capability is outside the scope of this project, but it is also important in organic farming to make sure the optimal crop health and quality. This system therefore functions as an important tool for environmental monitoring rather than a complete answer to every aspect of handling an organic farm.

This project highlights an important priority on simplicity and affordability, that aim to make the solution easy for small-scale organic farmer without requiring advanced technical expertise. This simplicity means that some advanced data analytics and machine learning features, which could improve understanding trends and estimations are not included. The Arduino IoT Cloud's remote access feature has limitations to collecting data in real time and giving out simple alerts. Therefore, while the system will offer critical, real-time environmental insights to farmers and it remains a foundational tool rather than a comprehensive smart farming solution, addressing immediate needs for real-time monitoring and accessibility but not yet achieving full farm automation.

1.4 Aims and Objectives

1. To design and implement a weather station that continuously monitors key environmental factors such as temperature, humidity, light intensity and rainfall.
   - The main objective of the project is to develop a fully functional weather station that can monitor temperature, humidity, light intensity, and rainfall in real-time. By providing continuous monitoring, the system will help farmers ensure that their crops are growing under optimal conditions. This system will be using the ESP8266 microcontroller that will process and send the data with Wi-Fi or cellular networks to the Arduino IoT Cloud where it can be accessed remotely through a user-friendly web and mobile dashboards. This allows farmers to monitor weather conditions from anywhere.
2. To enable remote access to real-time environmental data using the Arduino IoT Cloud.
   - This remote access enabling farmers to monitor their crops by know about the weather conditions from any location via a web and mobile dashboards. This will allow for timely decision-making and reduce the farmer being there in the field, improving both convenience and efficiency.
3. To promote sustainable farming practices.
   - The project aims to promote more sustainable farming practices. This includes better reduced resource waste. Resource optimization will be measured by analyzing the reduction in unnecessary fertilizer and pesticide use based on environmental data collected from sensors. The project's impact on long-term sustainability can be

quantified and validated by comparing these metrics before and after system deployment.

1.5 Factors/Parameters for Weather Station

Temperature	Tracking temperature is important for the weather station because solar radiation affects it through Earth's surface warming. The Earth's surface absorbs solar energy, which is turned into thermal energy and delivered to the atmosphere by condensation and conduction. It helps farmers to optimize crop health (Lima, n.d.).
Humidity	Capacitive or resistive sensors are used in modern weather stations to measure humidity. Resistors measure changes in resistance based on changes in conductivity brought on by humidity, whereas capacitive sensors detect changes in capacitance brought on by water vapor. Weather stations must follow demanding calibration procedures to ensure the accuracy of temperature and humidity measurements (Sensors, 2023).
Light Intensity	Light intensity is the measure of the brightness or strength of light over a surface, which varies based on distance and light source power. Light intensity directly influences the activity of photosynthetic process, which affecting how much energy plants can store as carbohydrates (Light Intensity – Intro to Botany Review, 2024).
Rainfall	This component includes a sensing pad with a multiple of open cooper lines. In dry weather, the traces are not connected to each other, but rainfall forms a temporary connection through the water droplets. This wet surface causes a change in resistance, functioning like a variable resistor that responds to water levels (How Rain Sensor Works and Interface it with Arduino, 2025).

Table 1.1: Lists of Factors/Parameters for Weather Station

1.6 Brief Methodology

Waterfall methodology is a commonly used project management method with a standard approach. Every step of the Waterfall workflow must be finished before it can proceed to the next. While there are different types of project management methodologies, the Waterfall Model is well suited for projects where the objectives are clearly defined from the beginning (Leeron Hoory, 2024), which is a good fit for my Weather Station System Project. So, Waterfall methodology was chosen for this project "Smart Agriculture Weather Station Powered by IoT".

Figure 1.1: Shows the Waterfall Model

(Retrieves from Ajackus)

Figure 1.1 shows Waterfall process. Normally, one phase does not start until the one before it is finished. The following are usual stages of the Waterfall process:

1. Requirements
   - Written requirements, usually covered in a single document, are used to describe each stage of the project, including the costs, assumptions, risks, dependencies, success metrics, and timelines for completion. It also needs to identify the problem statement, aims and objective to achieve.
2. Design
   - Designing a technical solution to the problems set out by the product requirements, including scenarios, layouts, and data models is needed. The aim and scope of the project, the general traffic flow of each component, and the process of integration points are all described in a higher-level or logical design.

3. Implementation
   - In this stage, after the design is done, technical implementation starts. The projects are ready to be built, and code based on project requirements and specifications. If there are significant changes, this might go back to the design stage.
4. Testing
   - Testing needs to be done to make sure the system has no errors, and all the requirements have been completed.
5. Maintenance
   - After all the stage is done, the system still needs to be maintained for updating and upgrading, implementing a software upgrade or testing for errors and fixing them if occurring.

This Waterfall Model is easy to understand and easily managed. It focuses on one step at a time, and it works well for small projects with clearly understood requirements to easily organize tasks (Patankar, 2021).

1.7 Significance of Project

This Smart Agriculture Weather Station project is significant because it directly addresses the challenges faced by small-scale organic farmers in managing environmental conditions crucial for crop growth. The system gives farmers the ability to make timely, well-informed decisions that improve crop health and yield by giving them the ability to access real-time data on factors such as temperature, humidity, light intensity, and rainfall. This project supports sustainable farming practices by enabling more efficient water use and reducing the need for manual labor, which is especially beneficial for organic farming, where synthetic chemicals are not used.

This project highlights an important focus on smart farming technology's accessibility. Anyway, in the market today, many accurate agriculture systems are expensive and complicated, which makes them too costly just for small farms with a limited budget. This weather station has been designed specifically to be affordable and easy to use for small and limited farms. The system offers remote access through the Arduino IoT Cloud that allows farmers to keep an eye on essential environmental data from any location without having to be there 24/7. Farmers can have flexibility time for making decisions that also can reduce labor costs and provide real-time information. From there, farmers no need to worry about the condition of the plants all the time.

This project encourages the global trend toward more resilient and sustainable farming methods while also promoting the use of IoT in agriculture. This project is important because agriculture is getting influenced by climate variability more and more. Other than that, this project not only makes technology-driven agriculture more accessible but also increases small-scale farmers productivity and environmental care. It offers these farmers a practical solution to improve their methods in an environmentally friendly way that brings benefits to both their operation (businesses) and their communities.

1.8 Project Schedule

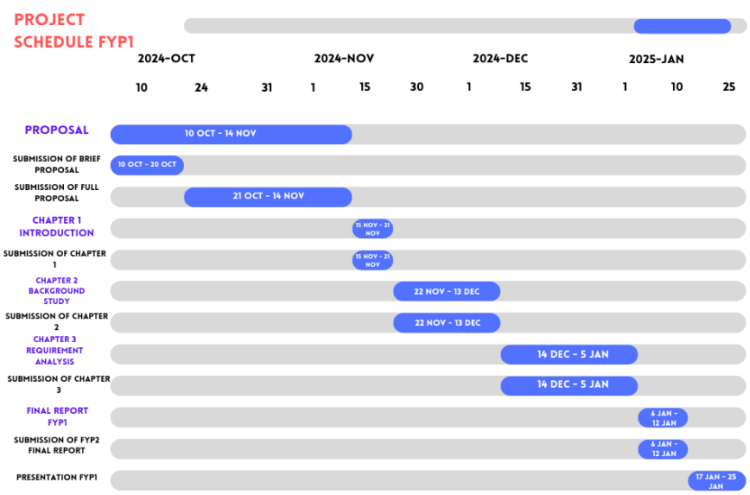

Figure 1.2: Shows the Project Schedule FYP 1 of submission Chapter 1, 2 and 3

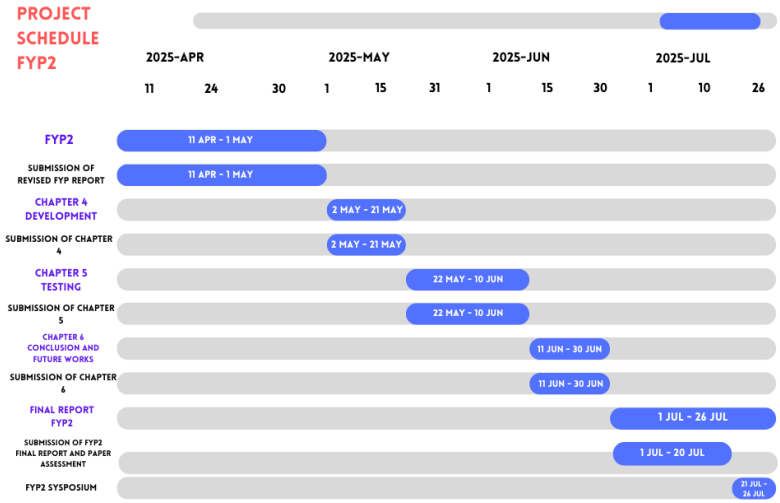

Figure 1.3: Shows the Project Schedule FYP 2 of submission Chapter 4, 5, 6 and Final Report

1.9 Expected Outcome

It has been projected that the "Smart Agriculture Weather Station Powered by IoT" project will improve small-scale organic farming in several of the important methods. First, by providing ongoing current data on important environmental parameters such as temperature, humidity, light intensity, and rainfall, the system will be used for improved real-time monitoring. Farmers will be able to improve farm productivity and reduce crop losses from sudden weather changes by using this real-time data to make timely adjustments to their crop environments.

Next, farm management will be more effective and convenient with remote access to environmental data through the Arduino IoT Cloud. Farmers will be able to improve their management processes and reduce the need for constant field access by using a web and mobile dashboards to monitor field conditions from any location. This feature is particularly beneficial for small-scale farmers, as it allows them to focus on other important tasks while still being informed about the conditions that affect their crops.

Furthermore, the project aims to support sustainable farming methods by giving farmers current information so they can make better choices. Sustainable agriculture depends on farmers being able to optimize water use, improve crop health, and reduce resource waste with right away access to environmental data. This will help them achieve good crops and higher yields without relying on manual labor or chemical input, benefiting both farm sustainability and produce quality.

Other than that, this weather station is made to be an affordable, easy-to-use solution that allows small-scale farmers to access smart farming technology. The project presents an affordable option without losing functionality, which addresses the accessibility factor that has historically prevented small-scale farmers from implementing smart farming methods.

This will enable them to use technology to improve produce and more effectively compete in the agricultural sector. This system also allows them to use technology to improve productivity and compete more successfully in the agricultural sector.

Lastly, while the system is not fully automated, it provides a foundation for integrating additional IoT capabilities in the future. This project could inspire future expansions, such as automated irrigation or pest control systems by laying the groundwork for basic environmental monitoring, to support comprehensive smart farming. Using impactful and easily accessible technology, these results aim to develop a more effective, productive, and sustainable approach to small-scale organic farming.

1.10 Summary

This proposed system aims to address the challenge faced by small-scale organic farmers due to uncertain weather, lack of real-time data for optimizing plant growth and limited access to smart farming solutions. The system powered uses IoT technology by the ESP8266 microcontroller to monitor key environmental factors such as temperature, humidity, light intensity, and rainfall in real-time and the data will send to the Arduino IoT Cloud that allow farmers to access remotely from web and mobile dashboards. The project is designed to be affordable, user-friendly, and sustainable, promoting better crop management and resource efficiency while supporting organic farming practices.

Chapter 2: Background Study

2.1 Introduction

This chapter provides an in-depth review and comparison of IoT-based weather station projects, focusing on their applications in smart agriculture. As smart farming gains global attention due to its potential to optimize resource usage and improve crop yields, IoT weather stations have emerged as essential tools for monitoring environmental factors such as temperature, humidity, light intensity, and rainfall. Through this comparison, the chapter will identify the strengths and limitations of current systems, offering insights into areas for improvement or innovation in the proposed design. Additionally, the review will explore the integration of advanced features such as predictive analytics, solar-powered setups, and mobile app compatibility, ensuring the new system is efficient and user-centric. By synthesizing these insights, this chapter aims to position the proposed system within the broader context of IoT-based agricultural technologies, demonstrating its relevance and potential impact.

2.2 Review on Existing Systems

Energy Neutral Weather Station for Precision Agriculture

This system is a sustainable and autonomous system designed for precision agriculture, capable of monitoring important environmental parameters such as temperature, humidity, wind speed, and rainfall using IoT-based sensors. Powered entirely by energy harvesting such as solar and wind, it reduces operational costs and maintenance requirements for farmers in remote locations. Its energy-neutral design includes advanced generated power and management technologies, such as Maximum Power Point Tracking (MPPT) algorithms, modular storage solutions like lithium-ion batteries or supercapacitors, and dynamic energy optimization strategies. This system is highly modular, allowing customization of sensors and energy solutions based on their specific agricultural needs such as focusing on rainfall during the monsoon or temperature monitoring during a heatwave. With advanced energy management strategies and data transmission optimization, it enables local weather data collection and forecasting to support informed decision-making, ultimately improving crop yields and resource efficiency.

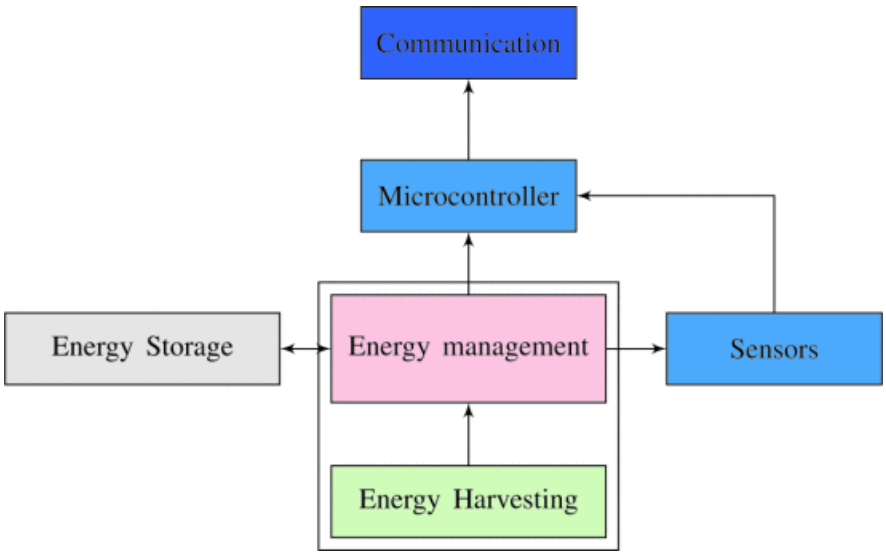

Figure 2.1: Shows the Basic Components of IoT based Weather Station

(Retrieves from Energy-neutral weather stations for precision agriculture: challenges and approaches)

In figure 1, the components of an IoT based weather station are shown, highlighting their functionality and interconnectivity. These basic components of IoT based weather stations include sensors, energy management, energy harvesting, energy storage, microcontroller and communication. Each component plays a vital role in making sure the system operates efficiently and autonomously.

A. Component Overview

The energy management manages the power supply and consumption of the weather station. It integrates seamlessly with the energy harvesting subsystem, which captures ambient energy sources like solar, wind energy or vibration to the power station. Solar panels and wind turbines are the main means of energy generation, and their output is advanced generated power and management technologies such as Maximum Power Point Tracking (MPPT) algorithms. This harvested energy is important for sustaining the station in remote or off-grid locations.

During period of low solar or wind activity, the harvested energy is stored in suitable storage solutions like lithium-ion batteries, lead-acid batteries and supercapacitors to make sure operation can maintain the consistency. Each type of storage has different advantages, which is lithium-ion

batteries are lightweight with high energy density that can storing more energy in less space but are sensitive to low temperatures, lead-acid batteries have lower upfront costs but shorter lifespans, making lithium-ion more cost-effective in the long term, (Marsh, 2023) and supercapacitors are durable for rapid charge or discharge cycles but not suitable for long-term energy storage. Complementing this is the energy storage unit, which is responsible for storing energy collected from energy harvesting for use when needed.

Next, the sensors collect important environmental parameters, such as temperature, humidity, wind speed and atmospheric pressure. Low-power sensors are prioritized to reduce the usage of energy on the system and their configuration is modified to meet the specific requirements of the agricultural environment. An example, a rain sensor can function periodically based on weather forecasts to save energy or temperature sensors can modify their sampling rates according to crop life cycle stages. This optimization makes sure accurate data collection without unnecessary energy use.

Microcontroller acts as the station's brain that processes sensor data and coordinating system activities. It also serves as the processing hub of the station. To save energy, low-power microcontrollers with features like deep sleep cycles and interrupt-based wakeups are used. These controllers remain in a low-power state during inactivity and are only activated by external signals, such as data availability from sensors or scheduled tasks. This effective energy management improved the operational life of the system and makes sure continuous functionality in remotes locations.

Last but not least, the communication block is responsible for transmitting the collected data to a remote location for analysis and visualization. Many of the station power usage is due to this energy-intensive block. To avoid this, the system employs energy-efficient communication protocols such as MQTT (Message Queue Telemetry Transport) and CoAP (Constrained Application Protocol). These lightweight protocols are designed for IoT devices and reduce the overhead associated with data transmission. The system may also incorporate advanced scheduling algorithms to prioritize critical data and reduce the frequency of transmissions, further optimizing energy use. This component plays a critical role in the IoT weather station, as it significantly impacts overall power consumption.

B. Development Lifecycle

The development lifecycle of an energy-neutral weather station focuses on ensuring that the station can operate autonomously, relying entirely on harvested energy, regardless of its duty cycle. This concept of "energy neutrality" developed as researchers looked for alternatives to battery-powered IoT devices, which often faced limitations due to energy depletion. Energy neutrality in IoT systems is achieved when the device functions sustainably using harvested energy customized to its application and deployment site. This structured development life cycle ensures the creation of a reliable, sustainable weather station customized to precision agriculture while addressing energy and maintenance challenges.

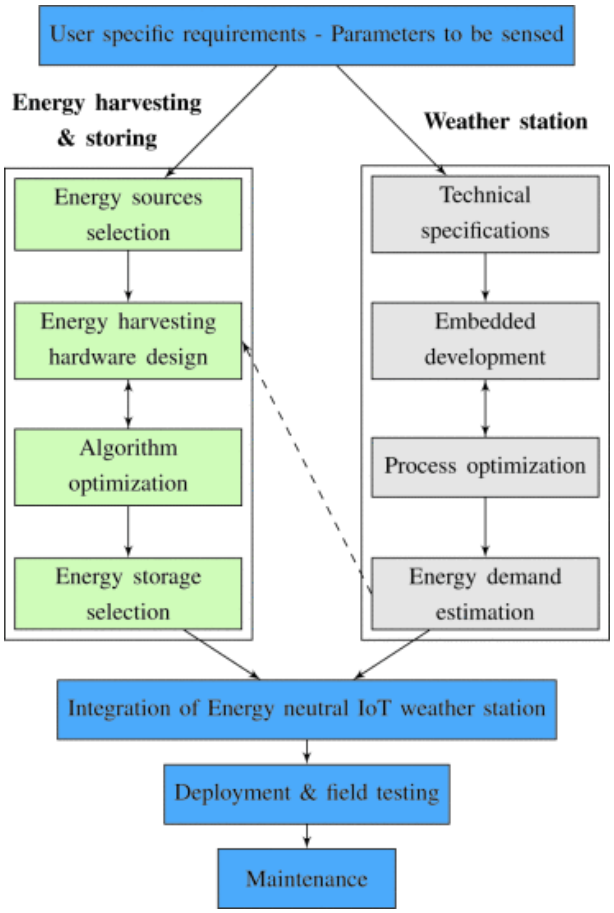

Figure 2.2: Shows the Proposed Development Life Cycle of IoT based Weather Station

(Retrieves from Energy-neutral weather stations for precision agriculture: challenges and approaches)

C. Key Design Challenges

The development of an energy-neutral weather station must overcome several key design challenges. Identifying an energy source is important, with solar and wind energy being the most popular options. However, these sources are unpredictable and seasonality, requiring careful consideration of panel size, efficiency, and cleanliness in addition to robust energy storage systems such as lithium-ion batteries, lead-acid batteries, or supercapacitors. Each storage has trade-offs with factors like temperature tolerance, energy density, and lifespan. Sensor optimization is power consumption can be minimized by reducing sampling rates data or using low-energy sensors based on environmental forecasts to save energy. Similarly, utilizing low-power microcontrollers with effective sleep cycles and external interruptions can optimize energy use, supported by energy-aware algorithms can customize data collection and processing. The communication block, being the most energy-intensive, requires selecting appropriate communication technologies such as LPWAN, Wi-Fi or GPRS and efficient scheduling of data transmission can significantly impact energy consumption. These challenges highlight the need for multi-disciplinary optimization of hardware, software, and processes to achieve energy neutrality.

D. Best Practices in Developing an Energy Neutral Weather Station

By selecting the best practices in the development process, we can develop a life cycle. A key consideration is the trade-off between accuracy and energy consumption. Choosing between high-accuracy sensors which consume more energy and low-power sensors which are less accurate to reduce the energy burden and simplify the energy harvesting process. Maintenance and cost are another important factor, especially devices in remote locations should be robust and easy to maintain. Components should be durable, cost-effective, and widely available to ensure the system lifespan and ease to replace, reducing maintenance costs. Additionally, balancing the data transfer rate based on energy availability ca reduce power consumption with scheduling algorithms like Torpor help manage transmission rates in accordance with the energy supply. It's because not all data needs to be sent frequently. A modular design is also recommended, where the hardware and software easily add or remove sensors based on the farmer's specific requirements. A modular design in both hardware and energy management ensures that the weather station can be adapted while maintaining energy consumption and reducing costs. Developing energy autonomous weather stations enables affordable, low-maintenance local weather data collection for farmers.

Development of a Weather Station Using IoT Platform Based Arduino Integrated with a Control System for Smart Agriculture Applications

Global warming has affected agriculture, caused uncertainty and increased vulnerability in farming practices like Climate-Smart Agriculture (CSA). Climate-smart agriculture (CSA) is a comprehensive strategy for managing landscapes, including croplands, livestock, forests, and fisheries, to tackle the interconnected challenges of food security and climate change (Climate-smart agriculture, 2024). Key factors like temperature, rainfall, humidity, CO2 levels, and solar radiation are a critical role in farming, requiring precise monitoring for improved resource management. Weather stations collect meteorological data using sensors and serve as tools for research, analysis, and decision-making in agriculture. This study aims to design an affordable weather station integrated with a control system capable of executing agricultural tasks based on predefined settings and transmitting plant-related environmental data to a cloud platform for remote access. So, this part will include multiple purposes such as environmental monitoring, plant monitoring, irrigation monitoring and controlling, CO2 monitoring and controlling and pesticide injection.

A. IoT-based Weather Stations

These systems combine multiple kinds of sensors that observe numerous climatic variables such as temperature, humidity, barometric pressure, wind speed and direction, rainfall and air quality. All data collected by the sensors and can be transferred to remote cloud platforms. Examples for remote analysis and decision making are ThingSpeak, Arduino Cloud and more. Advanced data analytics can generate forecasts and identify patterns by analyzing historical and real time environmental data. This very benefit farmers for their agricultural activities, improves irrigation and minimizes risks posed by extreme weather. Smart Systems can be integrated with weather stations such as automated irrigation systems, greenhouse and pest control mechanisms. Many IoT weather stations operate on renewable energy, such as solar power, making them sustainable and suitable for deployment in remote areas. The modular design allows users to custom the system to specific needs by adding or removing sensors, making it flexible for various applications. IoT-based weather stations are important in addressing modern challenges like climate change, resource optimization, and sustainable development, with applications spanning agriculture, environmental monitoring, and urban planning.

B. Design and Development of Control System

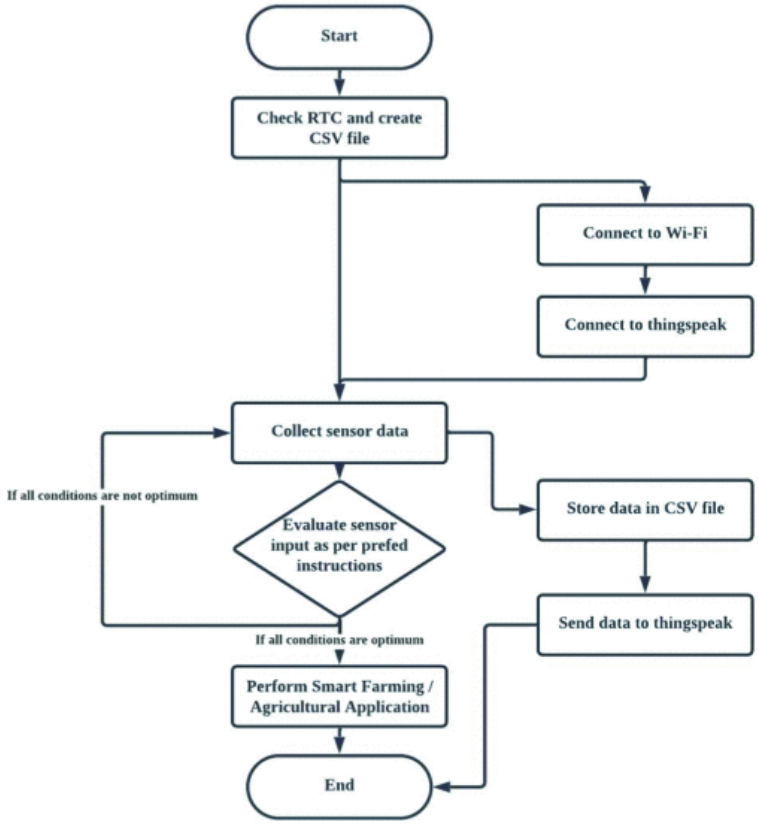

Figure 2.3: Shows the Block Diagram of Control System for Development of a Weather Station Using IoT

(Retrieves from Development of a Weather Station Using IoT Platform Based Arduino Integrated with a Control System for Smart Agriculture Applications)

In figure 3, the block diagram shows the data flow process of the Smart Agriculture Weather Station system. The Arduino Mega connects with SD card shield to make sure on-device data storage while establishing a Wi-Fi connection for data transmission to the ThingSpeak cloud platform. Once connected, the Arduino Mega connects with ThingSpeak platform using a unique Channel ID and API key. Following successful connectivity, the system starts to collect data from the sensors, then it's ready to be program to read and store data values at intervals of one minute. Then, the sensors provide raw data which is converted into their measurement units such as rainfall in millimeters (mm), wind speed in meters per second (m/s), wind direction in degrees, solar irradiation in watts per square meter (W/m^2), and CO_2 concentration in parts per million (PPM).

The processed data is then stored locally on the memory card and uploaded to the ThingSpeak platform. Next the collected data will be evaluated based on predefined conditions and instructions. The system assesses whether the measured parameters meet optimal limit. The Arduino Mega activates relays to perform necessary smart agricultural interventions, such as irrigation, ventilation, or other climate control actions. Following the completion of the steps, the process restarted, making sure a continuous loop of monitoring, analysis and responsive adjustments for effective agricultural management. This iterative system is made real-time data-driven decision-making for sustainable smart farming practices.

The complete structure of this self-sustaining system is composed of the following parts:

1. **Data Collection/Weather Station**

This system uses the Arduino Mega 2560 Rev 3 as its core controller because of GPIO pins and memory capacity, making it suitable for managing multiple components. The Arduino ecosystem offers a wide range of compatible sensors and libraries, simplifying integration. The weather station features include wind direction, anemometer, pyranometer, rain gauge, temperature, and CO2 sensors. All these features continuously monitor environmental data from plants. These features send data to the microcontroller at regular intervals, where it is collectively processed by pre-programmed instructions.

2. **Control System**

The collected data is processed through event-driven programming instructions embedded in the Arduino Mega. These logic-based evaluations determine the required actions based on insights from ecohydrology studies of oil palm. The control mechanism is automated, with the Arduino Mega managing tasks via multiple relays to facilitate smart climate agriculture practices.

3. **Data Transmission**

The Cytron ESP8266 Wi-Fi shield connects the Arduino to Wi-Fi, includes sensor data transmission to ThingSpeak, a cloud platform for visualization and analysis. This data is accessible in real time through web and mobile platforms and can be exported for analysis using software like Excel, MATLAB, or LabVIEW.

4. Data Storage

An SD card shield is integrated with the Arduino to locally store data in the absence of Wi-Fi. The data is saved as a CSV text file, which can later be imported into Excel for analysis. Additionally, a Real-Time Clock (RTC) is included to timestamp the data, making sure it can be used effectively for weather or data analysis in the future.

5. Solar Charging System

The system includes a 24-hour backup powered by a battery charged through solar panels using an MPPT charge controller. This setup enhances solar panel efficiency and ensures continuous operation with minimal data loss, contributing to the system's sustainability and reliability.

C. Installation

Figure 2.4: Shows the 3D Design of Weather Station along with actual on-site Installation of Structure

(Retrieves from Development of a Weather Station Using IoT Platform Based Arduino Integrated with a Control System for Smart Agriculture Applications)

D. Result and Discussion

This study details the design, development, and deployment of an IoT-based weather station and control system at Universiti Malaysia Sabah to support Climate-Smart Agriculture (CSA). Built around the Arduino Mega, the low-cost system collects environmental data from sensors such as wind direction, anemometer, pyranometer, rainfall, and CO2 sensors. The system supports both cloud-based and local data storage, enabling real-time analysis and secondary processing with tools like MATLAB and LabVIEW. Logical conjunction in the control system allows automated decision-making, activating relays for smart applications like irrigation, pesticide spraying, and CO2 injection. The flexible and scalable design extends its utility beyond environmental monitoring to applications such as plant health and productivity enhancement. While the system emphasizes affordability and efficiency, further research is needed to refine sensor calibration and data accuracy. This innovative approach provides a cost-effective solution for precision agriculture and smart farming, aligning with the goals of sustainable and efficient agricultural practices.

Fig. 5.(a)
CO2 concentration graph

Show All

Fig. 5.(b)
Solar irradiation graph

Figure 2.5: Shows the Meteorological Data a and b

Fig. 5.(c)
Wind speed graph

Show All

Fig. 5.(d)
Wind direction graph

Figure 2.6: Shows the Meteorological Data c and d

Fig. 5.(e)
Rainfall graph

Figure 2.7: Shows the Meteorological Data e

(Retrieves from Development of a Weather Station Using IoT Platform Based Arduino Integrated with a Control System for Smart Agriculture Applications)

An IoT based Low-cost Weather Monitoring System for Smart Farming

Crop production is heavily influenced by local weather conditions, with most agricultural activities being sensitive to atmospheric factors such as wind speed, air temperature, humidity, and precipitation. Usually, weather forecasting relies on manual observations of wind and cloud patterns, but advancements in technology have led to the development of automated weather stations (AWS). However, the high cost of weather instruments, such as rain gauges, anemometers, and barometers, makes it difficult for farmers to implement these stations on their own fields. Traditional manual weather stations also require constant monitoring, which adds labor costs and increases the burden on farmers. To address these challenges, a low-cost automated weather monitoring and real-time alert system has been proposed. This system, solar-powered and equipped with various sensors (such as temperature, rain, humidity, wind speed, and CO2), minimizes manual intervention, provides real-time alerts, and allows remote control via GSM networks. The goal is to offer an affordable, automated solution for farmers to monitor weather conditions, make informed decisions, and reduce the impact of weather-related disasters on crops, all while enhancing the sustainability and profitability of their operations.

A. Materials and Methods

1. Hardware Components:

- Arduino Uno Microcontroller: Central control unit for managing sensor data and system operations.
- Sensors:
    a. DHT11 Sensor: Measures both temperature and humidity.
    b. FC37 Rain Drop Sensor: Detects rainfall.
    c. Pressure Sensor: Monitors atmospheric pressure.
    d. Light Sensor: Measures light intensity.
    e. CO2 Sensor: Tracks carbon dioxide levels.
    f. Anemometer: Measures wind speed.
    g. Wind Direction Detector: Identifies wind direction.
- I2C Liquid Crystal Display: Displays real-time weather data at the field site.
- GSM-900A Module: Sends weather data to users via the GSM network.
- Solar Panel: Provides power for system operation.

- Backup Battery: Ensures uninterrupted operation during low sunlight conditions.

2. **Software Tools:**
    - Arduino IDE: Used for programming the Arduino Uno and integrating the sensors.
    - Altium Designer: Used for designing circuits and system layout.
3. **Methods:**
    - Sensor Integration:
        - All sensors were connected to the Arduino Uno microcontroller.
        - The Arduino was programmed using the Arduino IDE to collect real-time weather data from the sensors.
    - Data Processing and Display:
        - Weather data captured by sensors were processed by the Arduino.
        - Real-time data was displayed on the I2C Liquid Crystal Display for on-site monitoring.
    - Remote Data Access:
        - Weather data were transmitted to a GSM-900A module.
        - Users could send a "STATE" message to the GSM module to receive weather updates on their mobile devices.
    - Power Supply:
        - The system was powered by a solar panel, with a backup battery ensuring uninterrupted operation in low sunlight conditions.
    - Automation:
        - The entire system was automated, requiring no human intervention for data collection, processing, or transmission.

B. Results and Discussion

This system had three main units, which were power unit, sensing unit and output unit. This project developed a low-cost, solar-powered weather monitoring and alert system. A 10W solar panel generated power, converting high irradiation (21.2V) into usable voltage for the system. A solar charge controller adjusted this to 14V to charge a 12V battery, which supplied 5V to the Arduino Uno microcontroller and sensors through a voltage regulator. The system used

various sensors such as DHT11 sensors that measured temperature and humidity, BMP180 sensor measured atmospheric pressure, FC37 rain sensor detected rain with three conditions, which is "RAIN," "RAIN WARNING," and "NOT RAIN". Other than that, light sensor that can assess light intensity in five conditions which is "BRIGHT," "DIM", MG811 CO2 sensor measured atmospheric CO2 levels, anemometer was measured wind speed using a cup design, wind vane was determined wind direction using eight infrared sensors. The Arduino Uno was programmed with the Arduino IDE to manage these sensors. Weather data was displayed on an I2C LCD and transmitted via a GSM-900A module to users' mobile phones. A user could request weather data by sending the keyword "STATE" via SMS, and the system would respond with real-time information. The setup was housed in a louvered Stevenson screen and implemented in a Smart Agro Tech Park for testing. Data included parameters like temperature, humidity, pressure, light intensity, rain, CO2 levels, wind speed, and wind direction. This system really helped farmers by providing real-time weather data, enabling them to make informed decisions to manage crops and soil while reducing weather-related risks. Compared to alternatives like Wi-Fi or Bluetooth, GSM offered a longer transmission range of up to 35 km, making it suitable for outdoor use. Using solar power also makes sure cost-effectiveness and sustainability, benefiting farmers while reducing their dependency on electricity.

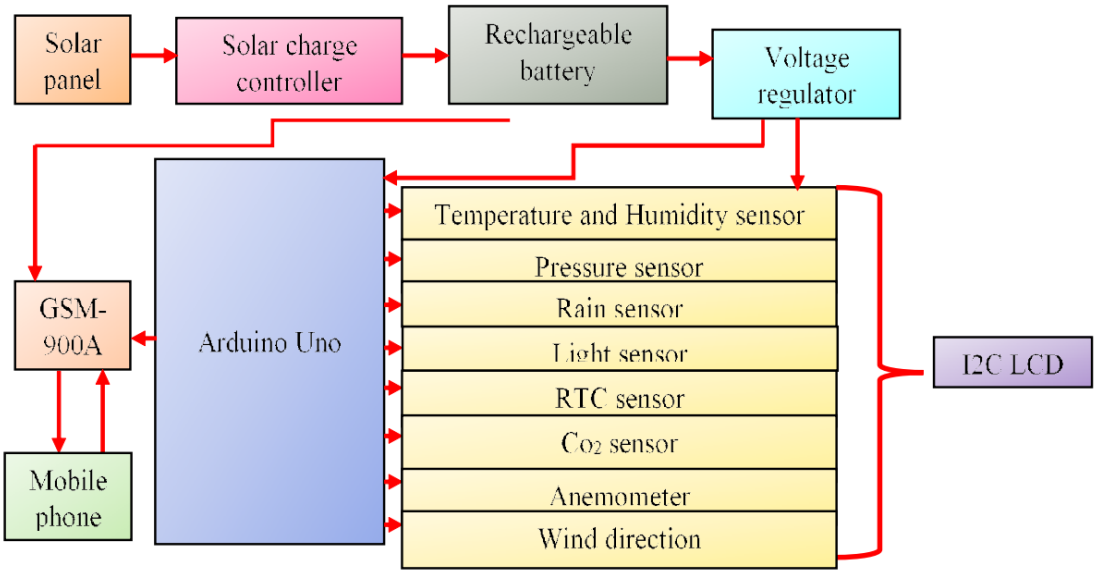

Figure 2.8: Shows the Block Diagram of the Smart Farming System

Figure 2.9: Shows the Output Unit of the Smart Farming System

Figure 2.10: Shows the Smart Weather Monitoring System

Figure 2.11: Shows the Low-cost Wind Vane of the Smart Farming System

(Retrieves from An IoT based Low-cost Weather Monitoring System for Smart Farming)

2.3 Comparison between Existing Systems

Feature	Energy Neutral Weather Station for Precision Agriculture	Development of a Weather Station Using IoT Platform Based Arduino Integrated with a Control System for Smart Agriculture Applications	An IoT based Low-cost Weather Monitoring System for Smart Farming
Provide real time update	No	Yes	Yes
Store data on history management	No	No	Yes
Application on mobile	No	No	Yes
Low-cost	Yes	No	No
Data display by computer	Yes	Yes	Yes

Table 2.1: Shows the Comparison between Existing Systems

2.4 Comparison between Proposed System and Existing Systems

Feature	Energy Neutral Weather Station for Precision Agriculture	Development of a Weather Station Using IoT Platform Based Arduino Integrated with a Control System for Smart Agriculture Applications	An IoT based Low-cost Weather Monitoring System for Smart Farming	Smart Agriculture Weather Station Powered by IoT
Provide real time update	No	Yes	Yes	Yes
Store data on history management	No	No	Yes	Yes
Application on mobile	No	No	Yes	Yes
Low-cost	Yes	No	No	Yes
Data display by computer	Yes	Yes	Yes	Yes

Table 2.2: Shows the Comparison between Proposed System and Existing Systems

2.5 Tools and Technology used in Development of the Proposed System

- ESP8266 Microcontroller
- Sensors:
    - Temperature Sensor
    - Humidity Sensor
    - LDR Sensor
    - Rain Sensor
- Arduino IDE
- Arduino IoT Cloud
- Libraries and Drivers:
    - Esp8266 Board
    - DHT Sensor Library
    - Jumper Wires
- Waterfall Model

2.6 Summary

Chapter 2 explores existing smart farming systems and their limitations for small-scale organic farmers that highlighting challenges such as high costs, lack of accessibility, and limited features like real-time monitoring, mobile access, and trend analysis. The proposed system aims to address these gaps by offering an affordable, user-friendly IoT-based solution that monitors key environmental factors like temperature, humidity, light intensity, and rainfall in real-time.

Chapter 3: Requirement Analysis and Design

3.1 Introduction

In this chapter, we cover the aspects of requirement analysis and the design process for the Smart Agriculture Weather Station Powered by IoT. As modern agriculture depends on data-driven knowledge to optimize productivity and sustainability, the IoT enabled system has become important. By providing a dependable, real-time weather monitoring system that improves operational efficiency and decision-making, this project aims to address the difficulties faced by farmers and other agricultural stakeholders. This chapter focuses on defining the project goal and identifying the requirement needed of each stakeholder and creating a robust framework for the whole development process. Understanding the requirement at this stage is crucial, as it directly affects the system's capacity to meet its intended functionalities and provide value to end users. This chapter establishes the foundation for effective and efficient system design by thoroughly analyzing user requirements, environmental considerations, and technology limitations. This project uses waterfall techniques to guarantee a systematic and well-organized approach. The waterfall model allows a step-by-step transition from one development phase to the next and the waterfall model makes sure that every step is fully comprehended and finished before going on. This structured approach provides clarity, precision, and a systematic pathway, reducing the likelihood of errors and oversights during development. This chapter provides the basis for other stages such as implementation and testing by defining a thorough requirement analysis and design approach. To guarantee that the finished system has the required performance, dependability and usefulness that provides a clear and manageable development roadmap for the Smart Agriculture Weather Station. The concepts and insights created here will direct the project to effectively accomplish its goals and produce significant outcomes for the agriculture industry.

3.2 Requirement Analysis

3.2.1 Software Requirements

Software	**Description**
Arduino IDE	The Arduino IDE allows users to write, compile, and debug code to the ESP8266 microcontroller using its user-friendly interface. The IDE works seamlessly across Windows, macOS, and Linux, ensuring development flexibility.
Arduino IoT Cloud	It provides a centralized platform for managing, storing, and accessing sensor data collected by the system. Graphical dashboards allow users to monitor real-time data trends. Farmers can access data and control system settings from any internet-connected device through the cloud platform.
ESP8266 Board	The firmware interfaces with sensors to collect raw environmental data. The ESP8266 communicates with the IoT cloud using Wi-Fi or cellular networks that managed by the firmware.
DHT Sensor Library	Prebuilt libraries for sensor modules. For example, DHT for humidity/temperature sensors.

Table 3.1: Lists of the Software Requirements with Description

3.2.2 Hardware Requirements

Hardware	Description
ESP8266 Microcontroller	• Collects data from connected sensors and processes it for transmission. • Provides Wi-Fi and Bluetooth connectivity for real-time data communication with the Arduino IoT Cloud. • Low power consumption, making it suitable for remote and energy-efficient applications.
DHT11 (Temperature and Humidity sensor)	Temperature Sensor: • Measures ambient temperature to monitor weather conditions affecting crops. • Wide measurement range suitable for agricultural applications. • High accuracy and quick response time. Humidity Sensor: • Measures air humidity to help farmers monitor conditions conducive to crop health. • Accurate relative humidity measurement.
LDR Sensor	• LDR Sensor is an electronic component that react to the brightness by changes of its resistance. • The sensor commonly used in circuits designed to detect and respond to light changes. • The more intense the light in the LDR, the lower the resistance.
Rain Sensor	• Detects presence and intensity of rain using a conductive plate or resistive sensor. • Works by measuring changes in resistance or voltage when water touches the sensor surface. • Provides digital (rain detected: HIGH/LOW) or analog (rain intensity level) output.
Jumper Wires	Facilitates connections on the breadboard for circuit prototyping.

Table 3.2: Lists of the Hardware Requirements with Description

3.3 System Design

System design is a critical phase in the development of the Smart Agriculture Weather Station Powered by IoT. This phase involves translating the requirements identified during the analysis stage into a structured blueprint for implementation.

3.3.1 Use Case Diagram

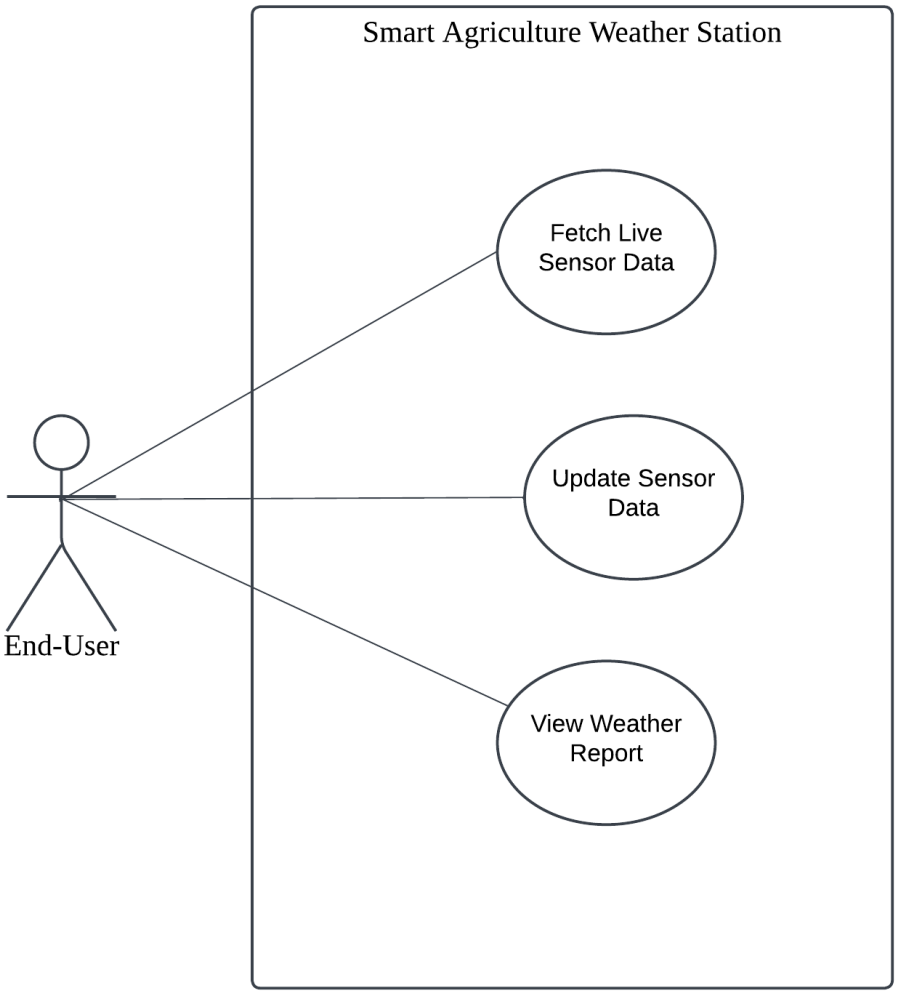

Figure 3.1: Shows the Use Case Diagram of Proposed System

Use Case Name	Smart Agriculture Weather Station
Actor	End-User, Proposed System
Flow of Event	1. The sensors (temperature, humidity, light intensity, rainfall) has been fetch environmental data in real-time. 2. The ESP8266 microcontroller processes the collected data and transmits data to the IoT cloud. 3. After that the data is updated. 4. Data is displayed in a user-friendly format, such as graphs or numerical values, for easy interpretation by Arduino IoT Cloud. 5. The end-user selects an option to view historical data from the Arduino IoT Cloud. 6. The system queries the IoT cloud database for stored data.
Exit Condition	The retrieved data is displayed and can been view, allowing the end-user to analyze trends and make informed decisions.

Table 3.3: Shows the Use Case Description

3.3.2 System Architecture

The block diagram illustrates the architecture of a weather monitoring system, detailing the flow of input data from sensors, the processing unit's operations, and the corresponding output.

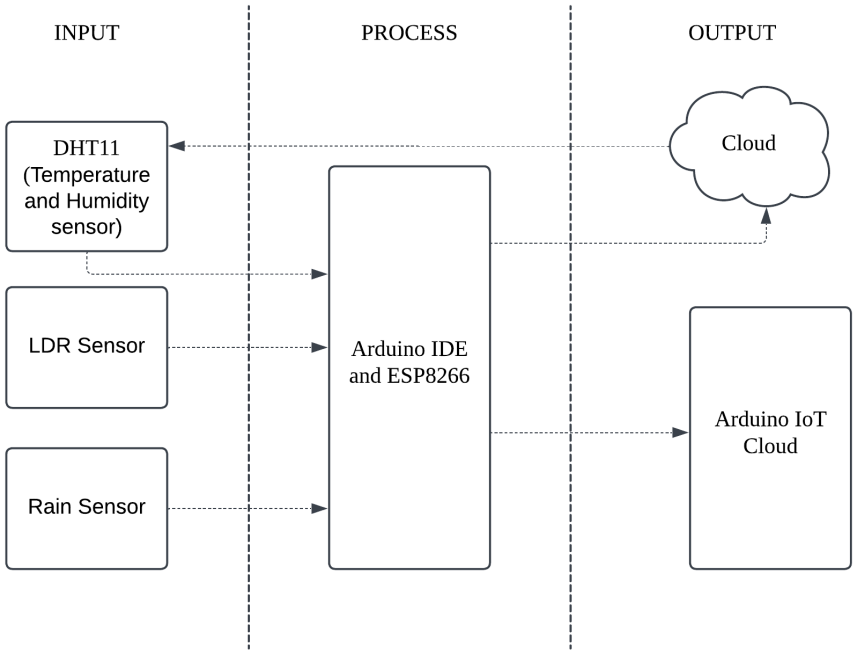

Figure 3.2: Shows the Block Diagram of Proposed System

Figure 16 shows the input stage consisting of three sensors which are DHT11, LDR Sensor and Rain Sensor. DHT11 measures temperature and humidity providing crucial atmospheric data. The LDR Sensor is used to detect the intensity of light. Meanwhile, the Rain Sensor detects presence and intensity of rain using a conductive plate or resistive sensor. The processing stage is managed by the Arduino IDE and ESP8266 microcontroller. ESP8266 serves as the system's brain by using the Arduino IDE to process the incoming sensor data efficiently. It performs computations and prepares the data for output while ensuring seamless communication between the input and output stages. In the output stage, the system delivers information via multiple channels. Besides, the processed data is transmitted to the Arduino IoT Cloud for remote monitoring and data analysis through web and mobile dashboards. Additionally, cloud storage ensures secure saving of data for future use to make the system more reliable for long-term weather tracking. This comprehensive architecture ensures real-time environmental monitoring, both locally and remotely, enhancing accessibility and reliability for users.

3.3.3 Data Flow Diagram

The Data Flow Diagram (DFD) shows the flow of weather data from various sources to its end-users:

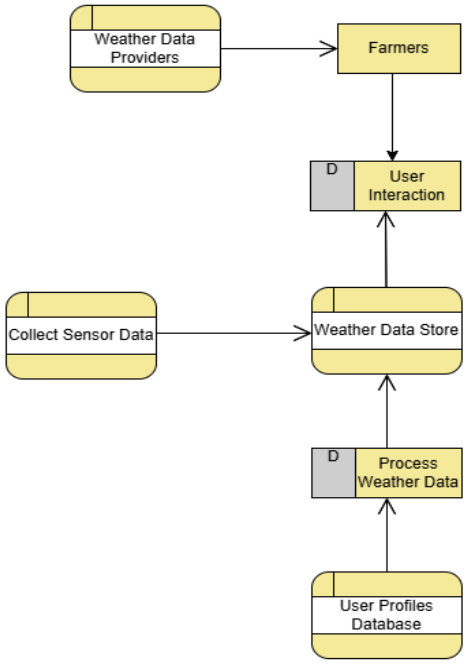

Figure 3.3: Shows the Data Flow Diagram of Proposed System

The DFD represents a system designed to collect, process, and deliver weather data to farmers. Firstly, Weather Data Providers supply weather information directly to farmers. Then, Collect Sensor Data gathers localized environmental data from sensors such as temperature, humidity, light intensity and rainfall. Both external data and sensor data are stored in the Weather Data Store to a centralized repository for organizing and managing all weather-related information. The stored data is then processed through the Process Weather Data module which analyzes the raw information to generate meaningful insights or modify plans to the needs of farmers. The system uses the User Profiles Database which contains farmer-specific details such as location and preferences to ensure the processed weather data is relevant and personalized. Finally, farmers interact with the system through the User Interaction module which serves as an interface such as a web and mobile dashboards that allows them to access actionable insights. This system streamlines the flow of accurate and customized weather information that can help farmers make informed decisions for better agricultural outcomes.

3.3.4 Entity-Relationship Diagram

The Entity-Relationship Diagram (ERD) represents how different entities interact in Smart Agriculture Weather Station system.

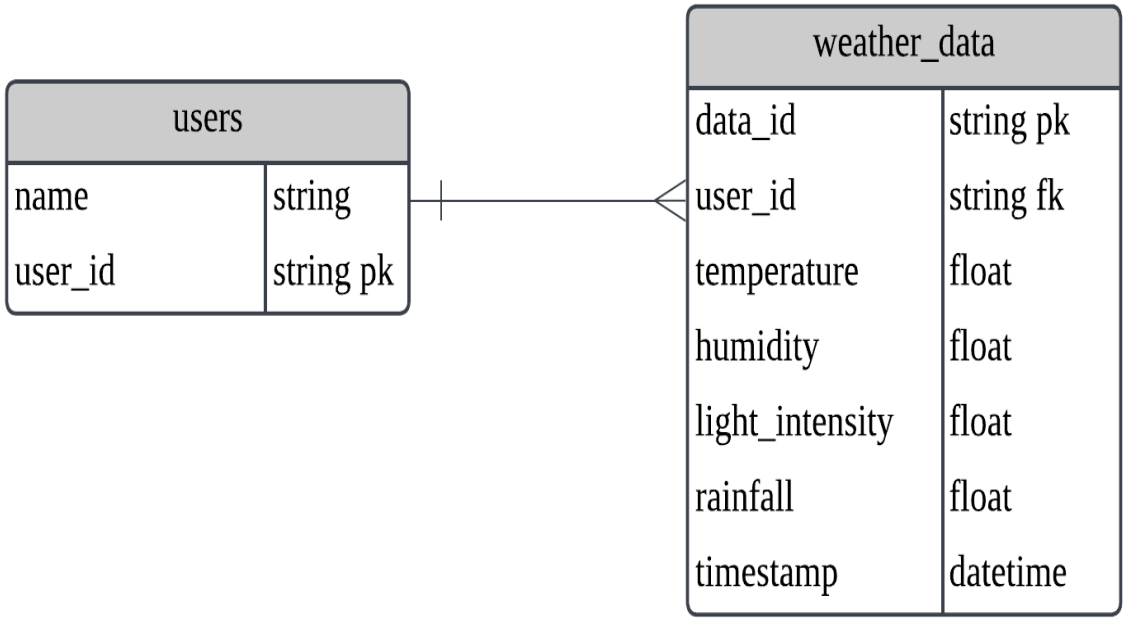

Figure 3.4: Shows the Entity-Relationship Diagram of Proposed System

Figure 3.4 shows the users entity stores details about farmers or system users, including their unique ID and name. It consists of two main entities: users and weather_data. The users table stores user-related information, including user_id as the primary key, which uniquely identifies each user, and name to store the user's name. The weather_data table is used to record various environmental measurements such as temperature, humidity, light_intensity, rainfall, and a timestamp to log when the data was captured. Each record in weather_data is uniquely identified by a data_id, which serves as the primary key, while user_id in this table acts as a foreign key referencing the user's table. This establishes a one-to-many relationship between users and weather_data, indicating that one user can generate multiple entries of weather data over time. This structure not only supports efficient storage and retrieval of sensor readings for individual users but also allows for historical tracking and personalized analysis of weather conditions based on user-specific data.

3.3.5 Prototype Design

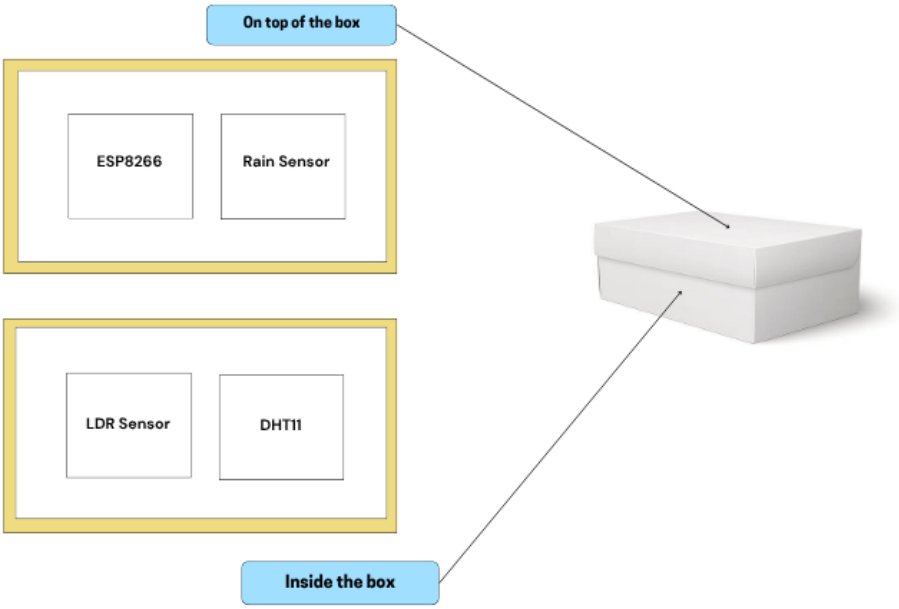

Figure 3.5: Shows the Design Prototype of the Proposed System

In figure 3.5 shows the prototype design of this Smart Weather Monitoring system that well organized to ensure good performance of the sensor and the durability. Each component in the system placed in a best suites location to its function by separating components that need direct exposure or inside for protection from the elements. The top surface on the box is the ESP8266 microcontroller along with the rain sensor. The microcontroller is exposed to the open air to optimizes its Wi-Fi performance where it can enhance its ability to transmit data continuously to the Arduino IoT Cloud. Keeping the rain sensor on the top the box allows it to function correctly during precipitation events. The LDR and DHT11 sensors are stored inside the casing to maintain stability in sensor performance. The LDR sensor inside is aligned with an aperture that admits enough light for accurate sensing. The prototype making it well-suited for continuous outdoor weather monitoring by combining reliability and environmental resistance. The arrangement of components highlights the importance of planning in balancing performance and environmental demands.

3.4 Summary

In this chapter, it recaps about how this proposed system developed by following the Waterfall Methodology. The tools and technologies utilized that including IoT sensors, Arduino IDE and Arduino IoT Cloud are selected for their affordability and compatibility with small-scale farming needs. Additionally, the proposed hardware components including sensors for temperature, humidity, light intensity, rainfall and along with the ESP8266 microcontroller are designed to deliver accurate and reliable data collection. This chapter ensuring that the weather station meets all the requirements needed to improve agricultural productivity through real-time monitoring, data-driven decision-making and sustainable farming practices.

Chapter 4: Development

4.1 Introduction

In this Chapter 4 designates the details of the development and implementation phase of the Smart Agriculture Weather Station Powered by IoT project. This chapter presents in detail how the system was built and made operational. All implementations will be designs that follow all the requirements needs. Each component is developed carefully to ensure processing and transmission of the data smoothly to the IoT cloud platform. Also, the details on how the hardware and software were assembled one by one were used will explain in this chapter. In addition, the step-by-step integration process between the sensors, microcontroller and cloud platform is presented clearly to provide a comprehensive understanding of the development.

4.2 Tools Deployment

The tools used in this system are divided into two main parts which are hardware and software components. For the hardware component, the following are the tools used:

- ESP8266 Microcontroller
- DHT11
- LDR Sensor
- Rain Sensor
- Jumper Wires

For the software component, all the software that will be used to develop Smart Agriculture Weather Station Powered by IoT needs to be installed and configured before used. Below is the software component needed in the system implementation:

- Arduino IDE
- Arduino IoT Cloud
- Libraries:
  - Esp8266 Board
  - DHT Sensor Library

4.3 Hardware Deployment

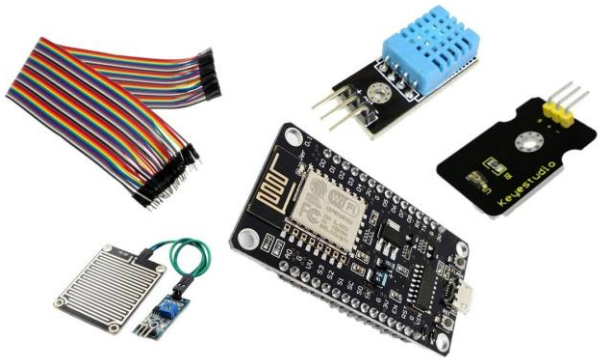

Figure 4.1: Shows the Component of the Proposed System

This figure just shows the component of the proposed system.

4.3.1 Hardware Setup Phase

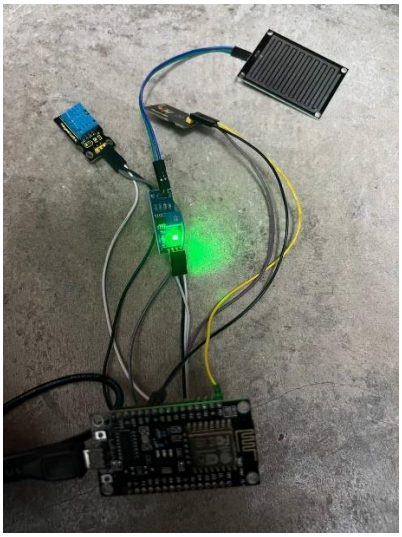

Figure 4.2: Shows the Hardware Final Setup Proposed System

Each sensor used in the system which is DHT11 for temperature and humidity, LDR sensor for light intensity and rain sensor was examined on its own to verify it perform correctly before being merged with the rest of the hardware. Real-time data from each sensor was displayed and evaluated on the Serial Monitor to validate their correct operation under different inputs. For example, exposing the LDR sensor to different lighting conditions resulted in changes to its light intensity readings. Meanwhile, spraying water on the rain sensor immediately triggered rain detection.

4.4 Software Deployment

This section explains the methods applied to configure and run the software used in the Smart Agriculture Weather Station. The system was built using the Arduino IDE for code compilation and the Arduino IoT Cloud for real-time data display to ensure the system work smoothly.

4.4.1 Arduino IDE

This is the several key steps for setup Arduino IDE:

1. Installation: Arduino IDE

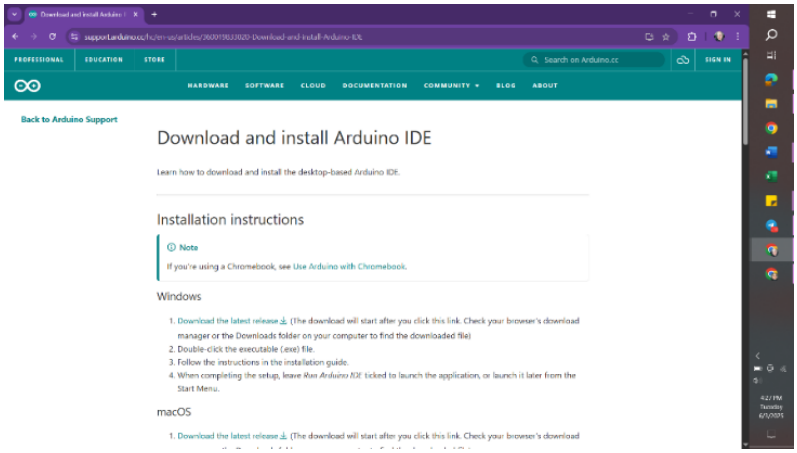

Figure 4.3: Shows the Website Page for Installation Arduino IDE

2. Board Configuration: The ESP8266 board was added to the Board Manager using the URL: https://arduino.esp8266.com/stable/package_esp8266com_index.json

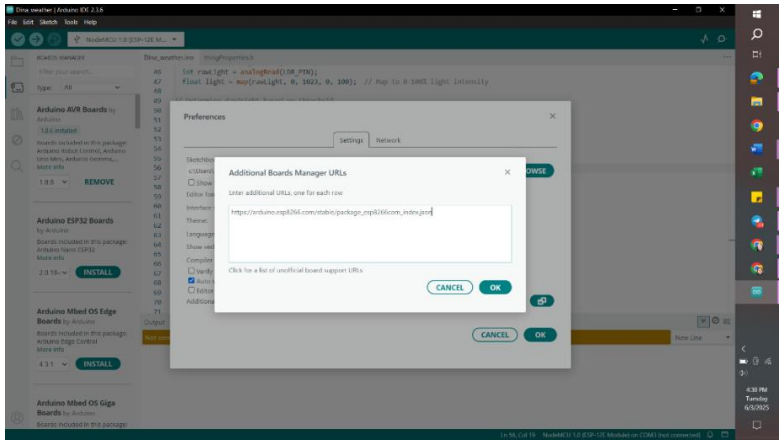

Figure 4.4: Shows the ESP8266 board was added in Arduino IDE

3. Library Installation: Required libraries such as DHT Sensor Library, Adafruit Unified Sensor, and Arduino IoT Cloud were installed through the Library Manager

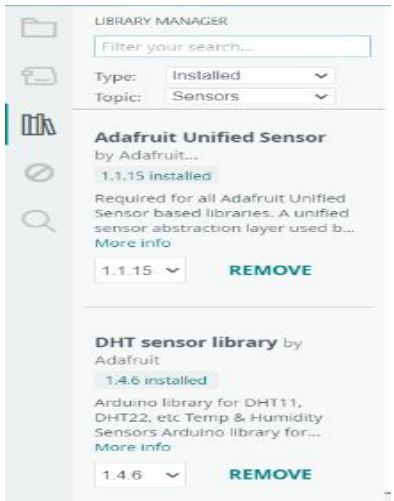

Figure 4.5: Shows the Library Installed in Arduino IDE

4. Port and Board Selection: The correct port and board type (NodeMCU 1.0 / ESP8266) were selected before uploading.

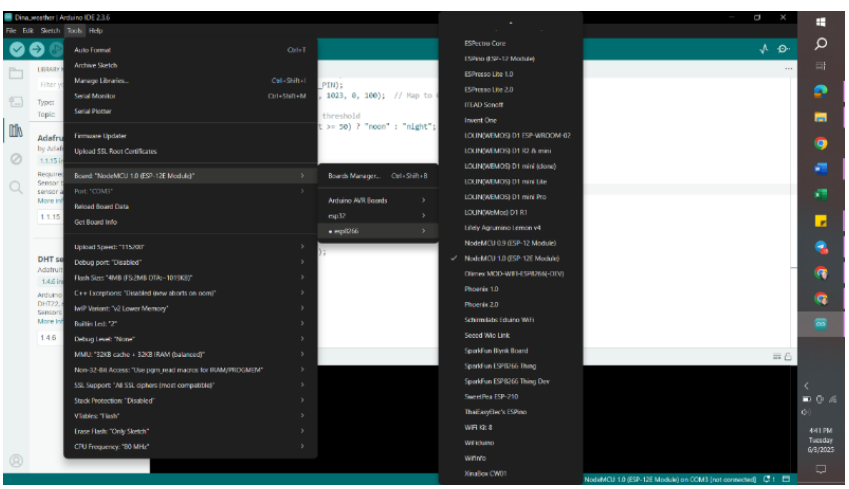

Figure 4.6: Shows the Port and Board Selection in Arduino IDE

5. Code Upload: The main sketch included the logic to read temperature, humidity, light intensity, and rainfall values, which were uploaded to the microcontroller.
6. Serial Monitoring: Serial Monitor was used to debug sensor values and confirm successful readings before connecting to the cloud.

4.4.2 Arduino IoT Cloud

This is the several key steps for setup Arduino IoT Cloud:

1. Thing Creation: A new "Thing" was created on the Arduino IoT Cloud dashboard to represent the Smart Weather Station.
2. Variable Configuration: IoT variables such as temperature, humidity, light detector, and rain detector were defined as cloud variables and matched in the code.

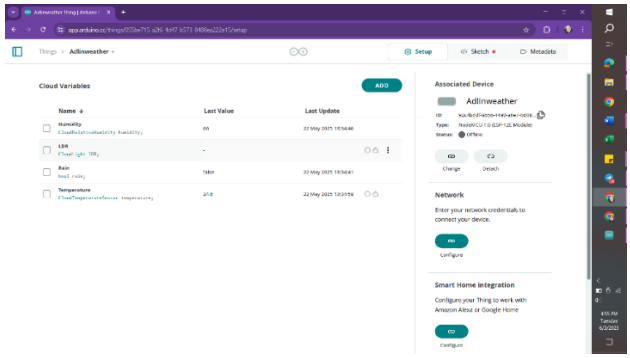

Figure 4.7: Shows the Variable Configuration in IoT Cloud Platform

3. Device Registration: The ESP8266 board was registered and linked to the created Thing by using a unique Device ID and Secret Key.
4. Dashboard Setup: Widgets such as gauges and labels were added to visualize sensor data in real time.

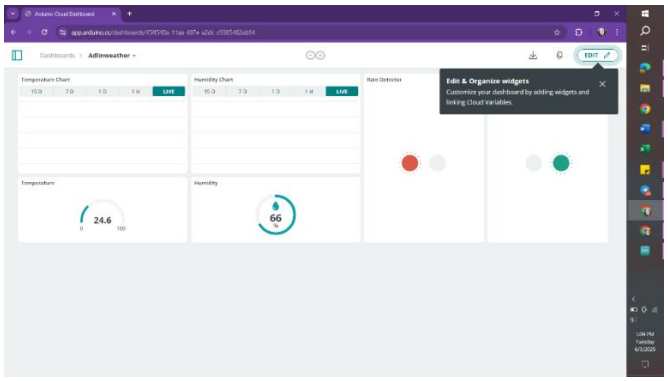

Figure 4.8: Shows the Dashboard Setup in IoT Cloud Platform

5. Cloud Synchronization: The code was automatically synchronized with the defined IoT variables using the thingProperties.h file, which was generated by the cloud interface.
6. Deployment Testing: Once deployed, live data was successfully received from the sensors and displayed on the IoT Cloud dashboard.

4.5 Summary

This chapter has explained the development process for hardware and software components of the Smart Agriculture Weather Station. The IDE provided the environment for development and the IoT Cloud was responsible for displaying sensor data online. These platforms worked together to create a fully functional system that collect and display live data on weather conditions such as temperature, humidity, light intensity, and rainfall. Cloud integration enhances system efficiency and enables farmers to be accessed remotely, promoting automation and smarter agricultural practices that align with the project's goals.

Chapter 5: Testing

5.1 Introduction

System testing plays a vital role in verifying the functionality and reliability of a system before deployment. It helps verify that both hardware and software components align with the goals of the project and fulfil the system requirements. This chapter is a detailed overview of the testing phases procedures applied to the Smart Agriculture Weather Station. It ensures that all critical components operate dependably, delivering accurate sensor data, stable connectivity, and a responsive interface. The chapter describes various testing methods implemented from component-level tests on specific sensors to full-system integration checks to ensure that all subsystems work together effectively. A well-documented testing phases also provides evidence of the system's functionality and quality.

5.2 Functional Testing

Functional testing aims to ensure that all system features align with the expected outputs for given inputs. It includes verifying correct sensor connections, ensuring accurate data readings, and confirming proper transmission to the cloud.

5.2.1 Component Testing

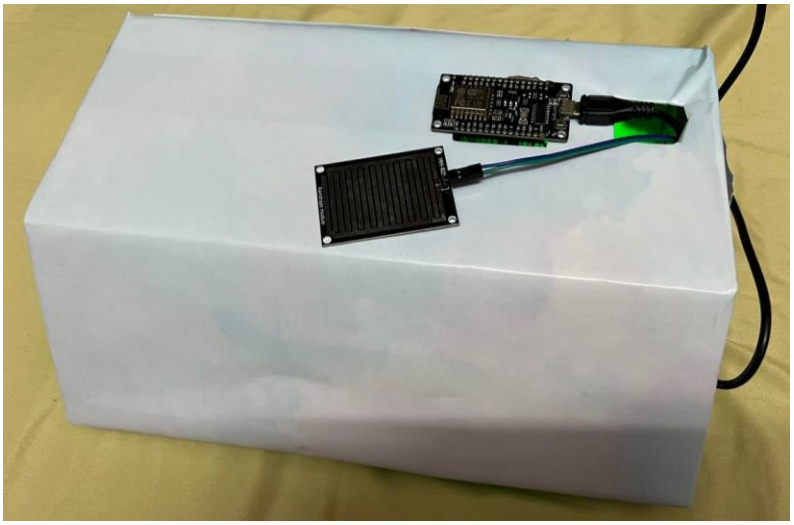

Figure 5.1: Shows the on Top of the Box that Complete Design of the Proposed System

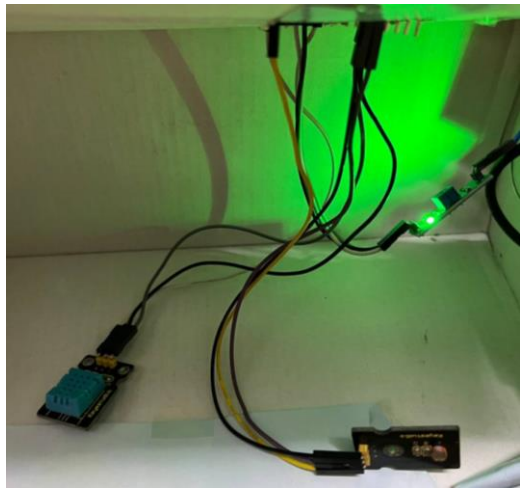

Figure 5.2: Shows the Inside of the Box that Complete Design of the Proposed System

Figure 5.1 and 5.2 shows the final hardware design of the Smart Agriculture Weather Station is organized using a structured enclosure where sensor components are strategically placed based on environmental exposure and functional needs. The system uses a shoes box to house and protect the components. On the top of the box, the ESP8266 microcontroller and the rain sensor are mounted. This placement ensures that the ESP8266 maintains a strong Wi-Fi connection for cloud communication, while the rain sensor is fully exposed to open air to accurately detect rainfall without obstruction. Inside the box, the DHT11 sensor and the LDR (Light Dependent Resistor) sensor are installed. DHT11 is used to measure temperature and humidity and is placed inside to protect it from direct exposure to water, which could affect its performance or damage it. The LDR is positioned in a way that allows it to receive ambient light through a small opening or transparent cover, enabling accurate light intensity readings while keeping it shielded from weather conditions. This design ensures both functionality and protection that allows accurate environmental data collection while maintaining durability in outdoor settings. The layout also promotes efficient wiring, easy maintenance and compact deployment that suitable for farmers.

5.2.2 System Testing

After verifying individual components, a completed system was tested collectively to ensure seamless integration. All sensors were connected and wired to the ESP8266 microcontroller, followed by uploading the final code through Arduino IDE. Real-time values were reviewed on the Serial Monitor to validate their performance and check for consistent updates.

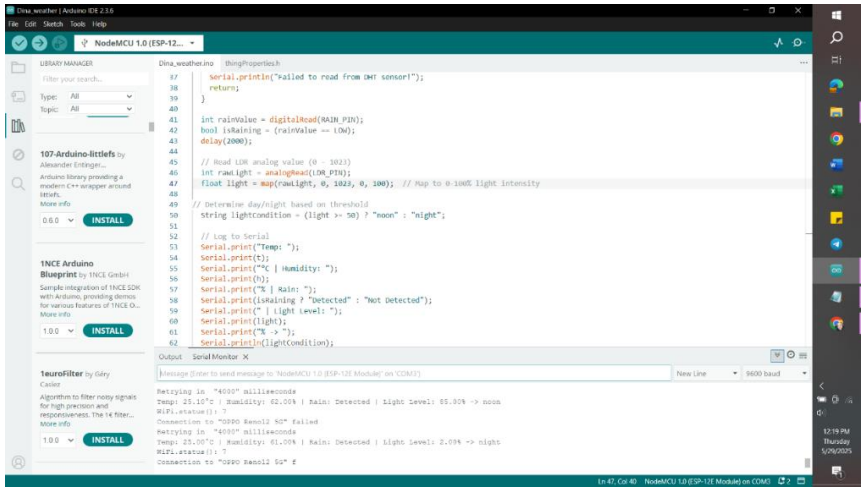

Figure 5.3: Arduino IDE page

Figure 5.4: Shows the Serial Monitor Sensor Reading 1 in Arduino IDE

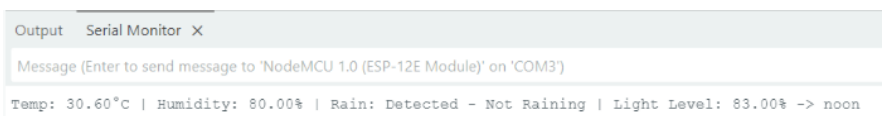

Figure 5.5: Shows the Serial Monitor Sensor Reading 2 in Arduino IDE

The Serial Monitor output shown in the image represents real-time environmental data collected by the Smart Agriculture Weather Station. Figure 5.4 displays a temperature reading of 30.70°C and a humidity level of 80.00%, indicating warm and moist atmospheric conditions. The rain sensor returns a raw digital value of 0, which the system interprets as “Rain: Detected – Raining,” confirming active rainfall. When the raw value is 1, it indicates no rain as the sensor detects no moisture on its surface that has been shown in figure 5.5. Additionally, the light level measured by the LDR sensor is 20.00% which is below than 50%, a value below the threshold for daylight, resulting in the classification “night”, while in figure 2 display its “noon” because the LDR sensor value above 50% which is 83.00%. This combination of high humidity, moderate temperature, rainfall detection, and low light intensity accurately reflects a nighttime rain event, showcasing the system’s ability to interpret environmental conditions precisely and communicate them effectively through the Serial Monitor.

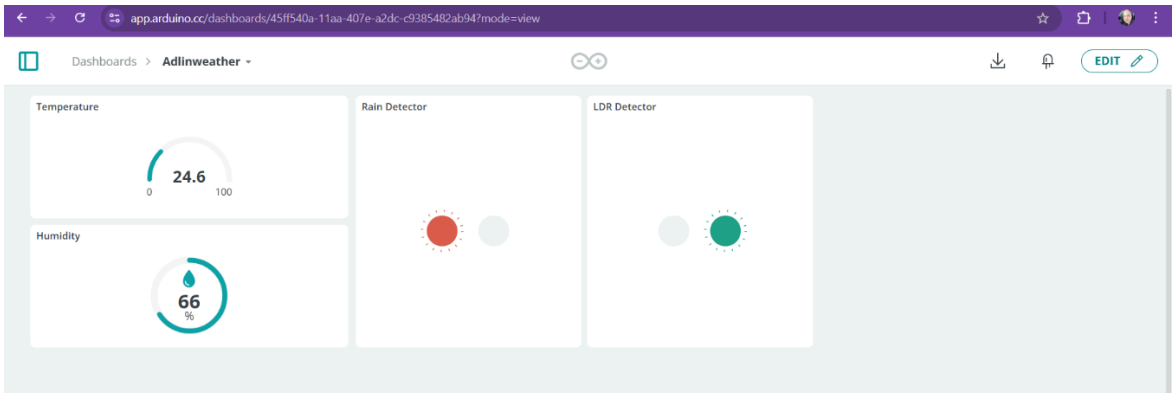

Figure 5.6: Shows the Web Dashboard in Arduino IoT Cloud of Proposed System (Adlinweather)

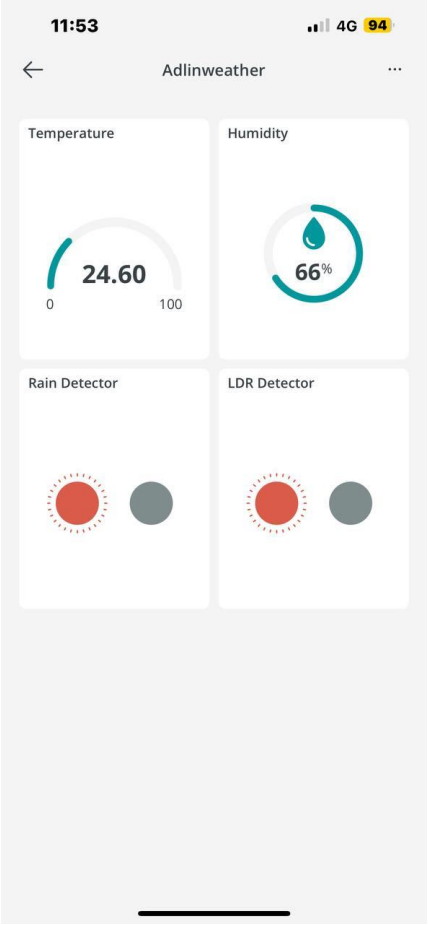

Figure 5.7: Shows the Mobile Dashboard in Arduino IoT Cloud of Proposed System (Adlinweather)

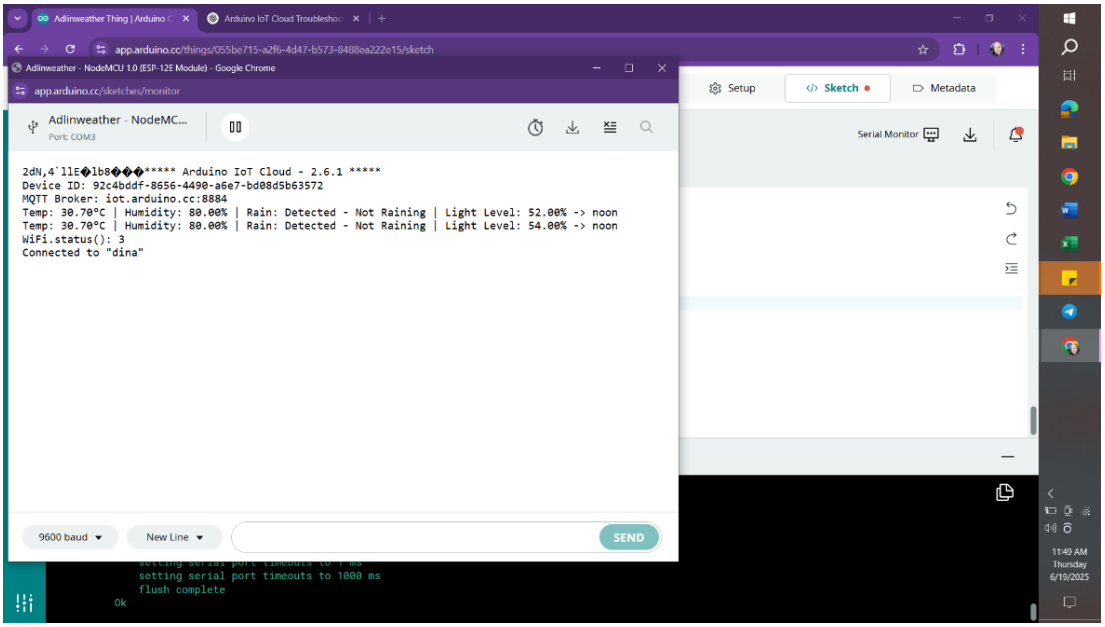

Figure 5.8: Shows the Serial Monitor in Arduino IoT Cloud of the Proposed System (Adlinweather)

For figure 5.6 and 5.7 is a seamless interface for monitoring real-time environmental data through both web and mobile dashboards. On the web and mobile dashboards, key sensor values such as temperature and humidity are displayed using gauge widgets for easy visualization. The Rain Detector and LDR Detector are represented with status indicators that change colour based on sensor readings. For the rain sensor, the indicator turns green when rain is detected and red when no rain is present. In contrast, the LDR sensor indicator turns red when it's bright (noon) and green when it's dark (night), based on light intensity thresholds. This colour-coded approach provides immediate visual feedback for the user. Additionally, the Serial Monitor displays detailed text-based outputs based on figure 5.8 that allowing the user to verify the precise sensor readings and system logic. Together, these tools demonstrate the system's real-time responsiveness and its ability to communicate weather conditions clearly and accurately across multiple platforms.

5.3 Non-Functional Testing

Non-functional testing was conducted to assess the overall performance and usability of the Smart Agriculture Weather Station under standard operating conditions. Unlike functional testing, which checks whether each component works as expected but non-functional testing focuses on how well the system performs. After testing the system, review and improvements to the system are made.

This testing is to validate the readiness of the system. In this project, only usability testing is used to measure the non-functional requirements of the system.

5.3.1 Usability Testing

Usability testing is a type of testing focused on evaluating how easy, user-friendly and accessible the system is for general users and for farmers that really need it for their farm. This focus on the interface especially the Arduino IoT Cloud dashboard was easy to use without technical knowledge.

The dashboard was tested on both desktop and mobile platforms and sensor data such as temperature, humidity, rain status, and light intensity were displayed through clearly labelled widgets. Rain and light detection were clearly represented with different colour signals that make it easy to understand system output. Users could recognize sensor status instantly through visuals without analyzing raw data.

The interface design was maintained clarity and usability across different screen sizes on both desktop and mobile views. Real-time updates were displayed without delay and ensuring the clear display and logically arranged. The Serial Monitor also supported usability by providing continuous feedback on system behaviour.

Overall, the system demonstrated good usability, with clear visuals, responsive updates, and accessibility across devices. These findings support the system's potential for deployment in real-world agricultural scenarios where simplicity and clarity are essential.

5.4 Summary

This chapter presented the testing procedures used to validate the Smart Agriculture Weather Station. Functional testing confirmed the accuracy and responsiveness of each sensor, while system testing verified the correct integration and data flow between hardware and cloud services. Non-functional testing ensured that the system performs reliably over time and across different platforms. Together, these tests confirmed that the system is both functional and reliable, making it suitable for real-world agricultural applications.

Chapter 6: Conclusion and Future Works

6.1 Introduction

This chapter concludes the Smart Agriculture Weather Station project by summarizing the work completed, the objectives achieved, and the knowledge gained throughout the development process. It discusses the limitations faced during implementation and testing and provides suggestions for potential improvements and future expansions of the system. The chapter also reflects on the overall experience and outcomes derived from the project.

6.2 Objective Achievement

The first objective was to design and implement a weather station that continuously monitors key environmental factors such as temperature, humidity, light intensity and rainfall. This objective was accomplished through the integration of multiple sensors connected to the ESP8266 microcontroller. Each sensor was individually tested and verified to produce accurate readings under different environmental conditions. The system continuously collected and processed real-time data from these sensors which enable core of the weather station's monitoring capabilities.

The second objective was to able remote access to real-time environmental data using the Arduino IoT Cloud. This was achieved by integrating the ESP8266 with the Arduino IoT Cloud where user could access real-time sensor trough both web and mobile dashboards. Users could monitor environmental conditions in real time from anywhere. This feature supported better time management and meeting the objective of enhancing convenience and accessibility.

The final objective as to promote sustainable farming practices. The system availability of live data allowed for better decision-making and helping farmers reduce inefficiencies in resource usage. Even without automated controls, the weather station equips users with the data required to adopt sustainable approaches. The system contributes to achieving sustainability goals by encouraging efficient and environmentally sound farming.

6.3 Project Limitations

During the development and testing phases of this project, the system faced some issues that made reduced the overall performance and user interaction quality. A major problem encountered was unstable Wi-Fi connectivity, which led to delays in uploading sensor data to the Arduino IoT Cloud. This caused noticeable lags in displaying live weather data especially in locations with low

signal coverage. The project also faced complications when configuring the Arduino IDE and uploading code to the microcontroller. Upload failures were common and usually linked to incorrect board configuration or unstable library integration. These issue not only interrupted development speed but also showed a gap in user support for IoT tools.

6.4 Future Works

Several areas in which this system offers multiple opportunities for future development to improve its performance and usefulness. Future enhancements could include automation that uses real-time sensor data to handle tasks such as irrigation without human input. For example, if humidity or soil moisture falls below a set point, the system could automatically switch on a water pump. This upgrade would not only improve efficiency but also helps minimize resource waste. To further improvements should consider adding cloud storage with features like cloud-based data storage with historical tracking and predictive analytics. With these features, farmers could analyze trends over time, forecast future weather conditions and take proactive steps in crop management. Additionally, implementing solar panel power supply would make the system more independence and sustainability especially in rural areas with unstable electricity. Finally, enhancing the user interface on mobile apps with push notifications or offline access could improve usability and ensure farmers receive timely alerts in all conditions.

6.5 Conclusion

In conclusion, the development of the Smart Agriculture Weather Station Powered by IoT proven that a low-cost and user-friendly weather monitoring system can be effectively developed for small-scale organic farming. The system supports real-time environmental data collection and remote access which is vital for improving crop health and sustainable farming practices. The hands-on experience of integrating hardware and software components, managing sensor data and testing the solution has provided valuable knowledge in IoT, cloud computing and user-centric design. Although the project had limitations, it laid a solid foundation for future expansion into more advanced smart farming systems. This project not only contributes technically but also socially by offering a scalable solution that empowers farmers and promotes environmental responsibility.

References

(n.d.). Retrieved from Vikaspedia: https://vikaspedia.in/agriculture/ict-applications-in-agriculture/iot-in-agriculture#:~:text=IoT%20in%20agriculture%20uses%20robots,save%20both%20time%20and%20money

A. A. Khan, M. Z. (2024). *Development of a Weather Station Using IoT Platform Based Arduino Integrated With a Control System for Smart Agriculture Applications*. Retrieved from IEEEXPLORE: https://ieeexplore.ieee.org/document/10649832

Climate-smart agriculture. (2024). Retrieved from WORLD BANK GROUP: https://www.worldbank.org/en/topic/climate-smart-agriculture

Eko Murdyantoro, R. S. (2019). *Prototype weather station uses LoRa wireless connectivity infrastructure*. Retrieved from PURPOS-LED PUBLISHING: https://iopscience.iop.org/article/10.1088/1742-6596/1367/1/012089/pdf

How Rain Sensor Works and Interface it with Arduino. (2025). Retrieved from Last Minute ENGINEERS: https://lastminuteengineers.com/rain-sensor-arduino-tutorial/

Leeron Hoory, C. B. (15 10, 2024). *What Is Waterfall Methodology? Here's How It Can Help Your Project Management Strategy*. Retrieved from FORBES ADVISOR : https://www.forbes.com/advisor/business/what-is-waterfall-methodology/

Light Intensity – Intro to Botany Review. (1 August, 2024). Retrieved from Fiveable: https://library.fiveable.me/key-terms/introduction-botany/light-intensity

Lima, I. P. (n.d.). *Weather parameters*. Retrieved from IMPETUS: https://impetus.aau.at/wp-content/uploads/2023/02/Weather-parameters-instruction.pdf

Marsh, J. (20 December, 2023). *Lithium-ion vs. lead acid batteries: How do they compare?* Retrieved from energysage: https://www.energysage.com/energy-storage/types-of-batteries/lithium-ion-vs-lead-acid-batteries/#lithium-ion-vs-lead-acid-batteries-overview

Padma Balaji Leelavinodhan, F. A. (2020). *Energy-neutral weather stations for precision agriculture: challenges and approaches*. Retrieved from IEEE: https://ieeexplore.ieee.org/document/9277565/figures

Patankar, S. (20 December, 2021). *Is the Waterfall model still applicable today?* Retrieved from AJACKUS: https://www.ajackus.com/blog/is-the-waterfall-model-still-relevant-in-todays-time

Sensors, R. (17 August, 2023). *The Role of Temperature and Humidity Sensors in Weather Stations*. Retrieved from RIKA: https://www.rikasensor.com/the-role-of-temperature-and-humidity-sensors-in-weather-stations.html#:~:text=Modern%20weather%20stations%20employ%20capacitive,on%20humidity%2Dinduced%20conductivity%20alterations.