DESIGN AND DEVELOPMENT OF IOT-BASED SMART GARDEN IMPLEMENTING EDGE COMPUTING

A **Design Project**

Presented to the

Department of Computer Engineering
College of Engineering
University of the East
Manila, Philippines

In Partial Fulfillment of the Requirements for the Degree
Bachelor of Science in Computer Engineering

By:

Danica Mae V. Alpuerto
James Matthew F. Tan
Johannes Christopher O. Cortez
Omar P. Ibrahim

Adviser:
A/P Errol John M. Antonio, M.Eng

May 2023

ABSTRACT

The agricultural sector in the Philippines is facing a labor shortage, which has significant implications for employment and food security. Additionally, managing water supply has become a critical challenge in agricultural regions due to droughts and increasing competition from non-agricultural sectors. There is a need for urgent and ambitious action to build resilient agricultural and farming systems. This study aims to design and develop an IoT-based smart garden system implementing edge computing and embedding it into the hydroponics system that reduces human intervention in plant production. The system integrates agricultural sensors and actuators through the Internet of Things and edge computing, enabling near real-time monitoring of various parameters. The user-friendly web-based dashboard acts as a monitoring tool, offering garden keepers an intuitive interface to facilitate efficient management. The system underwent development by Prototyping Methodology, which involves hardware identification, installation, and setup, along with data acquisition, preprocessing, and testing. The system underwent evaluation of its accuracy, functionality, and reliability through user-driven testing. The results showed that the IoT-based smart garden system successfully facilitated the growth of lettuce plants and provided dependable monitoring capabilities. Leveraging IoT and edge computing technologies in the hydroponics system offers a modern approach to agriculture, addressing labor shortages and enhancing productivity in the agricultural sector.

Table of Contents

Title Page

Abstract

Approval Sheet i

Certificate of Originality ii

Certificate of Plagiarism Check iii

Acknowledgement iv

Dedication v

Chapter I: Background of the Design Project

A. Introduction 1

B. Statement of Objectives 4

C. Conceptual Framework 5

D. Scope and Delimitation 6

E. Significance of the Study 8

F. Operational Definition of Terms 9

Chapter II: Review of Related Literature and Studies

A. Edge Computing 15

B. Internet of Things in the Agricultural Sector 17

C. Web-Based Monitoring System 19

D. Modern-Day Agriculture with Agricultural Sensors 23

E. Smart Garden System 25

F. Growing a Lettuce 27

G. Parameters for Controlled Environment Agriculture 30

H. Synthesis 33

Chapter III: Design Project Methodology

A. Methodology 36

B. Respondents / Beneficiaries of the Design Project 38

C. System Architecture 38

D. System Block Diagram 41

E. System Flowchart 43

F. Hardware Requirements 52

G. Software Requirements 65

H. Data Gathering and Testing Procedures 66

I. Design Project Validator/s 67

J. Design Project Evaluation Instrument 68

K. Statistical Treatment of Data 69

Chapter IV: Presentation, Analysis, and Interpretation of Data

A. Results and Discussion 70

Chapter V: Conclusion and Recommendation

A. Summary of Findings, Conclusions, and Recommendations 81

References

Appendices

A. Curriculum Vitae of the Beneficiary 89

B. Survey Questionnaire 91

C. Expected Outputs 92

D. Actual Outputs 99

E. Sample Testing Images 106

F. Schematic Diagram 111

G. Sample Source Codes 115

H. Sensors Accuracy Tables 125

List of Figures

Figure 1: Conceptual Framework 5

Figure 2: Growing a Lettuce 27

Figure 3. Overview of the Design Project Methodology 34

Figure 4: System Architecture 38

Figure 5: System Block Diagram 39

Figure 6: Insert Data to Database (Microcontroller) 41

Figure 7: Send Data to Local Host Server MySQL using HTTP Post Method Request 42

Figure 8: Attempt Database Connection 43

Figure 9: Pull Data from Database 44

Figure 10: Control Actuators (Automation) 45

Figure 11: Insert Data to Database (Web Application) 46

Figure 12: Display Data to Web Application 47

Figure 13: System Automation 48

Figure 14: Control Actuator Using the Web Application 49

Figure 15: ESP32 Microcontroller 50

Figure 16: MQ135 Air Quality Sensor Module (CO2 Sensor) 51

Figure 17: DHT22 Temperature and Humidity Sensor 52

Figure 18: Gravity: Analog Ambient Light Sensor 52

Figure 19: Soil Moisture Sensor Corrosion Resistant 53

Figure 20: Float Sensor 54

Figure 21: Air Cooler 55

Figure 22: Grow Light 56

Figure 23: Submersible Pump 57

Figure 24: Water Pump 58

Figure 25: Automatic Transfer Switch 59
Figure 26: Backup Generator 60
Figure 27: Exhaust Fan 61
Figure 28: Relay Module 62
Figure 29: Uninterruptible Power Supply 63
Figure 29: Accuracy of the Water Level Sensor 70
Figure 30: Accuracy of the Actuator of the Carbon Dioxide Sensor 71
Figure 31: Accuracy of the Actuator of the Humidity Sensor 72
Figure 32: Accuracy of the Actuator of the Light Sensor 73
Figure 33: Accuracy of the Actuator of the Soil Moisture 74
Figure 34: Accuracy of the Actuator of the Temperature Sensor 75
Figure 35: Accuracy of the Actuator of the Water Level Sensor 76
Figure 36: Reliability of the Bypass System 77
Figure 37: System Model (Front View) 89
Figure 38: System Model (Oblique View) 89
Figure 39: Sensor Casing (Top View) 90
Figure 40: Sensor Casing (Side View) 90
Figure 41: Log-In Page 92
Figure 42: Meter Page 92
Figure 43: Graph Page 93
Figure 44: Relevant Data Page 93
Figure 45: Bypass Actuators Page 94
Figure 46: Admins Page 94
Figure 47: Activity Logs Page 95
Figure 48: Account Settings Page 95
Figure No. 49. Prototype View 1 97

Figure No. 50. Prototype View 2	97
Figure No. 51. Prototype View 3	98
Figure No. 52. Prototype View 4	99
Figure No. 53. Log In Page	100
Figure No. 54. Meter Page	100
Figure No. 54. Meter Page	101
Figure No. 55. Bypass Actuators Page	101
Figure No. 56. Manage Admins Page	102
Figure No. 57. Activity Logs Page	102
Figure No. 58. Account Settings Page	103
Figure No. 59. Testing 1 for the Accuracy of the Soil Moisture Sensor	104
Figure No. 60. Testing 2 for the Accuracy of the Soil Moisture Sensor	104
Figure No. 61. Testing 1 for the Accuracy of the Light Sensor	105
Figure No. 62. Testing 2 for the Accuracy of the Light Sensor	105
Figure No. 63. Testing 1 for the Accuracy of the Humidty Sensor	106
Figure No. 63. Testing 2 for the Accuracy of the Humidty Sensor	106
Figure No. 64. Testing 1 for the Accuracy of the Carbon Dioxide Sensor	107
Figure No. 65. Testing 2 for the Accuracy of the Carbon Dioxide Sensor	107
Figure No. 66. Testing 1 for the Accuracy of the Temperature Sensor	108
Figure No. 67. Testing 2 for the Accuracy of the Carbon Dioxide	108
Figure No. 68. Schematic Diagram for the ESP32 (Carbon Dioxide, Humidity, Light, Temperature Sensor)	109
Figure No. 69. Schematic Diagram for the ESP32 (Soil Moisture Sensor)	110
Figure No. 70. Schematic Diagram for the ESP32 (Water Level Sensor)	111
Figure No. 71. Schematic Diagram for the ESP32 (Actuators)	112
Figure No. 72. Sending the Humidity, Temperature, Light, Carbon Dioxide Sensor Data to the MySQL Database	113

Figure No. 73. Sending the Soil Moisture Sensor Data to the MySQL Database 114

Figure No. 74. Sending the Water Level Sensor Data to the MySQL Database 115

Figure No. 75. Receiving the Light Sensor Data from the MySQL Database to Automate and Control the Grow Light 116

Figure No. 76. Receiving the Temperature Sensor Data from the MySQL Database to Automate and Control the Fan 117

Figure No. 77. Receiving the Humidity Sensor Data from the MySQL Database to Automate and Control the Sprinkler 118

Figure No. 78. Sending the Data from MySQL Database to the ESP32 to Automate and Control the Actuators 119

Figure No. 79. Insert the Sensor Data coming from the ESP32 Microcontroller into the MySQL Database 120

Figure No. 80. Display the Sensor Data to the Meter Page 121

Figure No. 81. Pulling the Data from the MySQL Database to Display in Meter Page 122

List of Tables

Table 1: Standard Environmental Conditions of Lettuce 30

Table 2: Likert Scale 67

Table 3: Average Accuracy of the 6 Sensors of the System 68

Table 4: Accuracy of the Carbon Dioxide Sensor 123

Table 5: Accuracy of the Humidity Sensor 124

Table 6: Accuracy of the Temperature Sensor 125

Table 7: Accuracy of the Light Sensor 127

Table 8: Accuracy of the Soil Moisture 129

UNIVERSITY OF THE EAST
Manila
COLLEGE OF ENGINEERING
DEPARTMENT OF COMPUTER ENGINEERING

APPROVAL SHEET

The ***undergraduate design project***

DESIGN AND DEVELOPMENT OF IOT-BASED SMART GARDEN IMPLEMENTING EDGE COMPUTING

which was presented before a Panel of Examiners of Department of Computer Engineering, College of Engineering, University of the East - Manila Campus, Philippines, on May 22, 2023 by

Danica Mae V. Alpuerto
James Matthew F. Tan
Johannes Christopher O. Cortez
Omar P. Ibrahim

is hereby **APPROVED** and **ACCEPTED** in partial fulfillment of the degree **Bachelor of Science in Computer Engineering (BS CpE)** with the overall rating of **EXCELLENT**.

Engr. ONOFRE O. CORPUZ , MES
Panel Chairman

A/P MARY ANN L. LIMKIAN, M.Eng
Panel Member

Mr. KEARLLAN M. DEL CASTILLO
Panel Member

A/P ERROL JOHN M. ANTONIO, M.Eng
Design Project Coordinator

Engr. DARWIN D. ALPIS
Associate Dean, OIC

UNIVERSITY OF THE EAST
Manila
COLLEGE OF ENGINEERING
DEPARTMENT OF COMPUTER ENGINEERING

CERTIFICATE OF ORIGINALITY

We, the undersigned design project proponents hereby declare that this manuscript is our own work and that, to the best of our knowledge and belief, it contains no material previously published or written by another person or organization, except where the due acknowledgement is made in the text.

We have duly acknowledged all the sources from which the ideas and extracts have been taken. The manuscript is free from any form of plagiarism act and has not been submitted elsewhere for any publication.

Design Project Title:

DESIGN AND DEVELOPMENT OF IOT-BASED SMART GARDEN IMPLEMENTING EDGE COMPUTING

Danica Mae V. Alpuerto
Author

James Matthew F. Tan
Author

Johannes Christopher O. Cortez
Author

Omar P. Ibrahim
Author

Attested by:

A/P Errol John M. Antonio, M.Eng
Faculty Adviser & Co-Author

UNIVERSITY OF THE EAST
Manila
COLLEGE OF ENGINEERING
DEPARTMENT OF COMPUTER ENGINEERING

CERTIFICATE OF PLAGIARISM CHECK

This is to certify that the manuscript of the design project entitled

DESIGN AND DEVELOPMENT OF IOT-BASED SMART GARDEN IMPLEMENTING EDGE COMPUTING

by

Danica Mae V. Alpuerto

James Matthew F. Tan

Johannes Christopher O. Cortez

Omar P. Ibrahim

A/P Errol John M. Antonio, M.Eng

has been subjected to similarity check by turnitin plagiarism software.

A/P Errol John M. Antonio, M.Eng
Design Project Coordinator

ACKNOWLEDGEMENTS

The proponents would like to express heartfelt gratitude to the following individuals and entities who have contributed to this Design Project Paper. To God, for His abundant blessings, unwavering guidance, and boundless grace; We are grateful for His presence in our life, which has provided us the strength, wisdom, and inspiration to undertake this endeavor and overcome challenges; To our Family, whose love, support, and understanding have been our pillars of strength. Their unwavering belief in our abilities and constant encouragement have motivated us to push beyond our limits. We are grateful for their sacrifices, understanding, and the countless ways they have nurtured our growth. To our Beneficiary, Mr. Marvin C. Baeza, whose collaboration and involvement have been invaluable throughout this project. Your insights, needs, and feedback have shaped the direction of this work. We extend our appreciation for your trust, patience, and willingness to participate, as this project addressed your requirements and made a meaningful impact. To our Faculty Adviser, A/P Errol John M. Antonio, M.Eng, whose expertise, guidance, and commitment have been instrumental in the success of this endeavor. His mentorship, constructive criticism, and valuable suggestions have helped us refine our ideas, improve our design project, and elevate the quality of this design project paper. We are deeply grateful for your dedication and support. To the three (3) Panels, Engr. Onofre O Corpuz, MES, A/P. Mary Ann L. Limkian, M.Eng, and Mr. Kearllan Aeymard M. Del Castillo, we extend our sincere appreciation for their time, rigorous assessment, and constructive feedback. Their thorough review has played a crucial role in enhancing the credibility and academic rigor of this work. We would also like to acknowledge the support and contributions of friends, classmates, and colleagues who have provided assistance, encouragement, and valuable discussions throughout this journey. Your presence and engagement have enriched Our experience and shaped the outcome of this design project. Lastly, We express Our gratitude to our academic institution, the University of East Manila, and all the resources it has provided, enabling us to undertake this design project and pursue our Academic aspirations. The completion of this Design Project Paper would not have been possible without the collective support and contributions of the aforementioned individuals and entities. Thank you all for your unwavering belief in us and for being an integral part of this significant achievement.

DEDICATION

This design project is dedicated to: To God, for His divine guidance, unwavering love, and boundless grace. His presence has been a constant source of inspiration and strength throughout this journey. We are deeply grateful for His blessings and for the opportunity to utilize our skills and knowledge for the betterment of others. To the Philippines and to our fellow Countrymen, a nation filled with resilience, spirit, and cultural richness. This dedication is a testament to the love we have for our homeland and to our fellow Countrymen, who have shaped our identity, nurtured our growth, and instilled in us the values of hard work, determination, and compassion. May this work contribute to the betterment of our beloved country and its citizens. To the Hydroponics Organization, Whose dedication to sustainable farming practices and innovative solutions has been a driving force behind this design project. Your commitment to promoting efficient agricultural methods has inspired us to explore the potential of hydroponics systems and their impact on food security and environmental sustainability. Lastly, to the Lettuce Farmers, Whose unwavering passion, hard work, and expertise in cultivating lettuce have been a source of inspiration. Your dedication to producing high-quality, nutritious produce has motivated us to develop, innovate, and refine hydroponic systems that can enhance crop yields, optimize resource utilization, and contribute to the well-being of farmers and consumers alike. This work is dedicated to God, the Philippines, and to our fellow Countrymen, the Hydroponics Organization, and the Lettuce Farmers. May this design project serve as a testament to our collective efforts and a catalyst for further advancements in sustainable agriculture, benefiting communities, and promoting a greener future.

Chapter Ⅰ

Background of the Design Project

Ⅰ. Introduction

The labor shortage is the most pressing issue in the agricultural sector in the Philippines. The total number of people working in agriculture decreased during the five-year reference period, and in 2019 the agricultural sector employed roughly 9.70 million people. As a result, the nation's total employment from the agriculture sector fell to 22.9%. Western Visayas continues to have the most significant employment in agriculture at the regional level. However, in 2019 it dropped to 873,000 people. In the majority of regions, there was a decline in agricultural employment. On the other hand, Central Visayas, Northern Mindanao, SOCCSKSARGEN, and BARMM all had increases in the number of people working in the agriculture sector, with 657,000, 776,000, 800,000, and 697,000 workers, respectively. In contrast, the minor count in the NCR persisted in 2019, with 25,000 people (Mapa et al., 2020). Moreover, managing water supply is a problem the agricultural sector faces. In recent years, agricultural regions have experienced severe and escalating water restrictions. Significant droughts in Chile and the US have reduced surface and groundwater supplies while affecting agricultural production. In addition to these adjustments, the growing urban population density and the water requirements of the energy and industrial sectors will put farmers in many places at greater risk of competition from non-agricultural users. Additionally, the expansion of polluting industries, salination brought on by rising sea levels, and the aforementioned changes in the water supply are all likely to cause water quality to worsen in many locations. The productivity of rain-fed and irrigated crops and livestock operations, particularly in some nations and regions, are projected to be seriously hampered by these water difficulties, which are expected to impact agriculture significantly. The agricultural sector depends heavily on water. These modifications may significantly affect markets, commerce, and overall food security. Without additional action, Northeast China, Northwest India, and the Southwest of the United States are expected to be among the regions most severely impacted, with consequences for both the local economy and the entire world, according to an Organisation for Economic Co-operation and Development (OECD) assessment of future water risk hotspots (Water and Agriculture - OECD, n.d.).

Despite the decline in the labor shortage, the agricultural sector still provides a living for over 2.5 billion people worldwide. Given the agricultural sector's inherent interactions with the environment, direct reliance on natural resources for production, and the importance of national-socio economic development, urgent and ambitious action to build more resilient agricultural systems is considered necessary. The global population is rapidly increasing, creating a need for safe and secure food to feed this growing population. As a result, traditional farming methods are inadequate to meet this demand (Abd El-Kader & El-Basioni, 2020). In the Philippines, the agriculture sector started as the backbone of the economy before the emergence of the service and industry sector and is now underdeveloped and the least desired by potential Filipino employees compared to the service sector, which is currently the dominant economic sector (2018 Census of Philippine Business and Industry: Economy-Wide | Philippine Statistics Authority, n.d.).

Exploring technologies in the agricultural sector, like the Internet of Things (IoT), will promote modernizing agriculture and elevate the sector's economic potential (Kar, n.d.). Internet of Things (IoT) refers to physical things with sensors, computing power, software, and other technologies that can link to other systems and devices via the Internet or other communication networks and exchange data [No Reference]. With the rapid expansion of the Internet of Everything (IoE), more smart devices are becoming online and producing significant amounts of data. This has led to issues with traditional cloud computing models like bandwidth strain, sluggish response times, inadequate security, and poor privacy. Edge computing solutions have emerged since traditional cloud computing can no longer serve today's intelligent society's diversified data processing needs. It is a fresh paradigm for doing calculations at the network's edge. It stresses being closer to the user and the data source than cloud computing. It is portable for local, small-scale data processing and storage at the network's edge. One way to connect physical devices with sensors to other systems is by using a microcontroller such as an Arduino microcontroller. The open-source electronics platform Arduino is built on simple hardware and software. Boards like Arduino can accept inputs like light from a sensor, a touch on a button, or a tweet and translates them into outputs like turning on an LED, starting a motor, or posting something online. Arduino in IoT applications can collect data from a sensor and send the data back to the system or receive data to control actuators. Such technology will benefit the agricultural sector and help mitigate existing problems for years (Cao et al., 2020).

The proponents chose this study to help the agricultural sector combat existing issues/problems. The proponents also believed the study would help farmers adapt to new technology and help manage their farms with a modern approach to producing plant products. The proponents tackled several research problems: How would farmers adapt to the system? How could the system aid farm labor shortage? How would the system manage the water supply? How would the system produce plant products with fewer human interventions? These questions were the reason why the study needed to be studied. This kind of study already existed; however, the proponents tried to innovate and design an almost independent system that farmers could simply adapt, manage, and learn.

Ⅱ. Statement of the Objectives

The study aimed to design and develop an IoT-Based Smart Garden System that raised economical and practical ways of aiding in the production of plants with less human intervention. Specifically, the study aimed to:

- design and develop a multi-sensor system integrated with agricultural sensors and actuators via the internet of things and edge computing;
- develop a user-friendly web-based dashboard to serve as the monitoring tool for the garden keepers;
- Use user-driven testing to evaluate the developed system's accuracy, functionality, and reliability.

Ⅲ. Conceptual Framework

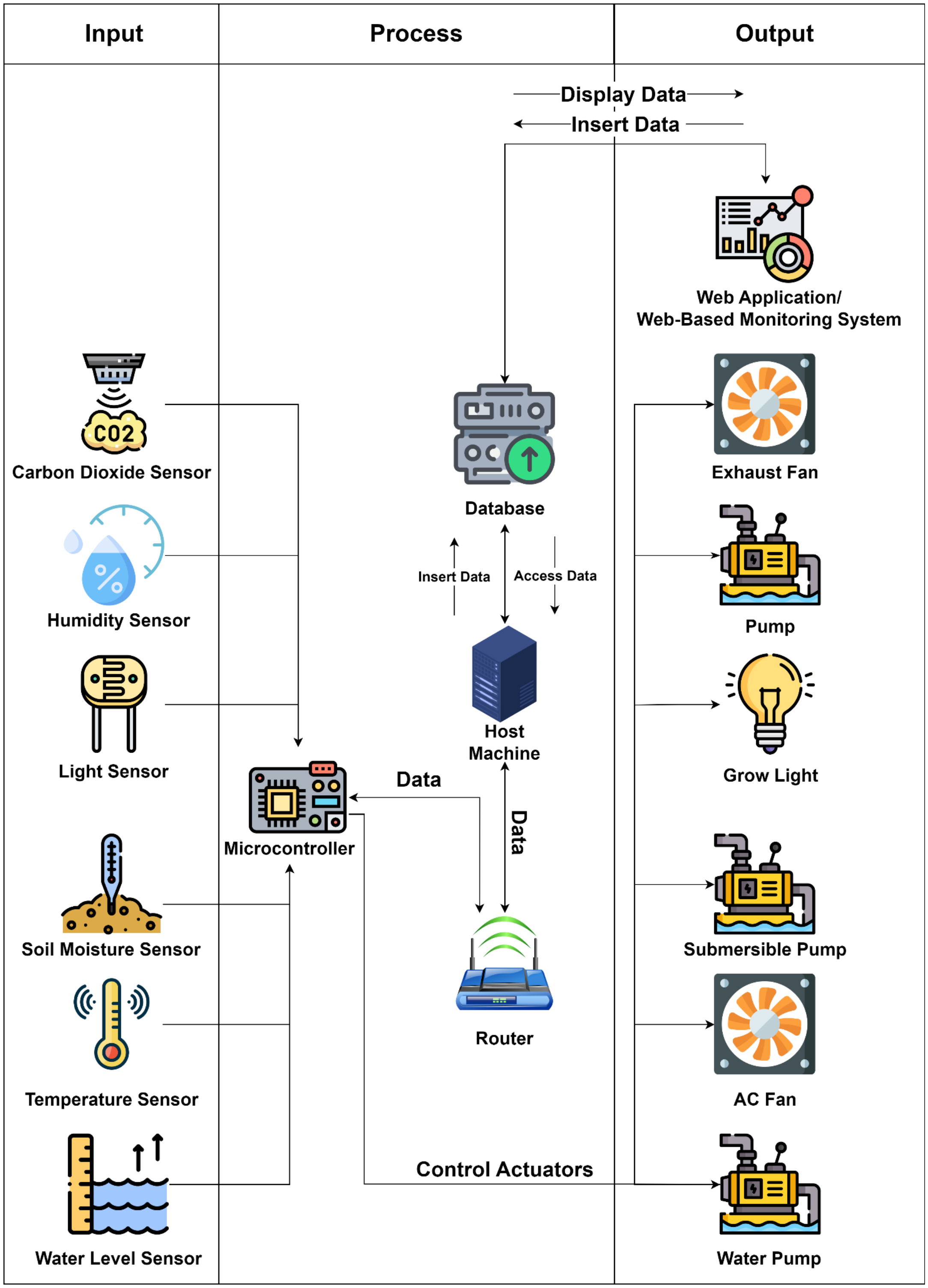

Figure 1: Conceptual Framework.

In figure 1, the carbon dioxide sensor, water level sensor, light sensor, soil moisture sensor, temperature sensor, and water level sensor served as the input sensors of the system connected to the microcontroller/ESP32. The ESP32 acted as a server and a client to send data to the host machine and receive data from the host machine. Using the WiFi/router, the ESP32 server, host machine, database, and web application could exchange data with one another. [1] When the host machine successfully receives the data from the ESP32 server, it inserts it into the database, and the web application can access and display it in near real-time. [2] When the host machine received the data from the database from the web application, it passed it to the ESP32 server and could now collect the data to control the actuators. AC fan, exhaust fan, grow light, submersible pump, and another two water pumps are the actuators/outputs of the system. The actuators are either automated or controlled via the web application. Therefore, the web application or the web-based monitoring system served as both the input and output of the system.

Ⅳ. Scope and Delimitations

The research is focused on developing and designing a system for the benefit of the agricultural sector, specifically for farmers in Miguela's Farm producing various lettuce. It does not mean the proponent's system will not be suitable for other farmers producing a different product. It will still be suitable, but with modifications, as different plants have distinct characteristics. In short, the system will be designed and developed that is fit for producing lettuce, specifically the Batavia Lettuce (Scientific Name: Lactuca sativa var. longifolia). Miguela's Farm provided the proponents with a greenhouse there; the proponents implemented the system in a span of 4 months. The system's functions are the following:

- the system detects, automates, and controls the carbon dioxide near the surrounding area of the lettuce plants. If the carbon dioxide goes beyond the given standard value, the system will filter the carbon dioxide;
- the system detects, automates, and controls the soil moisture of the lettuce plant if the plant's soil moisture is below the required amount. The system will moisten the soil with water from the water tank;
- the system detects the water level in the water tank and automate and control the water supply use; If the water in the tank is below the average level, the system will replenish the tank with water until it is filled;
- the system detects, automates, and controls the plants surrounding temperature; if the climate is above the minimum temperature for the lettuce plants to survive, the system will adjust the temperature of the area where the plant is located;
- the system detects and mimics sunlight. Sometimes sunlight is unavailable, and sunlight plays a vital role in plants' survival. If the system detects there is not enough sunlight, the system will provide the lettuce plants with artificial sunlight;
- the system detects, automates, and controls the humidity in the surrounding area of the lettuce plant; if the air has insufficient water vapor, the system will mist water in the vicinity;
- the system senses the air's carbon dioxide content;
- data or relevant data coming from the sensors are collected and stored in the database every 35 seconds;

- on top of this, the system has a web-based monitoring system that monitors the sensor's data in near real-time accessed in the database and can also send data to the server of the microcontroller to control the actuators like grow lights and motors.

Furthermore, the limits of the system are the following:

- it cannot predict or detect weather; the user cannot modify the system's front-end and back-end;
- it cannot protect the harvest from harsh calamities; it cannot protect the harvest from insects/pests;
- the hardware involved in this design project is budget-friendly due to the budget constraints of the proponents.
- the system is not fully automated and still need human intervention;
- the system only implemented edge computing without cloud computing;
- lastly, the user are not able to add plant sections, new sensors, or modify the original layout of the system.

V. Significance of the Study

- **Miguela's Farm/Lettuce Farmers** - The system is based on the environments of Miguela's farm, and the implementation of the system is also on Miguela's farm. After building the system, they may use the system to produce their products. Miguela's farm produces lettuce plants. Technically, lettuce farmers benefited firsthand from the proponents' study. The design project also helped farmers adapt to new technology and help manage their farms with a modern approach to producing plant products.

- **Hydroponics Organization** - The proponents' chosen benefactor is part of the hydroponics organization; the organization has access to the design project.
- **Department of Agriculture** - The design project focuses on lettuce, but it is possible that it can be integrated with various plants. The agricultural sector or the farmers may acquire the system and the proponent's service to implement it on their farms.
- **Community** - The system may contribute to philippine society and the economy by placing it on the market. It may also be bought by people interested in producing plant products, not just farmers.

VI. Operational Definition of Terms

- **AC Fan**

  An AC fan, or air conditioner fan, is used to circulate the air in a room or building and help cool it down. AC fans are commonly found in air conditioning units, which use refrigerant to cool the air and remove moisture to create a comfortable living environment. AC fan serves as an actuator for the temperature sensor.

- **Actuators**

  An actuator is a device used to control or move a mechanism or system. Actuators are commonly found in various mechanical, electrical, and hydraulic applications. Instead of farmers in the field, the actuator is responsible for growing the lettuce based on the given condition of the system.

- **Carbon Dioxide Sensor**

A carbon dioxide (CO_2) sensor is a device used to sense the concentration of carbon dioxide in the air. Carbon dioxide sensors are commonly used in various applications, including indoor air quality monitoring, greenhouse gas emissions monitoring, and industrial process control.

- **Edge Computing**

  Edge computing refers to processing data and carrying out computation at or near the point where the data is generated rather than in a centralized data centre or cloud. The goal of edge computing is to reduce latency and improve data processing efficiency by bringing computation closer to the data source. Edge computing is often used in applications where the data generated is too large or time-sensitive to be transmitted to a centralized location for processing. For example, edge computing is commonly used in the Internet of Things (IoT), where devices such as sensors and cameras generate large amounts of data that needs to be processed and analyzed in real time.

- **Exhaust Fan**

  An exhaust fan is a mechanical ventilation device that removes stale or polluted air from a room or building and expels it outside. Exhaust fans are commonly used in homes, offices, and other buildings to improve indoor air quality, reduce odours, and reduce moisture and heat buildup. Exhaust fans are typically installed in the ceiling or a wall and can be controlled manually or with a timer. Some exhaust fans also have built-in filters to remove dust, pollen, and other contaminants from the air before it is expelled outside. Exhaust fan serves as an actuator for the carbon dioxide sensor.

- **Grow Light**

  A grow light is a type of artificial lighting system used to provide light to plants during photosynthesis, which is necessary for their growth and development. Grow lights are designed to provide the wavelengths of light that plants need to photosynthesize. Grow light serves as an actuator for the light sensor.

- **Humidity Sensor**

  A humidity sensor is a device used to sense the relative humidity of the air. Humidity sensors measure the amount of water vapour in the air and express it as a percentage of the maximum amount of water vapour that the air can hold at a given temperature. Humidity sensors are commonly used in various applications, including climate control systems, industrial process control, and weather monitoring.

- **Internet of Things (IoT)**

  The internet of things (IoT) refers to the interconnected network of physical objects, devices, and sensors equipped with computing capabilities that can communicate with each other and external systems over the internet. The proponents' system is an internet of things (IoT) system. It will have 5 layers of internet of things (IoT) architecture: perception layer, edge layer, network layer, data layer, and application layer.

- **Light sensor,**

  A light sensor is a device used to sense light intensity and spectrum. Light sensors are commonly used in various applications, including photography, lighting control, and environmental monitoring.

- **Microcontroller**

  The ESP32 is a microcontroller that espressif systems develop. It is a low-cost, low-power system-on-chip (SoC) designed for the Internet of Things (IoT) and other connected applications. The ESP32 Microcontroller will read data from the sensor or input data based on conditions. Since the microcontroller has a built-in wifi module, it can connect to the network wirelessly.

- **Relevant Data**

  Relevant data describes the information that is important or beneficial for a specific goal or decision-making procedure. It is information pertinent to the issue at hand and from which it is possible to deduce important truths or insights.

- **Sensors**

  A sensor is a device that senses a physical property or phenomenon and converts it into a form that a system or device can interpret. Sensors are used in various applications, including scientific, industrial, and consumer products. The proponents used sensors capable of helping farmers monitor changes in the environment surrounding the lettuce plant and, at the same time, optimize the growth by controlling the actuators based on the given value of the sensors and the given condition.

- **Soil Moisture Sensor**

  A soil moisture sensor is a device that senses soil's water content. Soil moisture sensors are commonly used in agriculture, horticulture, and landscaping to monitor soil moisture levels and optimize irrigation.

- **Submersible Pump**

  A submersible pump is a type of pump that is designed to be fully immersed in a liquid. Submersible pumps are commonly used for various applications, including water well drilling, sewage treatment, and industrial process control. Submersible pumps are used when the liquid being pumped is too deep to be reached with a surface pump or when the pump needs to be located below the liquid level. Submersible pump serves as an actuator for the soil moisture sensors.

- **Temperature Sensor**

  A temperature sensor is a device that is used to sense temperature. Temperature sensors are commonly used in various applications, including climate control, industrial process control, and medical monitoring.

- **Water Level Sensor**

  A water level sensor is a device that senses the water level in a tank, reservoir, or other container. Water level sensors are commonly used in various applications, including irrigation, water treatment, and industrial process control.

- **Water Pump**

  A water pump is a device that moves water from one location to another. Water pumps are commonly used in various applications, including irrigation, water treatment, and industrial process control. Water pump serves as an actuator for the humidity and water level sensors.

- **Web-Based Monitoring System**

  An internet of things (IoT) web monitoring system is a tool used to monitor the performance and functionality of sensors connected to the Internet of Things (IoT). IoT Web monitoring systems are used to ensure that the sensors connected to the IoT are functioning correctly and providing accurate data. The system has a web-based monitoring system that can be accessed on any device since it can be accessed from the web locally. All the data coming from the sensors will be displayed in the said monitoring system. For the development of the monitoring system, the proponents used C++, CSS, JavaScript, HTML, and PHP for programming.

Chapter Ⅱ

Review of Related Literature and Studies

Ⅰ. Edge Computing

Various networks and devices at or close to the user are called edge computing, an nearemerging computing paradigm. Edge is about processing data faster and in a larger volume near the point of generation, providing near real-time action-driven solutions (What Is Edge Computing & Why Is It Important? | Accenture, n.d.). Edge computing offers further advantages of agility, real-time processing, and autonomy in the area of Industrial IoT to provide value for intelligent manufacturing (Chen et al., 2018).

Edge computing has several advantages (Edge Computing Paradigm | OpenMind, n.d.):

- **Compliance:** Some countries have data protection rules that require data to be retained and processed in the country where it was generated. Edge computing can help firms comply with these rules more easily.
- **Cost savings:** Less data transfer can also result in cheaper data transmission costs. It can be substantial, especially in an area where the price of mobile data is high.
- **Improved Monitoring:** Edge computing also enables organizations to keep track of a broad range of data that they would not be able to track otherwise. It is crucial for smart manufacturing, smart agriculture, smart grids, smart gardens, and IoT use cases.

- **Increased Dependability:** A core aspect of disaster recovery/business continuity (DR/BC) best practices is distributing your data across numerous physical locations. If the workloads are processed by a large number of smaller devices, the failure of a single unit is less likely to be catastrophic.
- **Increased Security:** Attackers have a large and appealing target if data is stored and processed in one area. However, edge computing makes it less likely for attackers to obtain a massive data treasure. Furthermore, edge computing makes distributed denial of service (DDOS) assaults more challenging to execute.
- **Reduced network traffic:** Today's devices generate so much data that networks often struggle to keep up. More processing at the edge decreases network bandwidth requirements, releasing capacity for the most critical applications.
- **Speed:** Users do not have to wait for data to get up to the cloud and back down if it is analyzed close to where it is created. This decrease in latency leads to faster performance.

The agricultural IoT generates significant data; edge computing overcomes data processing difficulties. Edge nodes must have the following to generate data: Strategies for storing data determine whether it is structured, unstructured, or semi-structured. The IT crew should know what kind and how much data the edge node will temporarily or permanently contain on the farm. In terms of storing data, The data obtained are sometimes kept for an extended time. Nevertheless, occasionally, only a portion of the data needs to be kept or kept in storage for a brief time. Due to the relative emergence of Edge computing, only a few scholars have applied this paradigm computing in agriculture. Edge computing research is abundant in the realm of smart cities and smart homes; it is worth noting that Edge computing will only spread further into the agriculture field, and it is estimated that 50% of the Internet of Things with more than 50 billion terminals will face network bandwidth limitations, and 40% of data will need to be analyzed, processed and cached at the edge of the network (Zhang et al., n.d.).

Ⅱ. Internet of Things in the Agriculture Sector

The network of physical devices, or "things," implanted with sensors, software, and other technologies to communicate and exchange data with other devices and systems through the internet is called the internet of things (IoT). These gadgets include anything from everyday domestic items to high-tech industrial gear. Today, there are more than 7 billion connected IoT devices, and according to analysts, there will be 10 billion by 2020 and 22 billion by 2025 (What Is the Internet of Things (IoT)? | Oracle Philippines, n.d.). The internet of things (IoT) is one of the most significant areas of future technologies. It is rapidly gaining popularity in various fields and applications related to smart cities, the military, education, hospitals, homeland security systems, transportation, agriculture, intelligent shopping systems, and other contemporary technologies (Internet of Things (IoT) for Automated and Smart Applications - Google Books, n.d.). The internet of things (IoT) is an innovative technology that offers practical and dependable answers for the modernization of numerous fields. Solutions built on the internet of things (IoT) are being created to autonomously maintain and monitor agricultural farms with the least amount of human participation. Much effort has been made to utilize IoT in agriculture to create intelligent farming solutions. Agriculture has undergone a significant shift. With the development of technology today, it is anticipated that agriculturalists and technologists will use IoT to address issues that farmers are now facing, such as water shortages. Challenges with cost control and production. Modern IoT technologies have identified these problems and offer fixes that will raise productivity while decreasing costs. Working on wireless sensor networks has allowed people to gather sensor data and communicate it to the primary servers. Sensor data provides information on various environmental conditions, enabling accurate system monitoring. Crop yield and climatic conditions are essential factors for crop evaluation. However, many others affect the production of the crops, such as field management, soil and crop monitoring, the movement of an undesired object, wild animal attacks, thefts, and others. Additionally, IoT offers a well-organized scheduling system for constrained resources, ensuring that the optimum use of IoT increases productivity (Farooq et al., 2019). There are three main layers of Conventional IoT Architecture: perception, network, and application. The perception layer is another name for the recognition layer; it is the lowest layer of the traditional internet of things (IoT) architecture. The primary duty of this layer is to gather and convert helpful information from the sensors or the environment into an electrical signal. The network layer comes after the perception layer and is regarded as the brain. The main objective of this layer is to aid in the secure transfer of data from the perception layer to the application layer. The application layer is the top layer of the internet of things (IoT) architecture. The application layer offers personalized service based on the user's requirements. The application layer's primary duty is to bridge the considerable distance between users and applications. The application layer integrates the business to provide high-level intelligent applications such as disaster monitoring, health monitoring, transposition, medical, and ecological environments and manages global management for all intelligent applications (Towards the Internet of Things: Architectures, Security, and Applications - Mohammad Ali Jabraeil Jamali, Bahareh Bahrami, Arash Heidari, Parisa Allahverdizadeh, Farhad Norouzi - Google Books, n.d.). By 2050, there should be a global growth in agriculture of more than 50% to feed an additional 2 billion people. This may help with issues and obstacles like inaccurate demand forecasting, inadequate assured irrigation, and needless misuse of chemicals and fertilizers while boosting production. It can enhance the quality of the harvest, address labor issues, forecast weather information, and other global problems (AI, Edge and IoT-Based Smart Agriculture - Google Books, n.d.).

Ⅲ. Web-Based Monitoring System

Implementing a monitoring system is an essential element in any system or device; it serves as the visual gateway for the user to keep a keen observation from a distance. A monitoring system can also keep the user from any potential harm and, most importantly, gives the user the convenience of attending to any other pressing matter. The system can come in many forms with its advantages and disadvantages; for the system, a Web-based monitoring system is implemented due to the relatively friendly user interface and requirement of internet access. In 2017, a pair of engineering students from Turkey created a web-based system to monitor homework called TPS: teacher parent's student system, which aims to ease the workload for teachers, provide convenience for the students, and give the parents the status of their children's performance. The monitoring system was developed using HTML, CSS, and JavaScript as the framework for the web pages. MySQL is the database that stores data, and PHP is the processing programming language. The TPS system is an innovation due to the lack of features such as a Web-based Operating System developed by Malaysia in 2014. Still, it only applies to the physics-based secondary-level operating system. The TPS boasts a wide range of features suited for all kinds of homework and can be adapted to any school and the parents' performance monitoring (Thesis & Alsaadi, 2018). A study conducted in 2019 investigated the student's perception of the effectiveness of the web-based homework system in mathematics. The investigation also included the features found in the web-based system, which include immediate feedback, multiple attempts, and others. It was concluded that the web-based system was seen positively by the students and motivated them to practice further (Serhan, 2019). The serviceability of a web-based monitoring system can also be adapted in the healthcare sector or any other sector. A Web-based monitoring system can accurately measure data, improve performance, increase response time, explicit and informed decision-making, and others. A well-developed monitoring system will be helpful to stakeholders, including project managers, project members, decision-makers, policymakers, and M&E officers, in tracking the projects' progress in the health domain as a tool for better and more informed decisions (Mleke & Dida, 2020). In 2020, engineering students from The Nelson Mandela African Institution of Science developed a Web-based monitoring and evaluation system to be deployed to the Ministry of Health in Tanzania. The framework was built around the use of Hypertext Preprocessor (PHP) for database connection and manipulation and carries website duties such as authentication, password handling, and forum managing. Although not as fast as compiling the code in C or a similar language, it is incredibly speedy and integrates seamlessly with HTML markup. HyperText Markup Language (HTML) and Cascading Style Sheets (CSS) for displaying web pages or any media being displayed on the web browser and can style any HTML element to change its dimensions, colors, borders, spacing, and others, but now users can also add animated transitions and transformations to your web pages, using only a few lines of CSS (Nixon, n.d.). JavaScript (Js) was incorporated due to its popularity and being adapted by major graphical web browsers; it has made modern web applications possible - applications with which users can interact directly without refreshing the webpage for every action (Flores, 2018). Web-based monitoring can apply to the proponent's current development of a monitoring system in the agriculture sector and can potentially be a positive outcome. A study in the same vein in 2022 developed a digital garden system with the implementation of an Arduino to grow Lettuce and Strawberries. The hardware of the monitoring system consists of a NodeMCU, a DHT11, a commonly used temperature, and humidity sensor, an AC light dimmer, an LED grow light, and a router. The hardware found in the hydroponic garden is sensors, actuators, and a cubicle enclosure to hold a controlled environment for the crops to grow. The monitoring system is displayed through a web-based system. A mobile application, a combination of HTML and PHP was implemented to display the data on the web page and the mobile application, along with a filter button from the web-based system to sift through the data which displayed the time, date, humidity, and temperature levels of the soil. SQL was the database used to store the collected data. The researchers concluded that the crops grew due to the proper environment created by the system indoors; it was noted that HTML and PHP are well suited for a monitoring system (View of Digital Garden System Using Arduino, n.d.).

Currently, the proponents will build upon this study by configuring additional sensors, limiting the monitoring system to a refined web-based model, and including Edge computing close to the sensors. Arduino microcontrollers typically use Web-based systems, Application-based systems, or any common system for data evaluation, databases, and others. The research was conducted in 2019 by Indonesian University students that designed an IoT data logging system for visual monitoring using an Arduino UNO R3 with an ethernet shield module and a temperature sensor (DHT11) in real-time. The system will measure important parameters such as the date time recorded using actual time, the Internet Protocol (IP) source address, IP destination address, the address in each node, device time, and room temperature. The output will be displayed as a table or graph on a web page constructed using a combination of PHP, CSS, and JavaScript programming. At the same time, MySQL will be the database used to store incoming data (Tsani & Subardono, 2019). Arduino's characteristic as the microcontroller makes it so that it cannot only transmit data to a monitoring system but also assist in activating devices as a result of a sensor reacting to stimuli and even adjusting parameters for a sensor. A study in 2021 used Arduino's characteristics to design a monitoring system to adjust the temperature of a vessel that required the vessel to be at a range of 40 degrees Celsius to 70 degrees Celsius. The user can visualize the change in temperature through access anywhere via a web-based monitoring interface (Gultom et al., 2021). The web-based system has proven to be a popular choice in terms of a medium for monitoring with the addition of CSS to create a user-friendly interface with MySQL's database, also sharing the same relatively user-friendly interface. The system can allow users to access the monitoring application anywhere in the vicinity and make well-informed decisions without delay.

Ⅳ. Modern-Day Agriculture with Agricultural Sensors

Given the fact that there is a noticeable growth in the human population in today's society, it is plausible to be an ongoing issue in the next few years. According to the United Nations, in 2017, the human population is expected to rise to 8.6 billion by the next 30 years (World Population Projected to Reach 9.8 Billion in 2050, and 11.2 Billion in 2100 | UN DESA | United Nations Department of Economic and Social Affairs, n.d.). Agriculture plays an essential role in the needs of the human race. It is the primary source of human beings' basic needs and necessities, such as raw and processed materials, for our daily lives (The Art and Science of Agriculture | National Geographic Society, n.d.). As the human population increases, the supply of nourishment to feed and sustain the rapid growth of individuals globally also increases. Considering the problems and crises that the agricultural sector is experiencing globally, agriculture needs to cultivate more strategically to provide a solution to meet the needs of the human population. The crises in agriculture, at this time, need more enhanced and advanced solutions through the current trends in technology. Hence, Smart Agriculture is being used and utilized in today's time. The concept of Smart Agriculture involves the use of sensors that relatively help farmers in managing their crops more efficiently. These agricultural sensors provide aid in many aspects of farm management. The use of intelligent sensors for agriculture is said to improve the monitoring system in agriculture farms. With that, various studies have proved that incorporating and embedding different sensors into a farming management system is a probable answer to the problems of farmers and other people involved in the agricultural sector.

According to a recent study in 2017 (Navulur et al., 2017), one of the applications of sensors in agriculture is knowing and understanding the condition of the soil chemistry in plants which is determined by running tests to check the pH, humidity, and temperature level of the surrounding field. Along with that is by running tests to check the moisture and nutrient content of the soil used for planting. There is a need to use Electrochemical Sensors such as Ion Selective Electrodes (ISE) and Ion Sensitive Field Effect Transistors (ISFET) sensors to monitor ion absorption in plants. Both of these sensors help to find pollutants among plants and water in the soil. The purpose of ISE sensors is to monitor the detection of nitrates in the mud and its production. These sensors work by estimating the growth rate and the status of the nutrient content of the plants. The results are reliant on the rate of nutrient absorption based on the needed nutrients of the plants. In another similar field of research, In 2020 (Ali & Alshmrany, 2020), there was a study on an Automatic Plant Irrigation System wherein a proposed system highlights the use of an Arduino Microcontroller integrated with the Internet of Things (IoT) to provide sufficient water supply to plants through a message alert. The sensors used are as follows: (1) Soil Moisture Sensor-- for measuring the moisture content of the soil in plants; (2) Humidity Sensor -- for estimating the amount of water vapor in the surrounding area; and Temperature Sensor -- for monitoring changes in air temperature. Aside from these sensors, the system also used other sensor nodes, PIR and Motion Sensors, to deliver an effective real-time farm monitoring system. In supporting the extraordinary demands for the betterment of the agricultural sector, it is much easier to incorporate an automated farming system to deliver the work more successfully (Mahbub, 2020). Embedding sensors - Agricultural Sensors can benefit not only the farmers striving hard to work and the market that consumes the farm end products but also the entire agricultural ecosystem for better advancement and prosperity.

Ⅴ. Smart Garden System

"Smart garden, dream garden" has become an inevitable trend in developing modern gardens. The construction of garden informatization is gradually changing the garden industry's survival and operation mode. It will make landscaping management more information-based and intelligent, lowering the cost of garden construction and greatly improving work efficiency (Xun & Ren, 2022). It is not new to install IoT systems in gardens. The online availability of sensors and microcontrollers has simplified the deployment of systems for monitoring plants, automating watering, providing artificial light, and disease detection in plants. Researchers created a technique for monitoring soil moisture, humidity, and UV radiation in shielded crops. Researchers introduced a wireless sensor network design for vegetable greenhouse growing to accomplish cultivation and minimize management expenses by monitoring temperature, humidity, and light. Similarly, in the context of greenhouses, researchers created a wireless sensor network utilizing open-source and low-cost hardware to assess the concentration level of different greenhouse gasses. The sensors, microcontrollers, data persistent storage, and actuators employed in these planters differ (Geng & Chun, 2022). Smart Garden typically consists of a core microcontroller to which additional items are linked. The smart garden comprises a microcontroller WiFi enabled such as NodeMCU as a hub to which various sensors like humidity, temperature, moisture, and ultrasonic sensors are linked. The ultrasonic sensor is linked to a water tank and displays the tank's water level. Other sensors are linked to their appropriate placements, and the data from these sensors is sent to NodeMCU, which has Wi-Fi technology. Firebase is an internet-accessible database that updates real-time sensor information every second. Android applications are created with the Android Studio program. The connection between the application and Firebase will be established within the program (Thamaraimanalan et al., n.d.). Another study using an Arduino Uno created a cloud-based Internet of Things (IoT) smart garden monitoring and watering system. Watering needs for plants may be altered by monitoring soil moisture. Measuring the soil moisture of the plant reveals if the plant is optimally watered, overwatered, or underwatered. Two garden parameters—soil moisture content and light intensity—are monitored and maintained by the system. Light intensity sensors and soil moisture sensors are used to achieve this. It continuously transmits the observed data to the IoT cloud, and the data collected from the system is also processed in the cloud. When a specified threshold of soil moisture is achieved, an action is sent from the cloud to the garden's autonomous watering system to irrigate the garden. Sunlight, which plants need for photosynthesis, is a vital source of energy. The device called an LDR Sensor reads the sunlight's intensity; the resistance of this gadget varies with the quantity of sunlight falling on it. We may use this equipment to see if the crops are growing correctly, provided with adequate sunshine, and take the necessary precautions to maintain the crops' health. The data from the sensors is frequently updated on an IoT cloud, and the user may verify the status of the water sprinklers at any time. Furthermore, sensor values are sent to an IoT Cloud channel to produce graphs for examination (Al-Omary et al., 2018). More sustainable farming may be accomplished with the help of IoT Systems and flexible technology that provides a wide range of applications and functions. There is reason to be optimistic since the fast growth of IoT applications in agriculture is changing traditional farming into smart farming (Saravanan M., 2021).

Ⅵ. Growing a Lettuce

Lettuce is a prominent salad vegetable in temperate areas. Lettuce is said to have originated in an area that included the Mediterranean and Siberia. Lactuca sativa is the scientific name for lettuce. It is a hardy annual herb plant of the Asteraceae family, also known as the sunflower family. Though it is grown for its delicious leaves, its seeds and stems are commonly consumed as food. Lettuce is a relatively simple plant to grow. Lettuce plants are annual herbaceous plants that grow 6 to 12 inches tall (15 to 30 cm). The leaves are brightly colored, primarily green and red. The plant has a tap root system that consists of a primary taproot and smaller subsidiary roots. Flower stalks with little yellow blooms can grow up to 3 feet (0.9 m) tall. Its fruit is covered in two rows of tiny white hairs and has 5-7 ribs on each side. Each fruit has one seed.

The color of the seeds varies according to the cultivar. Depending on the variety, it might be white, yellow, gray, or brown. There are two types of lettuce on the market: head-forming and non-head-forming. The most common non-head-producing lettuces are leaf lettuce (red-leaf and green-leaf variations) and romaine lettuce. Crisphead lettuce and Butterhead lettuce are lettuce varieties that produce heads of lettuce. Leaf lettuce is also known as loose leaf lettuce or bunching lettuce because it creates a loosely bunched cluster of leaves at the top of the stem when mature. Leaf lettuce is classified into red-leaf and green-leaf (Growing Vegetables: KnolKhol, Lettuce and Zucchini - Roby Jose Ciju - Google Books, n.d.). The figure below shows guidelines for producing Lettuce at an optimal yield (Crop Production Manual: A Guide to Fruit and Vegetable Production in the ... - Food and Agriculture Organization of the United Nations - Google Books, n.d.).

Lettuce (*Lactusa sativa*)

© FAO/ K. Hadfield

Recommended varieties:

Head Type:

- Great Lakes
- Boxhill

Leafy Type:

- Green Mignonette
- Butter crunch
- Rapid

Seed rate:

- 300 grams/ha

Planting time:

Head Type:

- March to October

Leafy type:

- All year round

Spacing:

- Between rows: 75 cm
- within rows: 30–40 cm

Germination:

- 5 to 8 days after sowing
- Raised the seedlings in seed trays/seed bed and transplant at 3 to 4 leaf stage
- Seedlings raised in seed trays

Fertilizer:

- Poultry Manure: 5 tonnes/ha
- Broadcast and mix well with soil 2 weeks before planting

Weed management:

- Manual weed control
- Inter row cultivation

Disease management:

- Practice crop rotation
- Avoid planting with high density
- Remove all left over crops

Harvest

- Leafy Lettuce matures in 50–80 days
- Head Type matures in 12–15 weeks

Yield:

- 8–10 tonnes/ha

Food value:

- Dietary Fibre, Source of Vitamin A,
- Vitamin B and C

Figure 2: Growing a Lettuce.

The seed rate pertains to the quality of a certain seed sowed in a unit area of soil for optimum crop production. The seed rate is affected by various factors, from desired plant population per unit area (DPP/UA) and germination percentage (PG) to factors that do not have causal effects, such as soil fertility and productivity, growing season, and even method of planting. Plant time defines when the crop should be planted to attain optimal output. The Head type (Great Lake or Box Hill) variation of lettuce has to be planted from March to October, while the Leafy type (Buttercrunch to Rapid) can be grown all year. The plant time may vary from region to region (Hasanuzzaman, n.d.). Spacing is an important factor when planting to give the crop enough room to grow but close enough to be considered companion planting, which assists similar crops in pest control (Crop Production Manual: A Guide to Fruit and Vegetable Production in the ... - Food and Agriculture Organization of the United Nations - Google Books, n.d.) (Lettuce Production Guide, n.d.). Germination begins a plant's life cycle to produce a healthy seedling able to grow, develop, and generate offspring. This is an important stage as it influences the quality and yield of the crop (Boter et al., 2019). Fertilization should be based on crop and soil analysis; fertilization can come in the form of organic or inorganic, with organics being the popular choice. Organic fertilizer can improve the properties of the soil, such as water-holding capacity and soil structure, which in turn allows better root growth and crop yield as a whole. Manure is a prime example of organic fertilizer with its high carbon and carbon compound contents which assists in building and fortifying the soil but contains low nutrient content (Crop Production Manual: A Guide to Fruit and Vegetable Production in the ... - Food and Agriculture Organization of the United Nations - Google Books, n.d.) (Lettuce Production Guide, n.d.). Weed management is required to prevent the growth of weeds, failure to remove weeds will only allow the weed to compete with the crops for absorption of nutrients, sunlight and space, thus, slowing the growth of crops (Lettuce Production Guide, n.d.). Disease management is an important process of preventing or minimizing the number of unmarketable crops; proper spacing is practiced by giving the crop the correct room for growth. Crop rotation is another method of disease management which, by growing crops on the same plot of land each season in succession, aims to create unfavorable conditions for the pest by disturbing the host-specific pests. Crop rotation also improves the soil, optimizes soil nutrients, and provides weed and disease management (Crop Production Manual: A Guide to Fruit and Vegetable Production in the ... - Food and Agriculture Organization of the United Nations - Google Books, n.d.) (Boter et al., 2019) (Crop Rotations - Rodale Institute, n.d.). Lettuce is particularly difficult to produce due to the Philippines' heavy rains; greenhouses or even housing are required to harvest and produce the crop (Gonzaga et al., 2017). Conducting the proper research and guides can help understand the characteristics of the crop and how to produce high yields.

Ⅶ. Parameters for Controlled Environment Agriculture

A farming setup in a controlled environment is one of the possible solutions to the problem of Agriculture in the Philippines, as the country's prone to natural calamities like typhoons and unpredictable weather conditions. Mainly, smart farming promotes producing crops and food through the utilization of natural resources beneficial for supplying the increasing food demands of the country (Singh et al., 2020). For supporting a controlled environment farming setup, research on crop production experimented under controlled conditions identifies the threshold values for abiotic factors such as carbon dioxide, light intensity, relative humidity, moisture, temperature, and water level that plants need. Carbon dioxide (CO2) concentrations in a greenhouse relatively affect and influence plant growth by increasing photosynthesis. In a typical outdoor environment, the carbon dioxide concentration is approximately about 380 parts per million (ppm) up to a maximum level of 1500 parts per million (ppm) (Semilla et al., 2018). However, a greenhouse environment requires increasing the carbon dioxide concentration to the recommended level of 800-1200 ppm will increase and boost the yield of Lettuce growth. Another factor to consider is the light intensity to produce lettuce in either indoor or outdoor systems. Zhou, J et al. (2019) suggested that the range of light intensity to grow lettuce is 400 to 600 $\mu mol \cdot m^{-2} \cdot s^{-1}$. However, in an indoor setup for growing lettuce plants, Paz, M et al. (2019) stated that a lettuce plant needs at least 6.5 to 9.7 $mol \cdot m^{-2} \cdot d^{-1}$ of light intensity daily. With a minimum daily light integral of the said range, a lettuce plant can develop through an indoor setup equipped with the preferable nutrients and growth level. Additionally, according to (Paz, M et al., 2019), daily records of light intensity below 6.5 $mol \cdot m^{-2} \cdot d^{-1}$ shows that lettuce is susceptible to withering and losing its freshness. Hydroponic Lettuce Handbook by Dr. Melissa Brechner & Dr. A.J. Both (2020) proposed that relative humidity in the greenhouse influences the water relations of plants. If the relative humidity in plants reaches a high level, the process of water transpiration is at risk causing the restriction of the attraction of nutrients from the soil. Soil moisture is also one of the crucial elements influencing the series of lettuce crop production. According to (Acurite, 2018), most crops and vegetables mainly require at least 41-80% of soil moisture level. However, Tagle et al. (2018) suggested that the relative humidity level to grow lettuce is approximately 50-80%. Relatively, the soil's moisture content involves the dryness of the soil used. Aside from the mentioned parameters, other parameters include the ambient temperature and water level that lettuce requires. A lettuce crop certainly needs a sufficient amount of water to grow. A minimum of 2 inches (5 cm) of water per week should be provided to a lettuce crop. Depending on whether the surrounding area strikes a higher temperature, it is advisable to gradually increase the amount of water by 1.5 to 2 inches (4-5 cm) weekly. Note that the average water level for lettuce is not specified due to the other factors to consider, such as the soil's moisture content and temperature, that need to be assessed throughout the growing process of lettuce. Shown in Table 1 are the data that signifies the standard level of requirement that the Lettuce plant needs.

Table 1: Standard Environmental Conditions of Lettuce.

Name	Ideal Values
Carbon Dioxide	800 - 1200 PPM
Relative Humidity	40 - 80%
Soil Moisture	50 - 80 %
Temperature	18 ℃ - 26 ℃
Light Intensity	17, 391.30 to 26086.96 Lux
Light Intensity (Indoor)	282.60 to 421.74 Lux

The data range shown in table 1 above is subject to change, particularly in response to the farm’s location and other unforeseen environmental changes like calamities (Tagle et al., 2018). In line with observations and tests gathered from various research and studies, this set range shows the threshold values required to meet to grow lettuce crops under an automated controlled environment.

Ⅷ. Synthesis

Various variants of lettuce emerged and became more abundant in many local marketplaces here in our country. As the demand for this product increases, the supply is also at risk of rising correspondingly. Lettuce is a delicate leafy crop that can be hard to manage when the country is affected by a calamity, like typhoons that bring heavy rains and flooding to agricultural farms (Gonzaga et al., 2017) (Singh et al., 2020). Hence, lettuce production can be a problem for the farmers in our country. Additionally, the Philippines is well known for its tropical climate, which makes the production of lettuce a laborious job for farmers as lettuce can grow more in colder weather conditions. Thus, different emerging technologies have become apparent today to solve problems and provide aid to farmers for better farming management. One of these emerging technologies is a Smart garden system for farming. This system is an economical gardening solution to enable plants to grow healthier through a modern approach than traditional gardening. This modern approach to gardening promotes solutions with less human intervention.

However, to start a Smart garden system under a controlled environment, it is essential to know the standard level of measurement for abiotic factors like ambient carbon dioxide, light intensity, relative humidity, soil moisture, ambient temperature, and water level. Various research studies have identified the standard level of these abiotic factors that a lettuce crop needs. Nevertheless, these numerical figures are alterable and changeable depending on and concerning different environmental circumstances that involve the lettuce plant (Tagle et al., 2018). Nonetheless, these threshold values can be used as the basis for guidance and reference to see whether a modern approach to gardening will be more advantageous than still relying on the traditional farming method.

Furthermore, it utilizes automation and technology to make farming simpler through agricultural sensors used for various purposes, such as monitoring the factors that inhibit the growth of plants—water, temperature, light, and nutrients; and a microcontroller that corresponds to the system's brain (Thamaraimanalan et al., n.d.) (Al-Omary et al., 2018) (Ali & Alshmrany, 2020). For better visualization and to see the data gathered from the sensors, a Smart garden system needs a system monitoring tool such as a web-based interface. It can include front-end development mediums such as HTML & CSS and back-end development mediums such as PHP and JAVASCRIPT (Nixon, n.d.) (Flores, 2018). These can help view results and observations from the system as user-friendly and efficient as possible. Moreover, a Smart garden system can assimilate the Internet of Things (IoT) by using interconnected devices as a medium for communication over the cloud (internet). IoT is a novel technology that caters comfort to people in smart working and living (Internet of Things (IoT) for Automated and Smart Applications - Google Books, n.d.). It offers convenience to different industries, like agriculture, as it aids in improving the quality of harvests in many farms and provides accuracy in the real-time monitoring of crops in farms. Another emerging technology named Edge Computing can also help improve the quality of work on an agricultural farm. Although processing data with edge computing comes in larger volumes, it reduces the latency issues with data processing. It also saves the system from crippling downtime, leading to damage to the properties and even the people involved in the industry.

The following topics of interest stated conclude the context of this design project. By integrating and combining every subject of interest mentioned, the proponents came up with the completion and development of a smart garden setup provided with an adaptable and user-friendly monitoring system for local farmers. However, due to some constraints, such as the selected market of the proponent's design as well as the advancement and functionality of the whole system, the design project cannot cater to other established farms as it may be restricted budget-wise and time-wise. Along with this, the proponent's design is adaptable and applicable but with modifications based on the crop of choice by other farmers.

Chapter Ⅲ

Design Project Methodology

Ⅰ. Overview of the Design Project Methodology

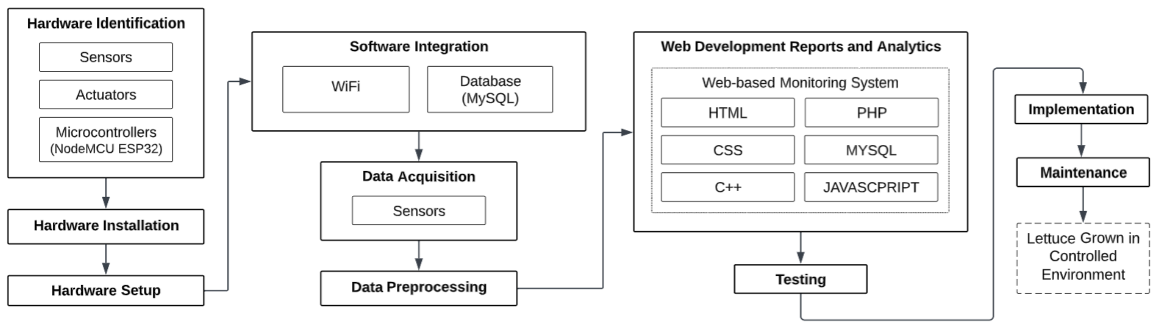

Figure 3. Overview of the Design Project Methodology

The proponents will decide on prototyping as the development methodology of this design project. The prototyping methodology includes developing, continuously testing, and reworking a system to ensure that the design is acceptable and compatible with development. This methodology allows engaged participation among the potential end user and the proponents of the model. Also, prototyping permits an increased development speed in improving the potential risks and issues associated with the system and its service to its end user. Figure 3 displays a graphical representation of the process used for the system by the proponent. The methodology will begin with hardware identification, where proponents specify hardware components such as microcontrollers, sensors, and actuators needed to provide the lettuce, even in a controlled environment. To be followed by hardware installation showing how the interconnection between the hardware components will transpire. The actuators will be connected to the microcontrollers by the relay module to speed up the link between hardware components and act upon the data based on the readings of the sensors. The sensors that will be in the system are as follows: temperature, soil moisture, humidity, carbon dioxide, light, and water level sensors. The third process in the system is the hardware setup, determining how the proponents will set up the hardware system on miguela's farm. For this, the system will be in the seed germination and growing process in the greenhouse of miguela's farm. The sensors will be used for the seed germination process to initiate the procedure of turning seeds into healthy-grown lettuce plants and for the actual growing process for monitoring and stabilizing the growth progress of lettuce plants. To support the hardware system, the proponents will introduce the use of database software - MySQL Database for handling and storing the data that will come from the hardware system.

On the other hand, WiFi will be present in the system for linking the hardware and the software components of the proponents' system. In data acquisition, only the sensors will be the basis for gathering the appropriate data before it goes through preprocessing. Preprocessing will include preparing the relevant data before raising and displaying it to the Web-based monitoring system that the proponents will develop. This monitoring system will also cater to reports and analytics for further assistance to the end-users on how they can quickly monitor and control the process of growing Lettuce plants. Various testing of the components of the whole system will take place to evaluate and verify whether the system will do what it is supposed to do, prevent errors, and find potential defects in the system. Assuming that the system will be successful, work comes to the stage of implementation and deployment, where end-users will freely use and test the system that the proponents will propose. Lastly, the maintenance of the whole system will materialize. In this stage, the system will successfully deliver its objectives and is compatible with the potential end-users of the model. This stage will also identify that the proponents' system will surely cater to what the end users demand.

Ⅱ. Respondents / Beneficiaries of the Design Project

The findings of the design project will serve as documentation to convey the essence of the internet of things (IoT) for the refinement of the agricultural sector. Moreover, the design project findings will benefit miguela's farm**,** the largest hydroponics farm in angono, rizal. Miguela's farm caters to nurturing and cultivating lettuce of different varieties through the science of hydroponics. They also offer lettuce-picking services for people who want to visit and are busy with freshly harvested sorts of lettuce**.**

Ⅲ. System Architecture

In figure 4, The first layer of the system consists of sensors and actuators; the following sensors are a carbon dioxide sensor, humidity sensor, light sensor, soil moisture sensor, temperature sensor, and water level sensor, while the following actuators are AC fan, exhaust fan, grow light, submersible pump, and two water pumps. If the data read by the sensors and the written condition in the program complement, the actuators will automate based on the given condition. A relay module connects the microcontroller and actuators to bridge the low-powered microcontroller with the high-powered actuators. The second layer is the edge; several microcontrollers are connected to the sensors and actuators. These microcontrollers will collect and, more importantly, process the raw data into processed data before sending the data to layer four via Router/WiFi. The third layer, WiFi, will enable layers two and four to exchange data. The host machine can gather the processed data in layer four using an HTTP post request. Using PHP script, the processed data will be inserted into the database. There will be two tables, one is for storing processed data, and the other is for storing relevant data. Layer five will house the web application displaying the near real-time processed data and the relevant data accessed from the localhost server MySQL database. The monitoring system will also allow the user to manually activate or deactivate the system's actuators. Different devices can also connect to the web-based monitoring system if the users are connected to the same local area network. The data from the web application will also be stored in the database; using HTTP post request and PHP, the microcontroller can retrieve the data from the database to the host machine to control the actuators. Furthermore, the system will have a second database that will act as a backup in case the first database fails to be accessed, is corrupted, or had data loss.

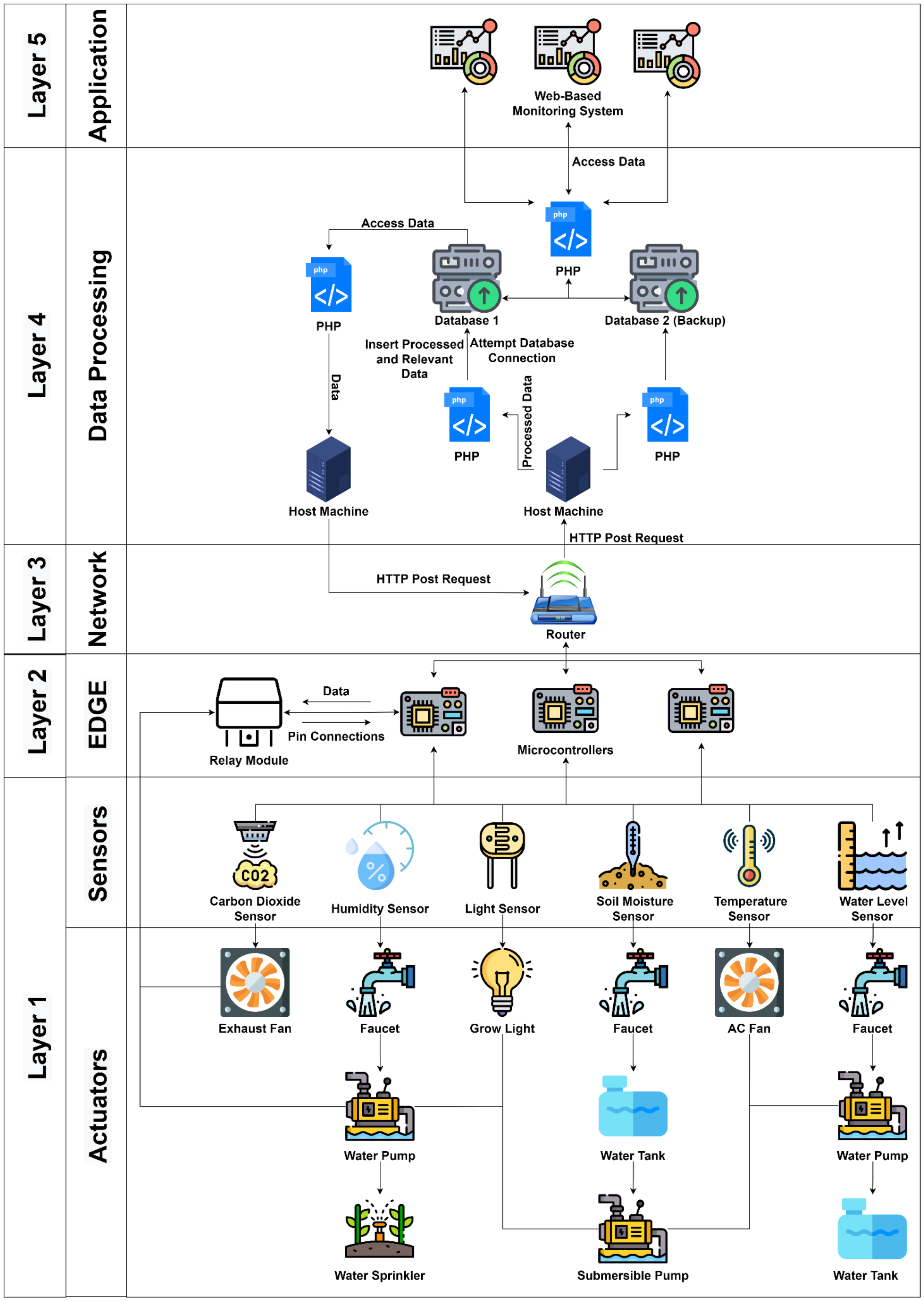

Figure 4: System Architecture.

Ⅳ. System Block Diagram

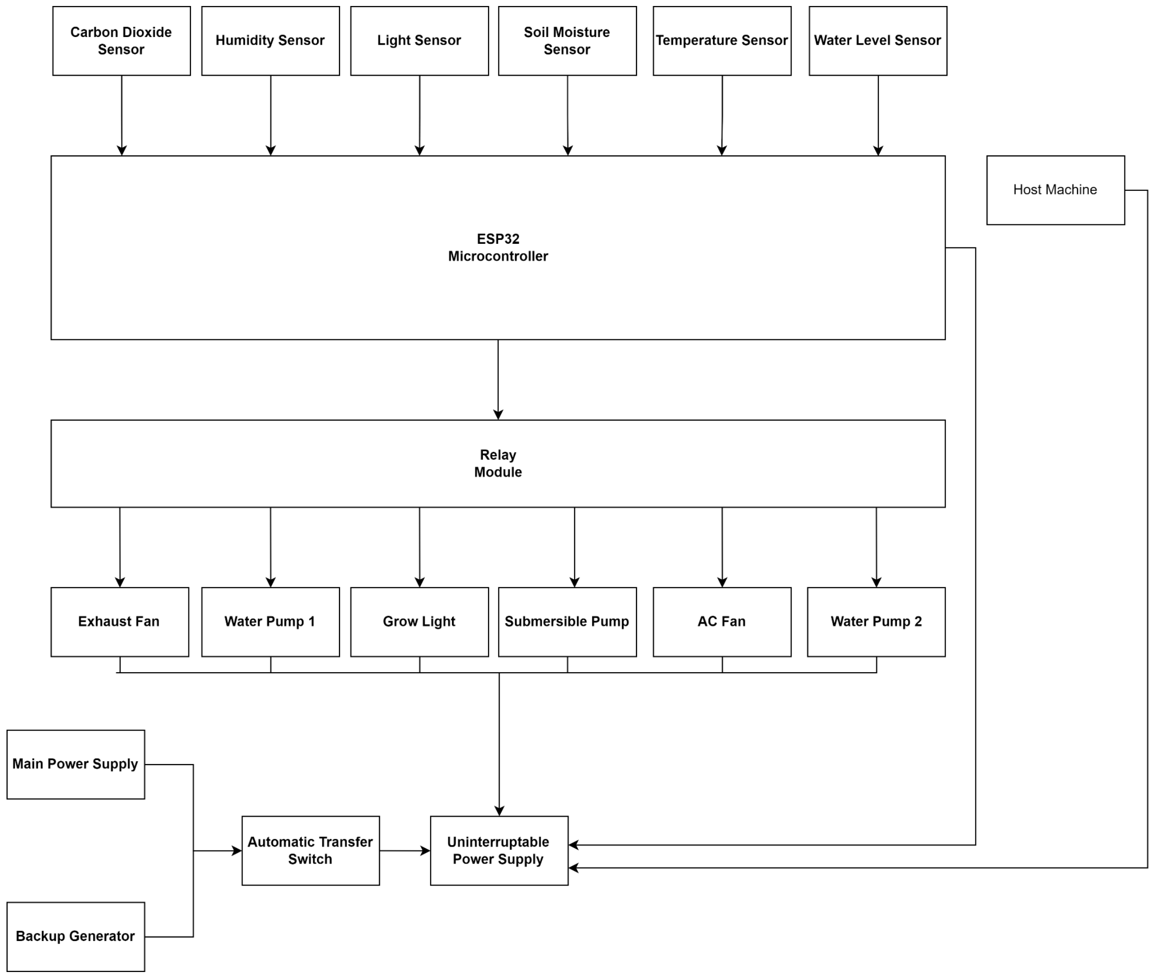

Figure 5: System Block Diagram.

In figure 5, All sensors and actuators are connected to the ESP32 microcontroller. Since all actuators require high voltage, the actuators cannot be directly connected to the microcontroller; it will require a relay module. The relay module will bridge a low-powered microcontroller and high-powered actuators. The relay module will allow the microcontroller to send a signal to the high-powered actuators without damaging the microcontroller and the connected devices. The actuator of the carbon dioxide is an exhaust fan; the humidity sensor is the water pump 1; the light sensor is the grow light;

the soil moisture sensor is the submersible pump; the temperature sensor is the AC Fan; the water level sensor is the water pump 2. Lastly, the system will have two sources of power the primary supply of electricity will be the main power supply, and the backup power supply will be the backup generator. The two power supplies are connected to an automatic transfer switch connected to the uninterruptible power supply. The host machine and the actuators are also connected to the uninterruptible power supply. Furthermore, an uninterruptible power supply will act as a battery until the backup generator is operational and ready to power the system. The automatic transfer switch will switch from the main power supply to the backup power supply when it detects that it fails to produce electricity.

V. System Flowchart

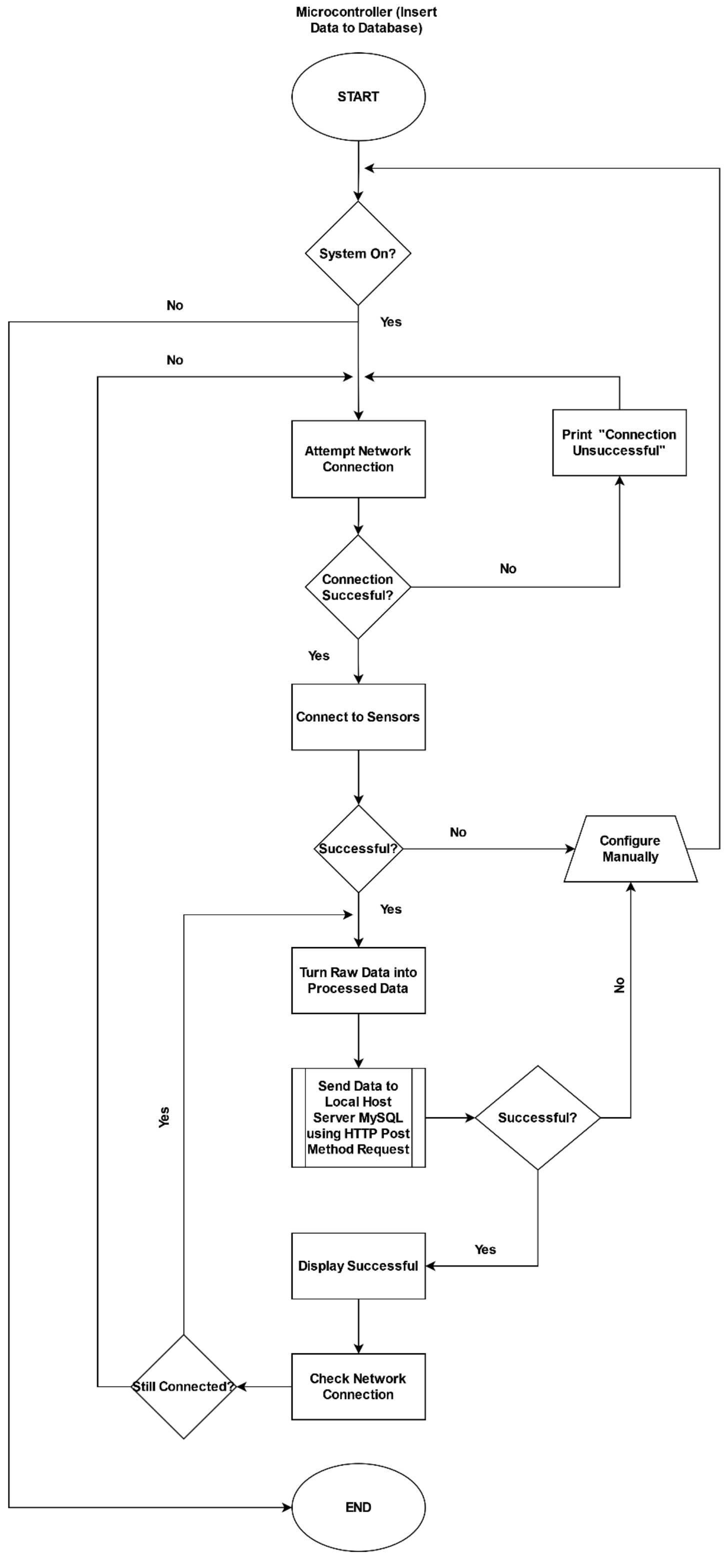

Figure 6: Insert Data to Database (Microcontroller).

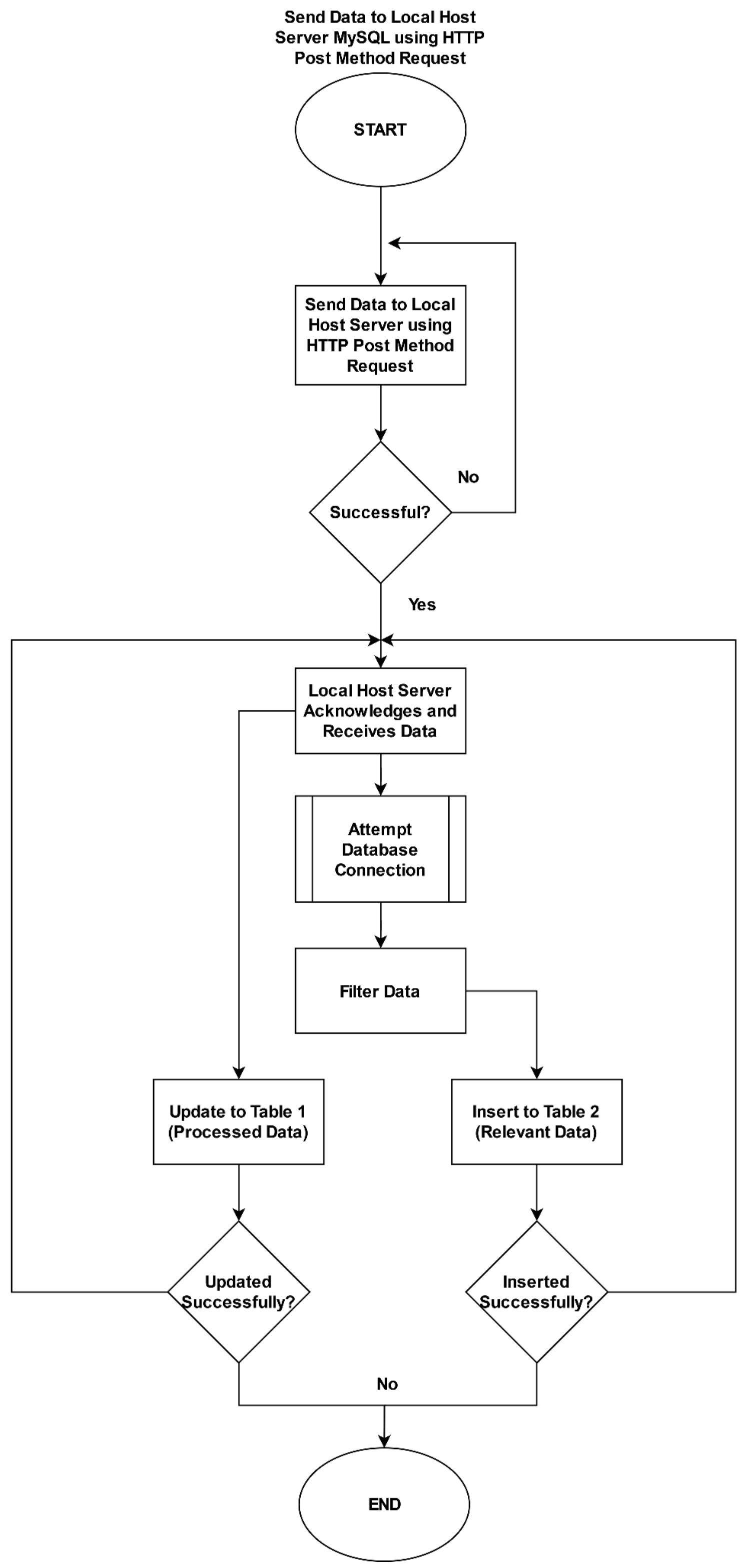

Figure 7: Send Data to Local Host Server MySQL using HTTP Post Method Request.

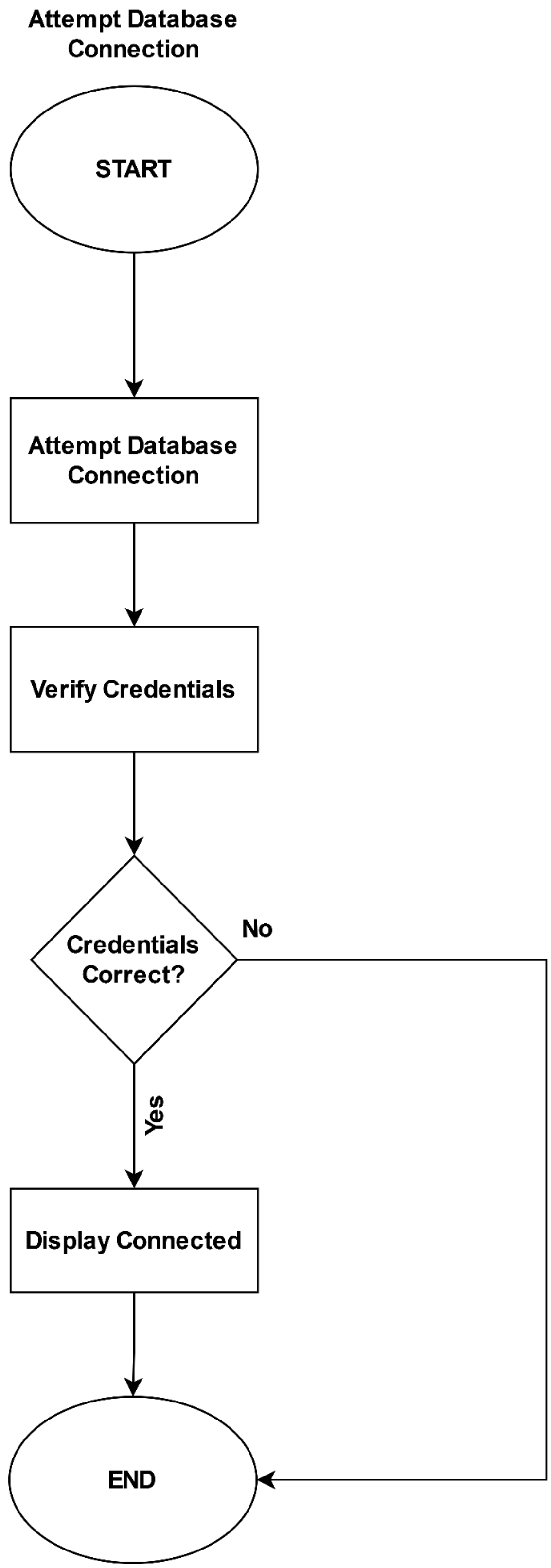

Figure 8: Attempt Database Connection.

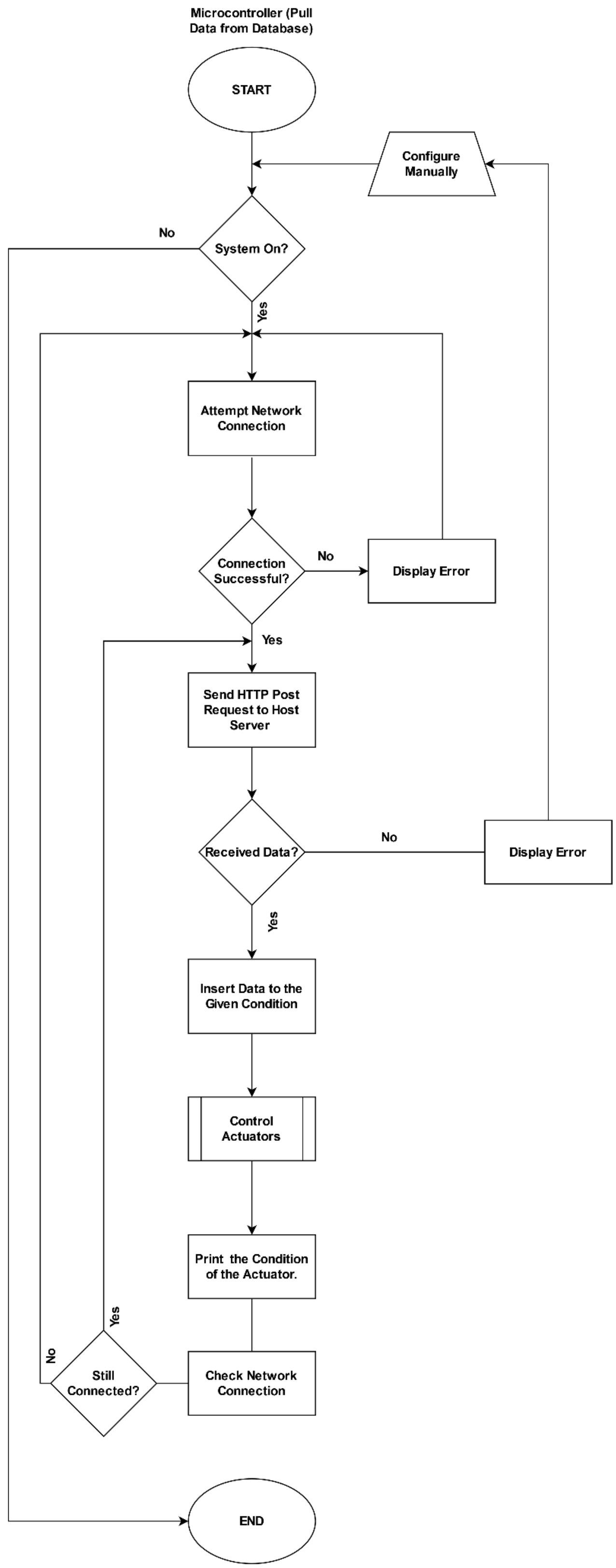

Figure 9: Pull Data from Database.

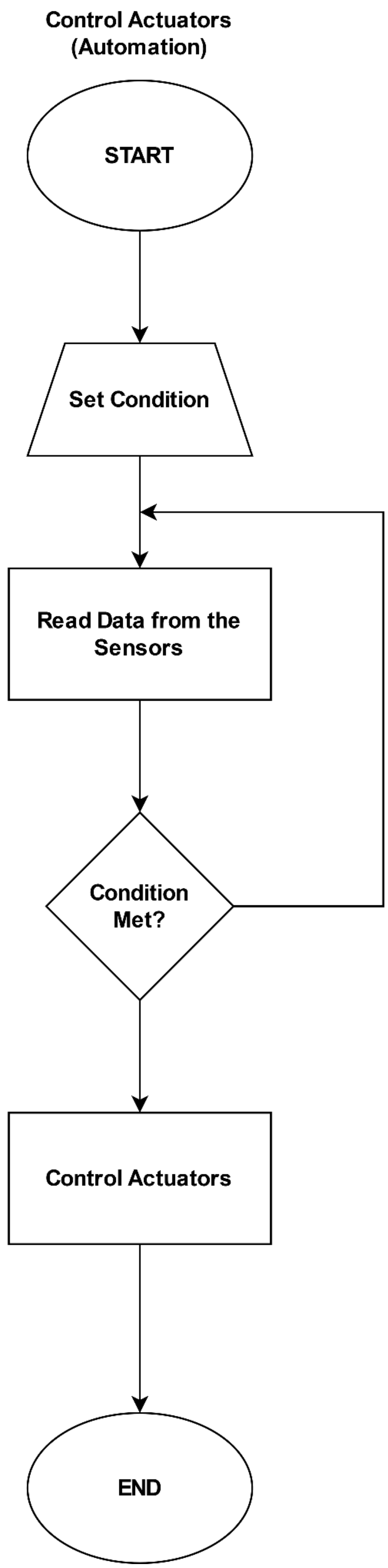

Figure 10: Control Actuators (Automation).

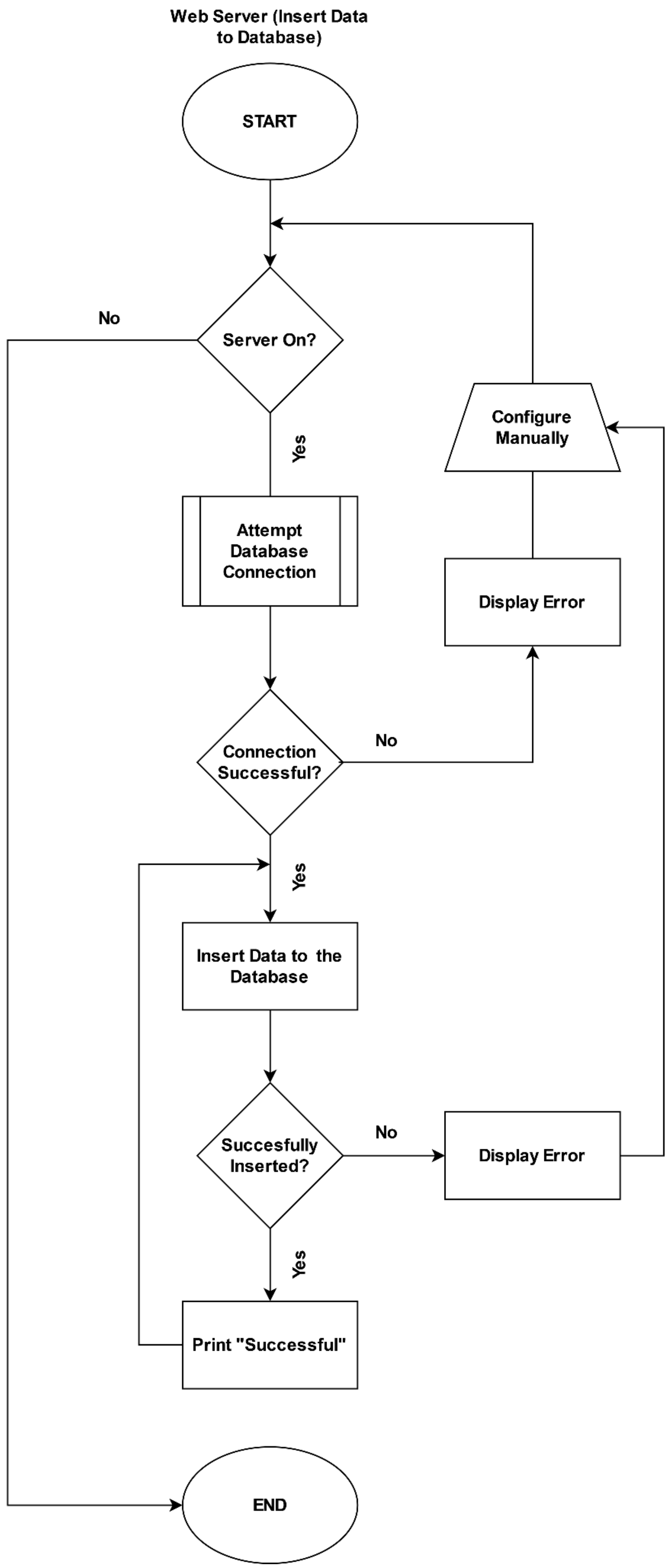

Figure 11: Insert Data to Database (Web Application).

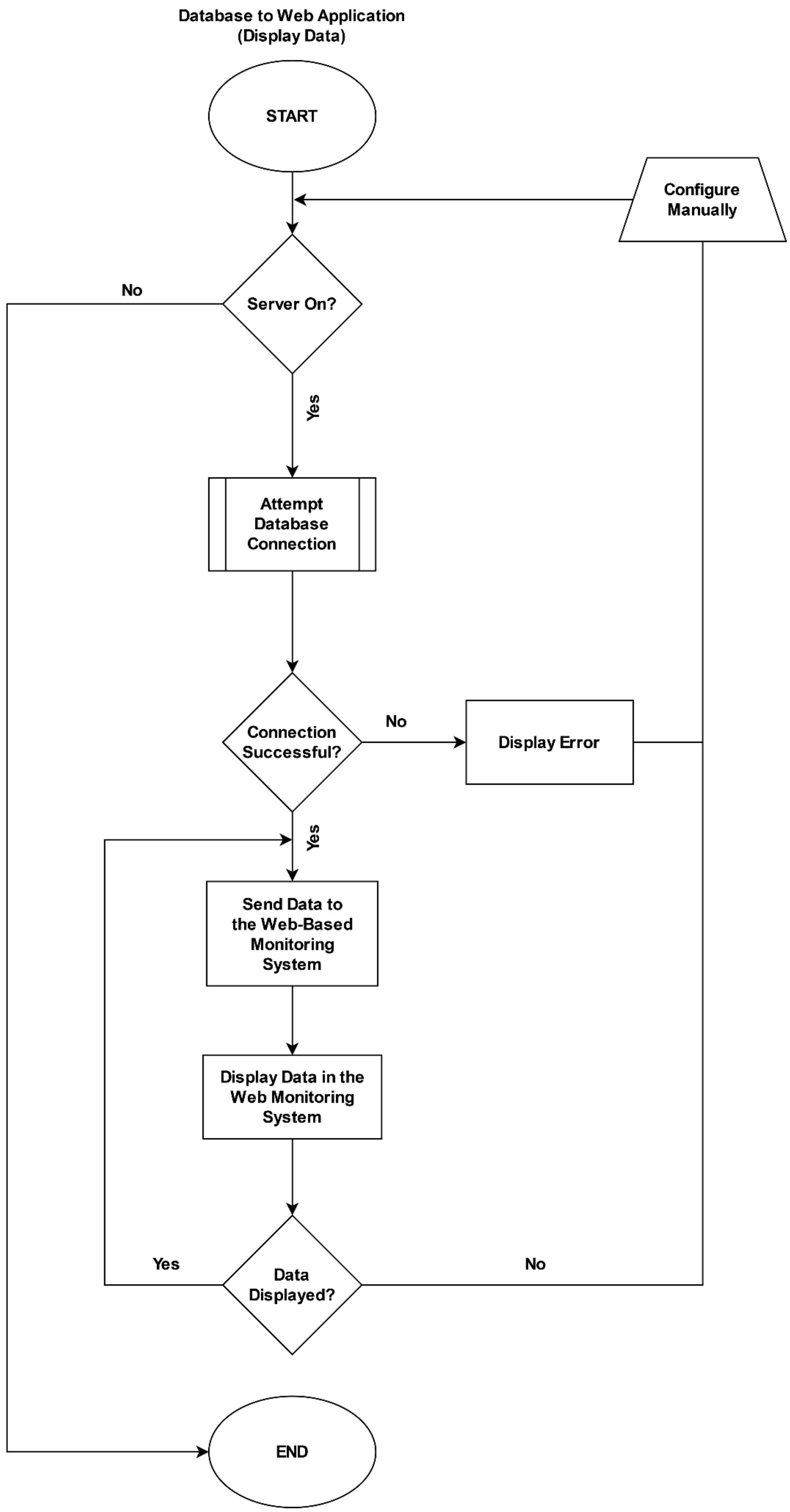

Figure 12: Display Data to Web Application

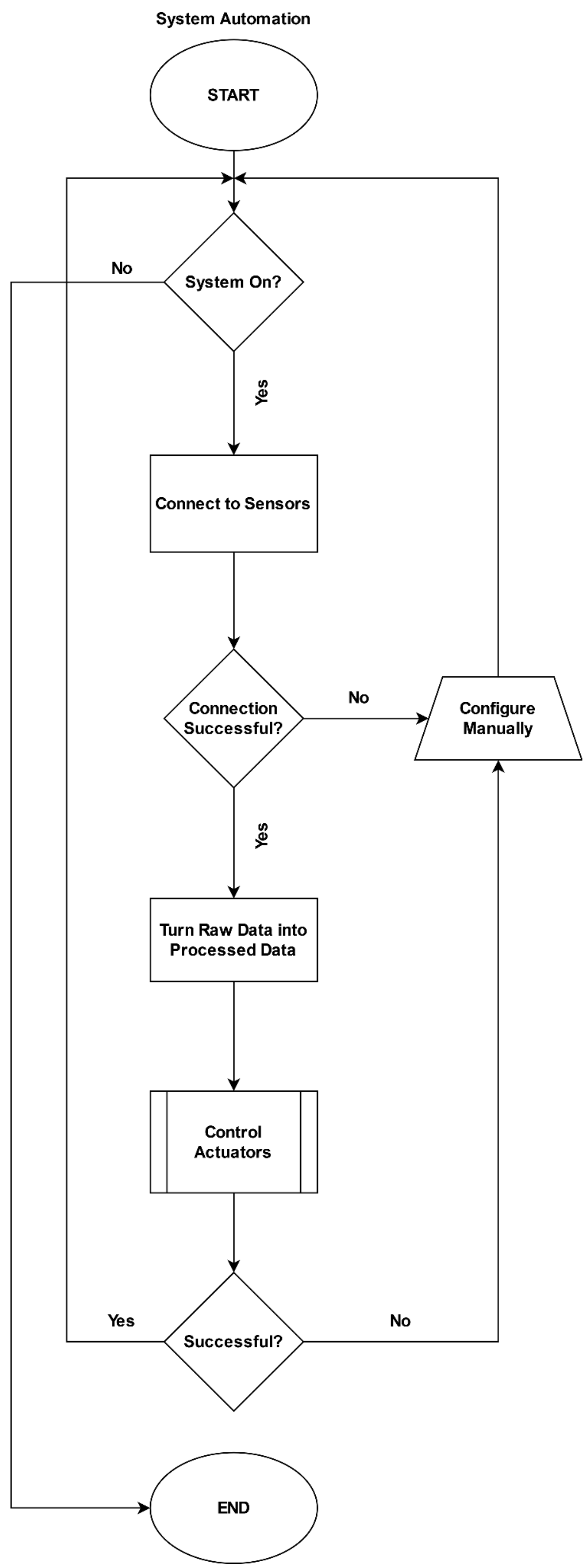

Figure 13: System Automation.

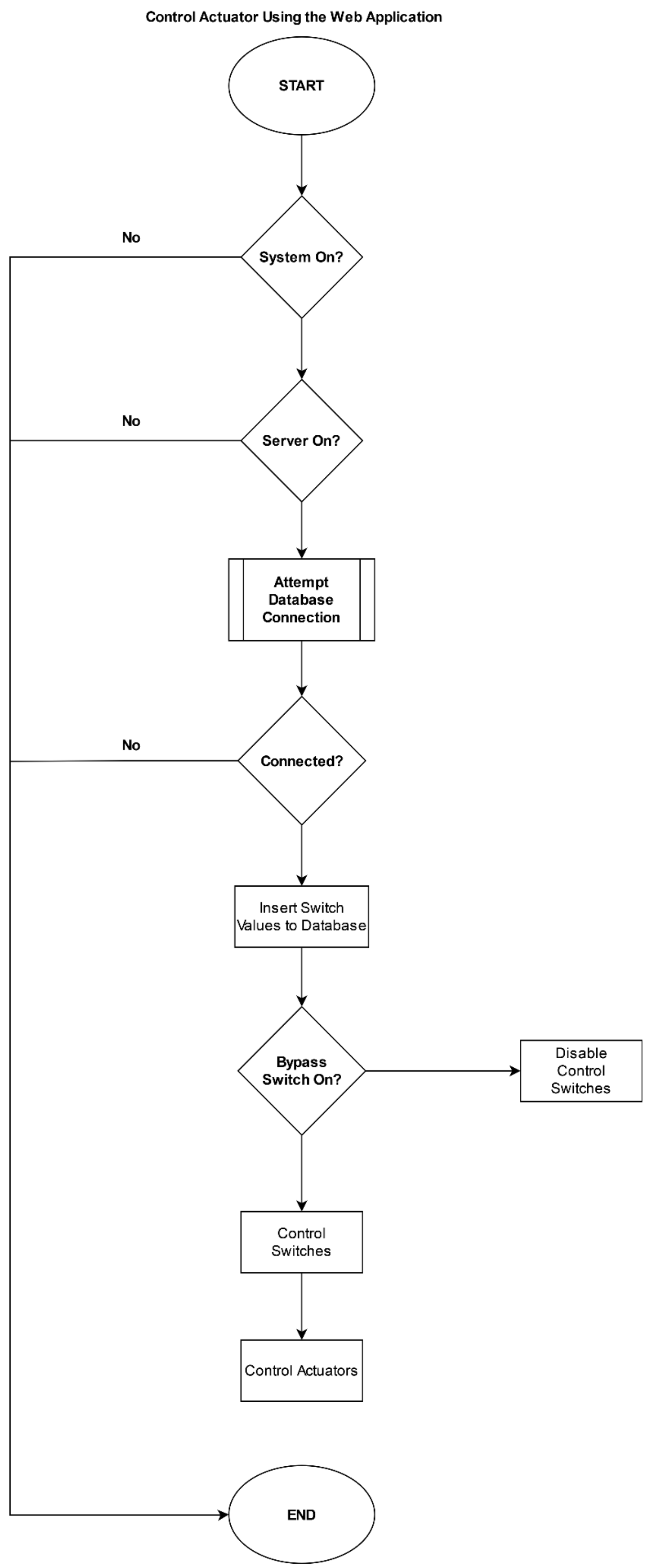

Figure 14: Control Actuator Using the Web Application.

Ⅵ. Hardware Requirements

The proponents considered the availability, cost, and specifications of the hardware listed below.

ESP32 Microcontroller

In figure 15, besides the capability to interact with the sensors, the ESP32 has a WiFi built-in module, and the proponents can use the microcontroller wirelessly to send data to the web-based monitoring system.

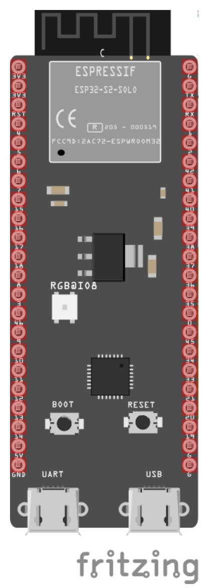

Figure 15: ESP32 Microcontroller.

MQ135 Air Quality Sensor Module (CO_2 Sensor)

In figure 16, it senses the level of carbon dioxide in the immediate vicinity.

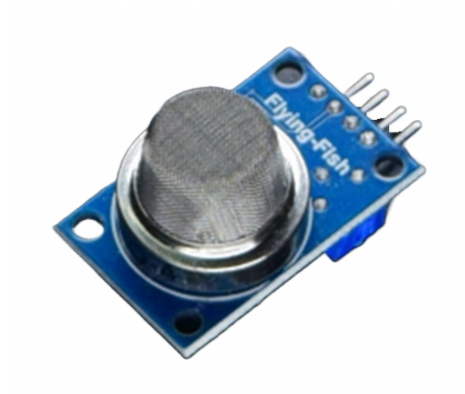

Figure 16: MQ135 Air Quality Sensor Module (CO_2 Sensor).

AM2320 Digital Humidity and Temperature Sensor

In figure 17, This device senses the amount of moisture in the air and senses the temperature in a given area.

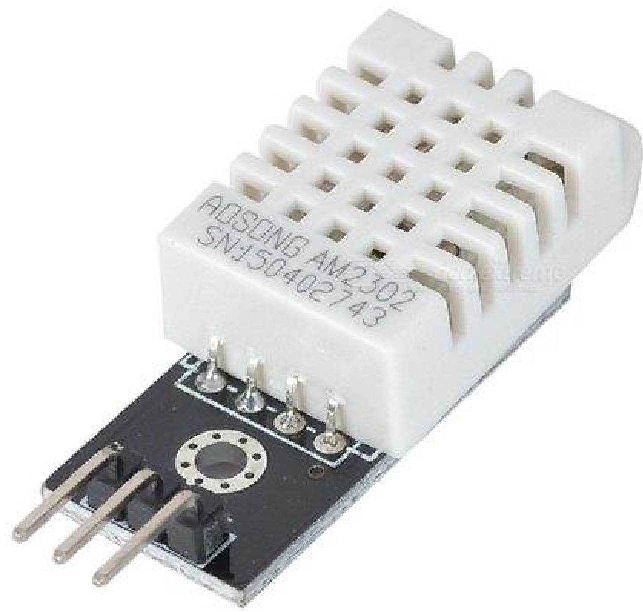

Figure 17: DHT22 Temperature and Humidity Sensor.

Gravity: Analog Ambient Light Sensor

In figure 18, This device senses the amount of light present.

Figure 18: Gravity: Analog Ambient Light Sensor.

Soil Moisture Sensor Corrosion Resistant

In figure 19, the soil moisture senses the moisture content.

Figure 19: Soil Moisture Sensor Corrosion Resistant.

Float Sensor

In figure 20, the float sensor senses the water level in the tank.

Figure 20: Float Sensor.

Miscellaneous

Air Cooler

In figure 21, the evaporative air cooler will be an actuator that will adjust the temperature of the surrounding area of the lettuce plants. The actuator will adapt based on the temperature given by the temperature sensor and the condition of the program.

Figure 21: Air Cooler.

Grow Light

In figure 22; it will be used as an actuator for the light sensor. Grow lights are good at mimicking natural light, which is helpful for growing indoor plants.

Figure 22: Grow Light.

Submersible Pump

In figure 23, It is an actuator for soil moisture. It will only be activated when the soil is below the required moisture level.

Figure 23: Submersible Pump.

Water Pump

In figure 24, the water pump will be an actuator for the humidity and water level sensor. It will only be activated when the required humidity level falls below the required humidity level and when the tank is almost empty.

Figure 24: Water Pump.

Automatic Transfer Switch

In figure 25, The automatic transfer switch, when it detects a failure or outage in the primary power source, an automatic transfer switch (ATS) automatically switches the power supply from the primary source to the backup source.

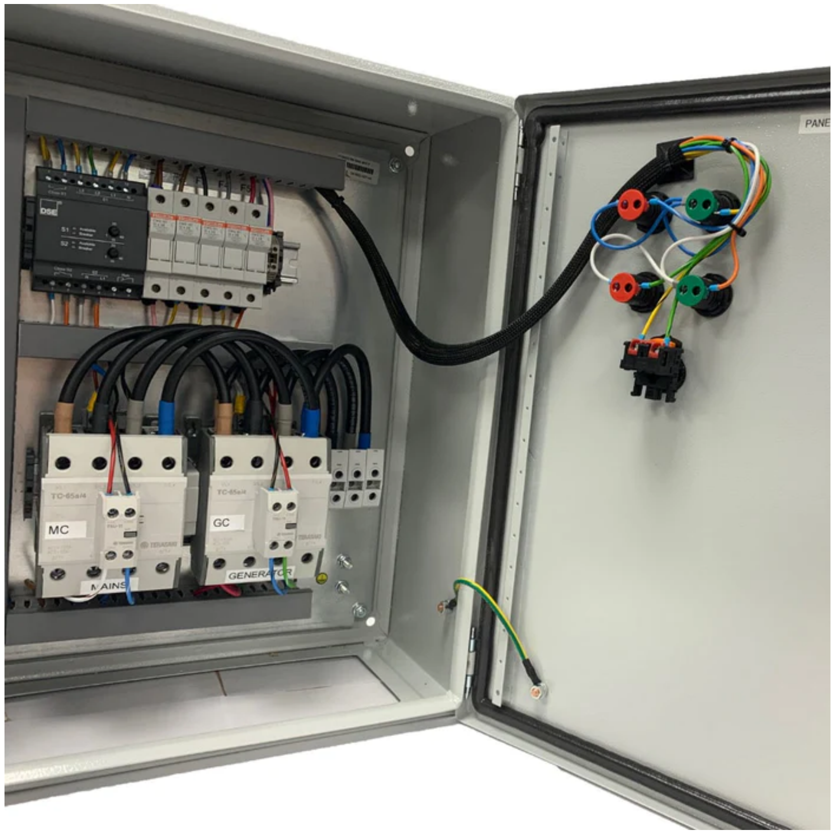

Figure 25: Automatic Transfer Switch.

Backup Generator

In figure 26, An automatic backup electrical system is known as a backup or standby generator. An automatic transfer switch detects the power loss immediately after a utility outage, starts the generator, and then switches the electrical load to the generator. The backup generator starts powering the circuits.

Figure 26: Backup Generator.

Exhaust Fan

In figure 27, the water pump will be an actuator for the carbon dioxide sensor. It will only be activated when the required carbon dioxide level goes above the required carbon dioxide for the lettuce plants.

Figure 27: Exhaust Fan.

Relay Module

In figure 28, a low-powered microcontroller and high-powered actuators will be connected through the relay module. The microcontroller can signal the high-powered actuators using the relay module without harming the microcontroller or the linked sensors.

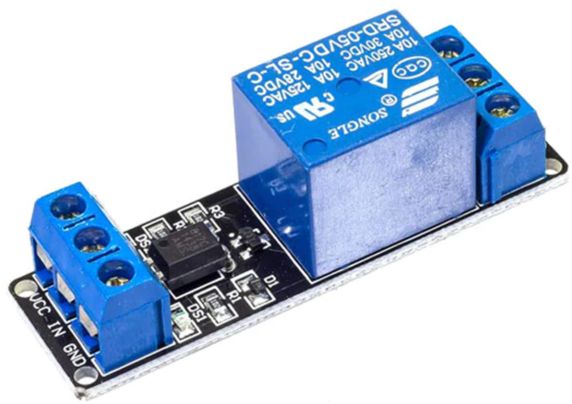

Figure 28: Relay Module.

Uninterruptible Power Supply

In figure 29, an electrical device known as an uninterruptible power supply (UPS) or uninterruptible power source (UPS) can temporarily supply a load with power in an emergency if the input power source or mains power fails.

Figure 29: Uninterruptible Power Supply.

Ⅶ. Software Requirements

Below are the software requirements:

- **C++** - To interact with the ESP32, the proponents will need C++ since it is the default programming language for the microcontroller.
- **CSS** - The proponents will be using CSS for the front-end framework of the web-based monitoring system; specifically, the proponents will mainly use bootstrap CSS.
- **JavaScript** - For the back-end of the web-based monitoring system, the proponents will use JavaScript since it is mainly used for developing web

applications. JavaScript will also serve as a bridge between the variables used in C++.

- **HTML** - This is one of the building blocks for creating a website besides CSS and JavaScript.
- **MySQL** - It will serve as the database of the system. It can only be accessed locally since the system is locally used. MySQL fits the requirements of the system for storing data.
- **PHP** - This is a server-side scripting language. With PHP, the system can store relevant data in MySQL and SQLite.
- **PlatformIO** - contains additional features than the Arduino IDE, capabilities that make writing and debugging code much more straightforward.
- **Visual Studio Code** - is an IDE and code editor that supports various programming languages, including C++, Java, JavaScript, Python, and more.

Ⅷ. Testing Procedures

The proponents will design and develop a prototype to detect any farm environment changes. The sensors will be picked and put on the farm based on the owner's advice and the farm's demands. Following the system's installation, a web-based monitor will be set up to display the sensor data. For this research, the proponents will use different sensors to gather data. The raw data accumulated from the sensors will be processed in the microcontroller. The data will be collected daily, with 35 seconds intervals between collections.

A series of tests are conducted to assess the system's accuracy, functionality, and reliability. The proponents compared the sensor's accuracy with different sensors or testers already available in the market using a percentage error for continuous values and a confusion matrix for discrete values to determine the sensor's accuracy. Furthermore, to assess the system's reliability, the proponents continuously tested the system; using a confusion matrix, the proponents can determine how reliable the system is. Testing the system's functionality, the proponents have surveyed the farmers on Miguela's farm to gather feedback on the system's performance in meeting their specific needs and requirements. This allowed the proponents to evaluate the system's functionality and identify any areas for improvement or additional features that could enhance its usefulness to farmers.

Lastly, Mr. Baeza, the proponent's beneficiary, will test the overall accuracy, functionality, and reliability and validate the system after implementing the system in Miguela's Farm.

IX. Design Project Validators

Mr. Marvin Baeza is our design project validator. He earned his Bachelor's Degree in Philosophy in 2001 at Christ the King Seminary in Quezon City. The same year, he began as a Service Representative at Hytech Integrated Products. This company repaired and assembled computer products, rendering inventory of computer parts and rendering service in some companies. In 2004, he began to work at a Local Government Unit in Irosin, Sorsogon. He was installing LAN and Internet services within the vicinity of Irosin. He then owned and operated a computer shop. He became a Computer

Technician in 2006 and is responsible for installing and upgrading all Shell Gasoline Station servers. He became a Field Engineer at Fourcort Unlimited, responsible for installing and maintaining Electronic Data Capture in all Shell Gasoline stations. He then owned a general merchandise shop that operated a soft drinks delivery. In March 2012, he became an Application Specialist responsible for rendering service and support for POS systems and POS Applications in SM, Ayala, Robinsons, Mega World, and other client companies. He then became an Operations Manager, responsible for daily sales management and achievement. He was promoted to a Team Lead/Station Supervisor that leads and motivates the crew. He formerly worked as an oil forecaster for the Philippines Gasoline and Diesel Price Update. He is presently the President of Katanim Hydroponics Grower, a local farmer association in the Philippines. He is an instructor who frequently travels to teach others how to create their hydroponics garden. He aims to promote hydroponics and create additional greenhouses, providing jobs and ensuring long-term farming. Through this advocacy, he hopes to educate others, particularly farmers, about alternative and effective agricultural methods that will enhance the lives of many. His curriculum vitae is attached in appendix a to support his claim.

Ⅹ. Design Project Evaluation Instrument

The proponents will utilize a survey questionnaire as a research tool to evaluate the system's accuracy, functionality, and reliability after testing it on research subjects. The proponents will use the Likert scale to evaluate the system. It is shown in table 1. The survey questionnaire is attached in appendix b. In the survey questionnaire, the first two questions will cover accuracy, followed by questions on functionality for questions three (3) to six (6). The last four (4) questions will cater to the overall system's reliability.

Table 2: Likert Scale.

Interpretation	Rating
Strongly Disagree	1
Disagree	2
Neutral	3
Agree	4
Strongly Agree	5

XI. Statistical Treatment of Data

Testing the accuracy of the sensors, the proponents used percentage error for continuous values and confusion matrix for discrete values. Testing the system's reliability, the proponents also used a confusion matrix. The proponents also used mean to describe the overall accuracy of the sensors.

A confusion matrix is a tabular representation that shows the performance of a sensor or the system by summarizing the expected and actual class labels for a set of instances. Determining the accuracy and reliability using confusion matrix (formula):

$$\frac{True\ Positive + True\ Negative}{True\ Positve + True\ Negative + False\ Positve + False\ Negative} \times 100$$

The percentage error measures the relative error between the expected and actual values. A lower percentage error indicates higher accuracy, while a higher percentage error suggests lower accuracy. Determining the accuracy of the sensor using percentage error (formula):

$$\frac{Observe\ Value - True\ Value}{True\ Value} \times 100$$

Mean also known as the average of the numbers - is determined by dividing the sum of the given numbers by the total number of digits.

$$Mean = \frac{\Sigma fx}{n}$$

f = frequency in each class.
x = midpoint of each class.
n = total number of scores.

Chapter IV

Results and Discussion

The proponents continuously tested the accuracy of the sensors and the system 30 times each; this determines how reliable the system is in growing lettuce. As mentioned in Chapter 3, the proponents used percentage error and confusion matrix to gather how accurate the sensors are and how reliable the system is. All numerical values will then be consolidated into tables located in the appendices. The said numerical values will be compared to the standard (ideal) values from table 2 in chapter 2.

Table 3: Average Accuracy of the 6 Sensors of the System.

	Average Sensor Accuracy	**Average Third-Party Sensor Accuracy**	**Average Percentage Error**
Carbon Dioxide	452.13 PPM	489.63 PPM	9.56%
Humidity	64.57%	49.13%	31.44%
Temperature	31.93℃	33.5 ℃	5.02%
Light Sensor	1522.23 Lux	1597.67 Lux	8.59%
Soil Moisture	53.8%	47.74 %	12.84%

The carbon dioxide sensor shows varying values and an average of 452.13 PPM from the system but never exceeds 500 PPM. The third-party sensor shares a similar value with the occasional outliers and an average value of 489.63 PPM but does not exceed 600 PPM, below the standard values of 800 PPM to 1200 PPM required for lettuce growth. The percentage error ranges from a minimum of 0.42% to a maximum of 24.82% and averages 9.56%. Testing of the sensor was accomplished by allowing the sensor to measure the carbon dioxide emissions from brunt newspaper and would activate the corresponding actuator to disperse the carbon dioxide gas. According to an article written by Florida Atlantic University (2023), sources of carbon dioxide appear naturally in the atmosphere due to the decay of organisms, volcano eruptions, forest fires, etc. When hydrocarbon fuel (Wood, oil, fossil fuel) is burned, carbon dioxide and water vapors is emitted when the carbon from the fuel interacts with the surrounding oxygen. The system provided a low variability of humidity values within 62% to 69% and averaged at 64.57%, while the third-party sensor provided similar groupings ranging from 46% to 53% and 49.13%. The values provide the proponents with percentage values ranging from 26% to a maximum of 38% and averaging 31.44%. The values show that the humidity is consistent throughout testing and falls within the standard range of 40% to 80%. The system presented low variability of temperature values from the temperature sensor. From a minimum of 30℃ to a maximum of 34℃ and an average of 31.93℃, while the third-party sensor shows similar readings ranging from 31℃ to 35℃ and an average of 33.5℃. All are within the standard growing range of 18C to 26C for lettuce to grow. The table shows that the system can provide the user with temperature readings with a small margin of error, only reaching an error percentage of 11.76% and an average percentage error of 5.02%. The system's light sensor shows a high variability of Lux values. The third-party sensor also shares the same variability of values and sometimes shows a percentage error of 0% to a maximum of 35%. The average Lux value of the system, 1522.23 Lux, and third party sensor, 1597.67 Lux, are on the higher end of the spectrum while the percentage error averages at a reasonable 8.59%. Both values go beyond the standard range of 282.60 Lux to 421.74 Lux, which is the ideal range to supply light to the lettuce. The soil moisture presents consistent and averages at 53.8% moisture value, while the third-party sensor says otherwise. The third-party sensor shows a range of values from 45.13% to 49.59% and averaging at 47.74%, which provided low variability of distribution at a lower range compared to the range of the soil moisture sensor of the system, with both being on the lower end of the standard range of 50% to 80%. The percentage error ranges from 8.08% to 19.65% and averages 12.84%.

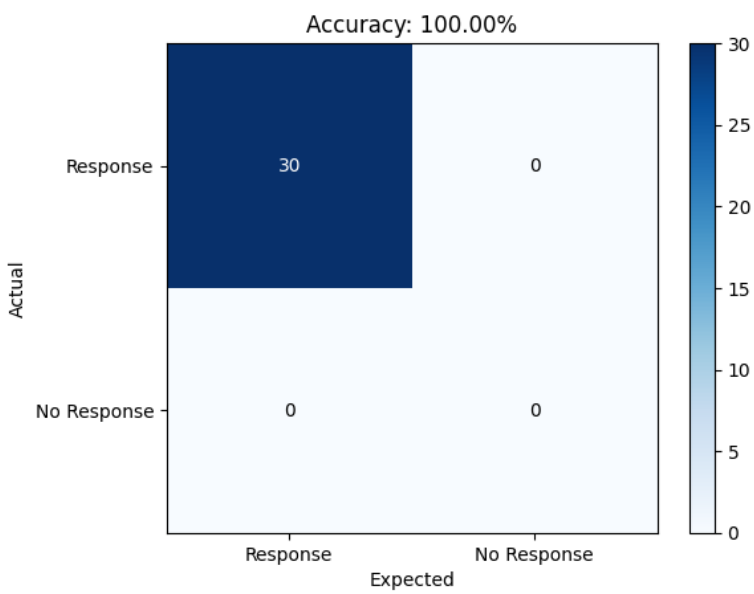

Figure 29: Accuracy of the Water Level Sensor.

In Figure 29, the testing of the water level sensor provided the proponents with 100% confidence in the accuracy of the water level sensor when activated by external stimuli.

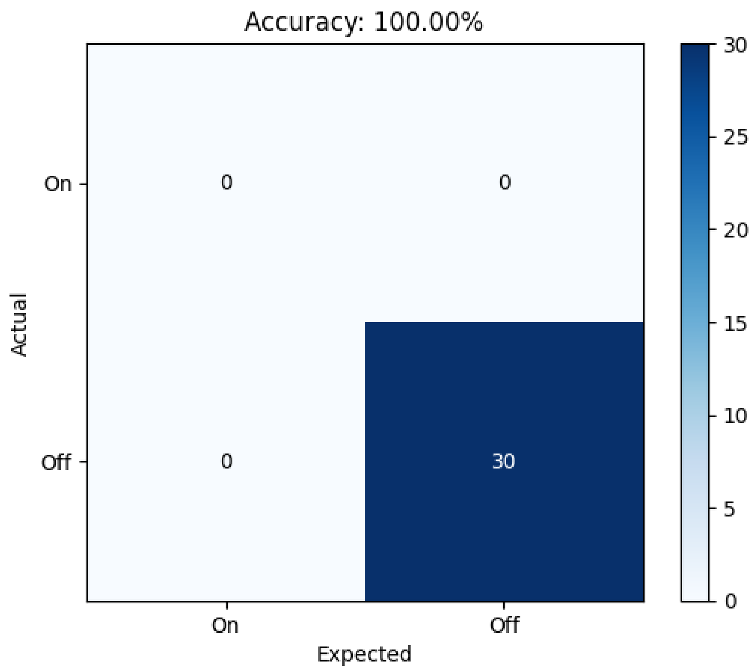

Figure 30: Confusion Matrix for the Actuator of the Carbon Dioxide Sensor

In Figure 30, the sensor-to-actuator response achieved an accuracy value of 100% during testing. It provided the user with a responsive actuator control to mitigate the undesired levels of carbon dioxide gas surrounding the lettuce.

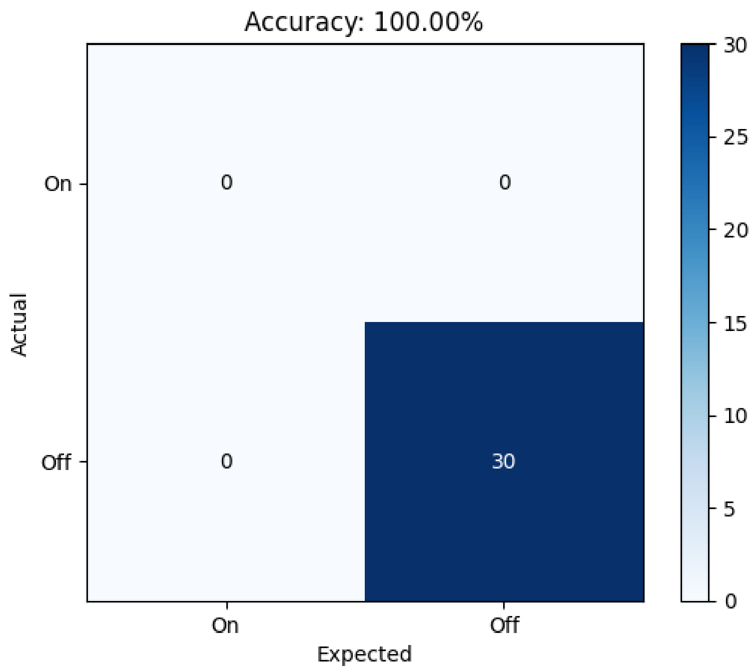

Figure 31: Confusion Matrix for the Actuator of the Humidity Sensor

In Figure 31, the response accuracy achieved during continuous testing by the actuator of the humidity sensor is 100% and proves the accuracy of the responsive actuator control. This will provide users with control of water management.

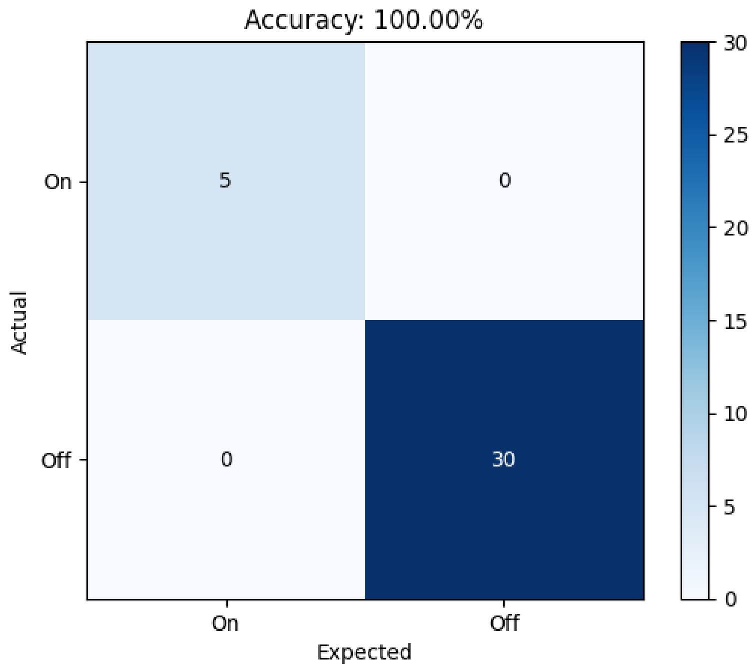

Figure 32: Confusion Matrix for the Actuator of the Light Sensor

In Figure 32, the system also achieved 100% in the response testing of the actuators of the light sensors and ensures that the artificial light source will be active when sunlight is not abundant or otherwise.

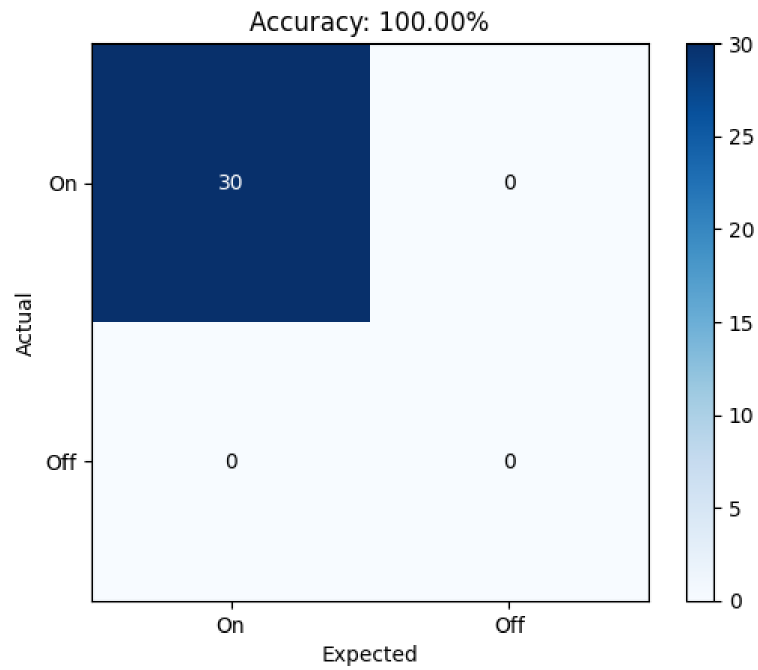

Figure 33: Confusion Matrix for the Actuator of the Soil Moisture

In Figure 33, the system acquired 100% in the accuracy of the actuator of the soil moisture sensor as the system provides the soil with a good water source at appropriate intervals to avoid drying out the lettuce soil. The system will ensure that the user uses efficient water management to minimize water waste.

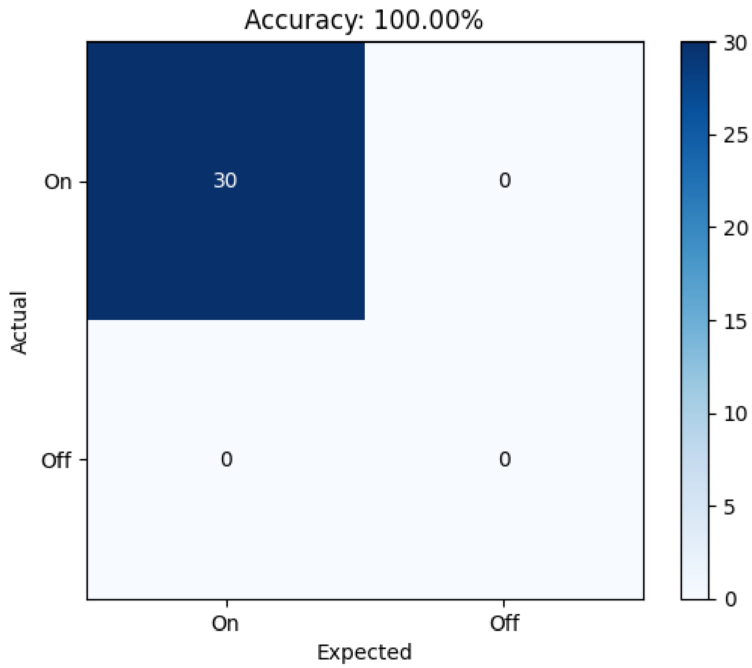

Figure 34: Confusion Matrix for the Actuator of the Temperature Sensor

In Figure 34, the actuator of the temperature sensor responds to the stimuli collected by the temperature sensor 100% of the time during the testing phase; this will allow users to fully manipulate the temperature surrounding the lettuce and efficiently manage electrical resources.

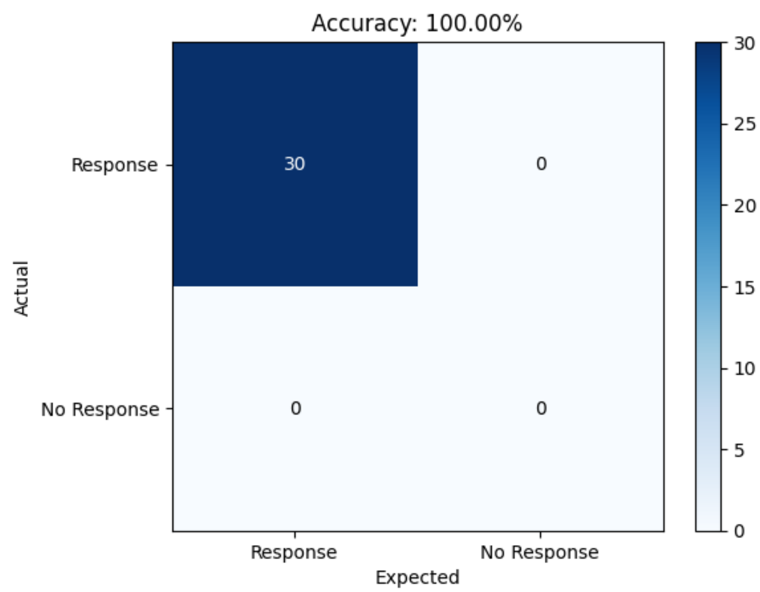

Figure 35: Confusion Matrix for the Actuator of the Water Level Sensor

In Figure 35, the actuator of the water level sensor responded accurately 100% of the time during testing. This goes hand-in-hand with activating the soil moisture sensor's actuator, reducing water waste, and providing better water management.

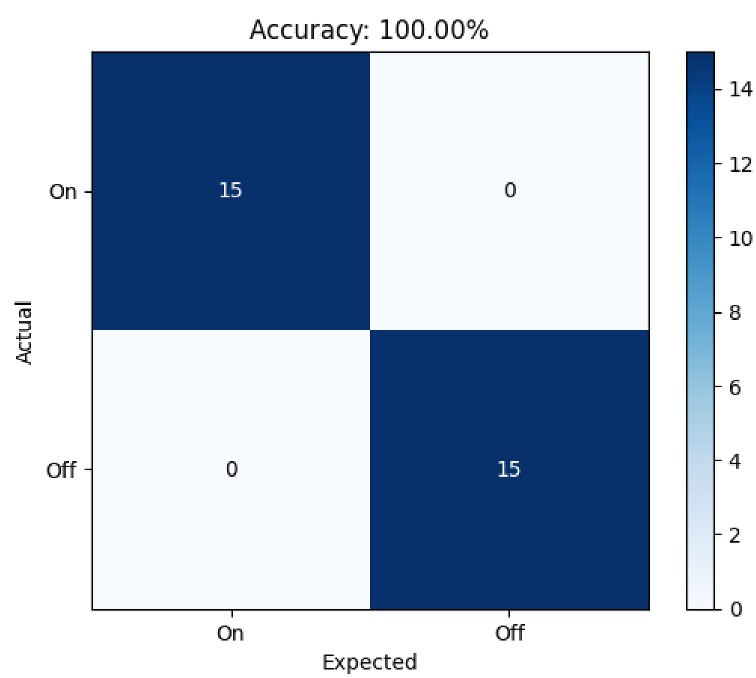

Figure 36: Reliability of the Bypass System.

In Figure 36, the bypass actuators system proves reliable in controlling the actuators and demonstrates exceptional performance with a remarkable 100% accuracy achieved during testing. This outstanding accuracy showcases the system's effectiveness and reinforces its reliability in accurately manipulating the actuators as intended. The flawless precision achieved by the bypass actuators system further solidifies its suitability for critical applications where precise control and consistent performance are of utmost importance.

Chapter V

Summary of Findings, Conclusions, and Recommendations

The proponents conducted extensive testing to evaluate the sensors' and system's accuracy and reliability for growing lettuce. They used various methods, such as percentage error and confusion matrix, to assess the accuracy of the sensors and compared the results to ideal values from previous research. The findings are summarized as follows:

1. Carbon Dioxide Sensor:

- The system's carbon dioxide sensor showed varying values, averaging 452.13 PPM, never exceeding 500 PPM.
- The third-party sensor had similar values, averaging 489.63 PPM and not exceeding 600 PPM.
- The percentage error ranged from 0.42% to 24.82%, with an average of 9.56%.

2. Humidity Sensor:

- The system provided consistent humidity values between 62% to 69%, averaging 64.57%.
- The third-party sensor also showed similar values, ranging from 46% to 53% and averaging 49.13%.
- The percentage error ranged from 26% to 38%, averaging 31.44%.

3. Temperature Sensor:

- The system presented low variability in temperature values, ranging from 30℃ to 34℃ and averaging 31.93℃.

- The third-party sensor exhibited similar readings, ranging from 31℃ to 35℃ and averaging 33.5℃.
- All temperature values fell within the standard range of 18°C to 26°C for lettuce growth.
- The percentage error ranged from 0% to 11.76%, averaging 5.02%.

4. Light Sensor:

- The system's light and third-party sensors showed high variability in Lux values.
- The average Lux value for the system was 1522.23 Lux, while the third-party sensor averaged 1597.67 Lux.
- The percentage error averaged 8.59%.

5. Soil Moisture Sensor:

- The system provided consistent soil moisture values, averaging 53.8%.
- The third-party sensor showed lower variability, ranging from 45.13% to 49.59% and averaging 47.74%.
- Both sensors fell on the lower end of the standard range of 50% to 80% moisture.
- The percentage error ranged from 8.08% to 19.65%, with an average of 12.84%.

6. Water Level Sensor:

- During testing, the actuator of the water level sensor consistently responded with 100% accuracy. This successful response aligns with activating the soil moisture sensor's actuator, reducing water waste, and improving water management.

In addition to sensor accuracy, the proponents also tested the responsiveness and accuracy of the system's actuators. The results are as follows:

- Water level sensor actuator: Achieved 100% accuracy in response testing.
- Humidity sensor actuator: Achieved 100% accuracy in response testing.
- Light sensor actuator: Achieved 100% accuracy in response testing.
- Soil moisture sensor actuator: Achieved 100% accuracy in response testing.
- Temperature sensor actuator: Achieved 100% accuracy in response testing.
- Bypass actuators system: Demonstrated exceptional performance with 100% accuracy in controlling the actuators.

Overall, the findings indicate that the system's sensors provide accurate readings within acceptable ranges for lettuce growth. The actuators exhibit precise and reliable control, ensuring efficient management of environmental factors for lettuce cultivation.

The IoT-based smart garden system implementing edge computing was successfully designed and developed, integrating multiple agricultural sensors and actuators through the Internet of Things and edge computing. This system enables near real-time monitoring of parameters such as temperature, humidity, soil moisture, light, carbon dioxide, and water level. Additionally, A user-friendly web-based dashboard was successfully developed to serve as a monitoring tool for the garden keepers. This dashboard provides an intuitive interface for monitoring the data collected by the sensors. Furthermore, the proponents have successfully tested the system's accuracy, functionality, and reliability through user-driven testing and found out that the system is dependable in growing lettuce plants.

Due to budget constraints, the system has only used the typical fan and failed to adjust the surrounding temperature of the environment. However, removing the budget constraint, the proponents suggest replacing the fan with an air conditioner to better impact the temperature surrounding the lettuce plants. Moreover, the proponents also recommend that the system is adequate in indoor planting, especially the temperature, where the environment is controlled. The system is not perfect and still needs to be improved to enhance further the system's performance; ongoing research and development should be conducted to identify areas for improvement and optimize the system's capabilities.

References

2018 Census of Philippine Business and Industry: Economy-Wide | Philippine Statistics Authority. (n.d.). Retrieved November 24, 2022, from https://psa.gov.ph/content/2018-census-philippine-business-and-industry-economy-wide

Abd El-Kader, S. M., & El-Basioni, B. M. M. (2020). Precision agriculture technologies for food security and sustainability. Precision Agriculture Technologies for Food Security and Sustainability, 1–437. https://doi.org/10.4018/978-1-7998-5000-7

Acurite. (2018, April). Soil Moisture Recommendations for Flowers, Plants, and Vegetables. https://www.acurite.com/blog/soil-moisture-guide-for-plants-and-vegetables.html

AI, Edge and IoT-based Smart Agriculture - Google Books. (n.d.). Retrieved November 27, 2022, from https://books.google.com.ph/books?id=RnINEAAAQBAJ&printsec=frontcover&dq=iot+in+agri&hl=en&sa=X&ved=2ahUKEwi0vaDsi-L6AhVPrlYBHTksC_IQ6AF6BAgIEAI#v=onepage&q=iot%20in%20agri&f=false

Ali, A., & Alshmrany, S. (2020). Internet of things (IoT) embedded smart sensors system for agriculture and farm management. International Journal of Advanced and Applied Sciences, 7(10), 38–45. https://doi.org/10.21833/ijaas.2020.10.005

Al-Omary, A., AlSabbagh, H. M., & Al-Rizzo, H. (2018). Cloud based IoT for Smart Garden Watering System using Arduino Uno. Smart Cities Symposium 2018, 33 (6 pp.)-33 (6 pp.). https://doi.org/10.1049/CP.2018.140110.1049/CP.2018.1401

Amare, G., & Gebremedhin, H. (2020). Effect of Plant Spacing on Yield and Yield Components of Tomato (Solanum lycopersicum L.) in Shewarobit, Central Ethiopia. Scientifica, 2020. https://doi.org/10.1155/2020/8357237

Boter, M., Calleja-Cabrera, J., Carrera-Castaño, G., Wagner, G., Hatzig, S. V., Snowdon, R. J., Legoahec, L., Bianchetti, G., Bouchereau, A., Nesi, N., Pernas, M., & Oñate-Sánchez, L. (2019). An Integrative Approach to Analyze Seed Germination in Brassica napus. Frontiers in Plant Science, 10, 1342. https://doi.org/10.3389/FPLS.2019.01342/BIBTEX

Cao, K., Liu, Y., Meng, G., & Sun, Q. (2020). An Overview on Edge Computing Research. IEEE Access, 8, 85714–85728. https://doi.org/10.1109/ACCESS.2020.2991734

Chen, B., Wan, J., Celesti, A., Li, D., Abbas, H., & Zhang, Q. (2018). Edge Computing in IoT-Based Manufacturing. IEEE Communications Magazine, 56(9), 103–109. https://doi.org/10.1109/MCOM.2018.1701231

Crop production manual: A guide to fruit and vegetable production in the ... - Food and Agriculture Organization of the United Nations - Google Books. (n.d.). Retrieved November 27, 2022, from https://books.google.com.ph/books?id=IhjRDwAAQBAJ&printsec=frontcover&dq=growing+a+lettuce+in+lowland&hl=en&sa=X&ved=2ahUKEwi-rI-S2fL6AhWPmVYBHdRCAPg4ChDoAXoECAMQAg#v=onepage&q=lowland&f=false

Crop Rotations - Rodale Institute. (n.d.). Retrieved November 27, 2022, from https://rodaleinstitute.org/why-organic/organic-farming-practices/crop-rotations/

Dr. Melissa Brechner, & Dr. A.J. Both. (2020). Cornell CEA Hydroponic Lettuce Handbook - Cornell Controlled Environment Agriculture Hydroponic - Studocu. https://www.studocu.com/pe/document/universidad-cesar-vallejo/conocimientos-especificos-2/cornell-cea-hydroponic-lettuce-handbook/24511197

Edge Computing Paradigm | OpenMind. (n.d.). Retrieved November 27, 2022, from https://www.bbvaopenmind.com/en/technology/digital-world/edge-computing-paradigm/

Farooq, M. S., Riaz, S., Abid, A., Abid, K., & Naeem, M. A. (2019). A Survey on the Role of IoT in Agriculture for the Implementation of Smart Farming. In IEEE Access (Vol. 7, pp. 156237–156271). Institute of Electrical and Electronics Engineers Inc. https://doi.org/10.1109/ACCESS.2019.2949703

flores, david riquelme. (2018). Eloquent JavaScript 3rd edition. Eloquent 3 Edicion. https://www.academia.edu/40513205/Eloquent_JavaScript_3rd_edition

Florida Atlantic University. (2023, March 4). Causes of climate change. Climate Science Investigations South Florida - Causes of Climate Change. https://www.ces.fau.edu/nasa/module-4/causes/sources-carbon-dioxide.php

Geng, X., & Chun, O. (2022). Multiscale Layout and Planning of Smart Gardens in the Environment of Agricultural Internet of Things. Computational Intelligence and Neuroscience, 2022. https://doi.org/10.1155/2022/4746629

Gonzaga, N., Pepito, S. L., Octavio, R., Gonzaga, A., & Rogers, G. (2017). Growth and yield performance of lettuce (Lactuca sativa Linn) under protected and conventional cultivation. Annals of Tropical Research, 137–143. https://doi.org/10.32945/atr39sb11.2017

Growing Vegetables: KnolKhol, Lettuce and Zucchini - Roby Jose Ciju - Google Books. (n.d.). Retrieved November 27, 2022, from https://books.google.com.ph/books?id=c0GZDwAAQBAJ&printsec=frontcover&dq=lettuce+growing&hl=en&sa=X&ved=2ahUKEwiQobKiv-b6AhXNEnAKHa_lC4oQ6AF6BAgLEAI#v=onepage&q=lettuce%20growing&f=false

Gultom, G., Rahmansyah, A. A., Situngkir, D., & Ginting, M. (2021). Temperature Control System for Heating-Holding Heater Based Arduino With Monitoring Via Webserver. JOURNAL OF INFORMATICS AND TELECOMMUNICATION ENGINEERING, 5(1), 84–91. https://doi.org/10.31289/JITE.V5I1.5253

Hasanuzzaman, M. (n.d.). Calculation of Seed Rate Study on the Calculation of Seed Rates. Retrieved November 27, 2022, from www.hasanuzzaman.webs.com

Internet of Things (IoT) for Automated and Smart Applications - Google Books. (n.d.). Retrieved November 27, 2022, from https://books.google.com.ph/books?id=O0P8DwAAQBAJ&printsec=frontcover&dq=what+is+iot

&hl=en&sa=X&ved=2ahUKEwi78am6j-L6AhXbcGwGHek2C3AQ6AF6BAgLEAI#v=onepage &q=what%20is%20iot&f=false

Kar, A. (n.d.). Transformation Philippine Agriculture (P172579).

Lettuce Production Guide. (n.d.).

Mahbub, M. (2020). A smart farming concept based on smart embedded electronics, internet of things and wireless sensor network. Internet of Things, 9, 100161. https://doi.org/10.1016/J.IOT.2020.100161

Mapa, D. S., Duran, M. C. G., Nalugon, M. S., Clarinda, M., de Guzman, E., Rose, D., Sigua, A., & Villan, M. C. (2020). PHILIPPINE STATISTICS AUTHORITY EMPLOYMENT AND WAGES IN THE AGRICULTURE SECTOR PHILIPPINE STATISTICS AUTHORITY Chief Statistical Specialist.

Mary Grace, S., Agulto, I., Espino, A., & Sicat, E. (2018). Indoor production of loose-leaf lettuce (Lactuca sativa L.) using artificial lights and cooling system in tropical lowland. MATEC Web of Conferences, 192, 03016. https://doi.org/10.1051/MATECCONF/201819203016

Mleke, M. N., & Dida, M. A. (2020). A Web-based Monitoring and Evaluation System for Government Projects in Tanzania: The Case of Ministry of Health. Engineering, Technology & Applied Science Research, 10(4), 6109–6115. https://doi.org/10.48084/ETASR.3435

Navulur, S., Sastry, A. S. C. S., & Giri Prasad, M. N. (2017). Agricultural Management through Wireless Sensors and Internet of Things. International Journal of Electrical and Computer Engineering (IJECE), 7(6), 3492–3499. https://doi.org/10.11591/IJECE.V7I6.PP3492-3499

Nixon, R. (n.d.). Learning PHP, MySQL & JavaScript.

Paz, M., Fisher, P. B., & Gómez, C. (2019). Minimum Light Requirements for Indoor Gardening of Lettuce. *Urban Agriculture & Regional Food Systems*, *4*(1), 1–10. https://doi.org/10.2134/urbanag2019.03.0001

Saravanan M., P. M. K. N. K. (2021). Internet of Things Based Intelligent System for Monitoring and Improving the Crop Growth Rate Using Deep Learning Technique. Annals of the Romanian Society for Cell Biology, 25, 7877–7889. https://www.annalsofrscb.ro/index.php/journal/article/view/2329

Serhan, D. (2019). Web-Based Homework Systems: Students' Perceptions of Course Interaction and Learning in Mathematics. International Journal on Social and Education Sciences, 1(2).

Singh, H., Poudel, M. R., Dunn, B. L., Fontanier, C., & Kakani, G. (2020). Effect of Greenhouse CO2 Supplementation on Yield and Mineral Element Concentrations of Leafy Greens Grown Using Nutrient Film Technique. Agronomy 2020, Vol. 10, Page 323, 10(3), 323. https://doi.org/10.3390/AGRONOMY10030323

Thamaraimanalan, T., Vivekk, S. P., Satheeshkumar, G., & Saravanan, P. (n.d.). Smart Garden Monitoring System Using IOT. Asian Journal of Applied Science and Technology (AJAST) (Open Access Quarterly International Journal, 2(2), 186–192. Retrieved November 27, 2022, from www.ajast.net

Tagle, S., Benoza, H., Pena, R., & Oblea, A. (2018). 6 the DLSU Innovation and Technology Fair.

The Art and Science of Agriculture | National Geographic Society. (n.d.). Retrieved November 27, 2022, from https://education.nationalgeographic.org/resource/agriculture

Thesis, M. S., & Alsaadi, S. A. (2018). TPS: Developing a web based homework monitoring system (teacher, parents, student) for Iraqi schools. http://openaccess.altinbas.edu.tr/xmlui/handle/20.500.12939/1563

Towards the Internet of Things: Architectures, Security, and Applications - Mohammad Ali Jabraeil Jamali, Bahareh Bahrami, Arash Heidari, Parisa Allahverdizadeh, Farhad Norouzi - Google Books. (n.d.). Retrieved November 27, 2022, from https://books.google.com.ph/books?id=7jidDwAAQBAJ&printsec=frontcover&dq=iot+architecture&hl=en&sa=X&redir_esc=y#v=onepage&q=iot%20architecture&f=false

Tsani, Y. R., & Subardono, A. (2019). Data Logging Implementation on Web-Based Communication on Arduino Devices. Jurnal Online Informatika, 3(2), 103–109. https://doi.org/10.15575/JOIN.V3I2.252

View of Digital Garden System Using Arduino. (n.d.). Retrieved November 27, 2022, from https://asrjetsjournal.org/index.php/American_Scientific_Journal/article/view/7554/2542

Water and agriculture - OECD. (n.d.). Retrieved November 24, 2022, from https://www.oecd.org/agriculture/topics/water-and-agriculture/

What Is Edge Computing & Why Is It Important? | Accenture. (n.d.). Retrieved November 27, 2022, from https://www.accenture.com/ae-en/insights/cloud/edge-computing-index

What Is the Internet of Things (IoT)? | Oracle Philippines. (n.d.). Retrieved November 27, 2022, from https://www.oracle.com/ph/internet-of-things/what-is-iot/

World population projected to reach 9.8 billion in 2050, and 11.2 billion in 2100 | UN DESA | United Nations Department of Economic and Social Affairs. (n.d.). Retrieved November 27, 2022, from https://www.un.org/development/desa/en/news/population/world-population-prospects-2017.html

Xun, Y., & Ren, G. (2022). Smart Garden Planning and Design Based on the Agricultural Internet of Things. Computational Intelligence and Neuroscience, 2022. https://doi.org/10.1155/2022/8522751

Zhang, X., Cao, Z., & Dong, W. (n.d.). Overview of Edge Computing in the Agricultural Internet of Things: Key Technologies, Applications, Challenges. https://doi.org/10.1109/ACCESS.2020.3013005

Zhou, J., Li, P. P., Wang, J. Z., & Fu, W. (2019). Growth, Photosynthesis, and Nutrient Uptake at Different Light Intensities and Temperatures in Lettuce. HortScience, 54(11), 1925–1933. https://doi.org/10.21273/HORTSCI14161-19

Appendix A: Curriculum Vitae of the Beneficiary

Marvin C. Baeza
#12 Chrysanthemum St., Modesta Village, San Mateo Rizal
Mobile +639772432914
Email: marvinbaeza@gmail.com

ARDS Enterprises (September 11, 2014 – May 28, 2018)
Station Supervisor (Team Lead)(Phoenix Mindanao 1)
Leads and motivates the Forecourt Crew to provide excellent customer service in order to up-sell and ensure the cleanliness and safety of the station, equipment and parts. Responsible in performing administrative duties that improves operational effectiveness. Recruits Purchase Order clients in order to boost product volume both fuels and lubricants. Forecast upcoming deliveries and its stocking in connection with the price increase or decrease for the coming weeks.

ALPSO Trading Corporation (June 2012 – August 2014)
Operations Manager (Phoenix Lubao Pampanga)
Responsible for the day-to-day management and achievement of the daily sales, including its customer service, policies and reports, ordering and receiving products, and the availability of the merchandise display. Responsible for the deposit of cash sales and collecting checks from the Purchase Order clients and for the payroll of the employees.

Pacific Office Machines (March 2012 – June 2012)
Application Specialist
Pre-Sales support of POS, Document Management Systems (DMS) and other IT related products. Responsible for rendering service and support for POS System and POS applications in SM, Ayala, Robinsons, Mega World, St. Luke's Medical Center and other client or companies.

Baeza General Merchandise (March 2008 – February 2012)
Operated Soft Drinks Delivery/Free Lance Computer Technician
Responsible for accepting order calls and delivery of soft drinks. Responsible for stocks forecasting and stocks booking for bulk deliveries. Repair and rehab of computers.

Fourcort Unlimited (August 2007 – February 2008)
Position: Field Engineer
Responsible for the installation and maintenance of Electronic Data Capture (EDC) in all Shell Gasoline Station.

Datalogic Systems Corporation (September 2006 – December 2006)
Position: Computer Technician
Responsible for the installation and upgrading of Base POS server and Base POS client in all Shell Gasoline Station around the Philippines. In charge for the installation of General Packet Radio Services in all Shell Station and upgrading the MSDE or MS SQL on every sites. Repairing and upgrading computers on site.

Baeza Enterprises (December 2003– July 2006)
Position: Owned and Operated Computer Shop
Responsible for the installation and upgrading of computers. Maintenance of Network and Internet Connection. Accepting computer repair and rehabilitation.

Local Government Unit – Irosin (January 2004– June 2004)
Position: Computer Technician
LAN and internet installation on Public offices within the vicinity of Irosin as a project of LGU Irosin in cooperation with PH Domain. Checking and maintaining internet connection on this site.

Hytech Intergreted Products (June 2001 – October 2002)
Position: Service Representative
A field of repair and assemble of computer products. Rendering inventory of computer parts from Taiwan and China. Rendering service in some company by installing, repair, assemble, and LAN installation.

Skills
Knows in forecasting pricing of Petroleum products if it will decrease or increase. Knows in forecasting deliveries of fuels and its stocking. Knows about database installation, database upgrade and database applications, like SQL and MS SQL. Knows how to install software, troubleshoot computer and computer parts assembly. I do computer maintenance, computer rehabilitation and upgrade. I know how to install network such as LAN and WAN.

Personal Particular
Age: 38
Date of Birth: October 16, 1980
Nationality: Filipino
Gender: Male
Marital Status: Married

Educational Background
College
Divine Word Mission Seminary
(Christ the King Seminary)
E.Rodriguez Sr. Blvd., Quezon City, Philippines
Bachelors Degree
Bachelor of Arts in Philosophy
1997-2001

Appendix B: Survey Questionnaire

UNIVERSITY OF THE EAST
Manila Campus
COLLEGE OF ENGINEERING
DEPARTMENT OF COMPUTER ENGINEERING

DESIGN PROJECT SURVEY QUESTION
Alpuerto, D.M., Cortez, J.C., Ibrahim, O., Tan, J.M.

Sir/Madam: Good day!

We are currently conducting a design project titled "Design and Development of IoT-Based Smart Garden Implementing Edge Computing" that will implement Internet of Things (IoT) and Edge Computing in an agricultural farm to help monitor and control plant production, particularly on Lettuce plant production for the partial fulfillment of our course CPE PRACTICE AND DESIGN. Your feedback will be of help to the proponents to decide on the future direction of this design project. Your answers will be highly appreciated.

Date: _______________

Legend: 5 – Strongly Agree, 4 – Agree, 3 – Neutral, 2 – Disagree, 1 – Strongly Disagree

Question	5	4	3	2	1
1. The actuators have accurate and precise movement.					
2. The application is capable of handling errors.					
3. The system can handle various processes at the same time.					
4. The system is better at producing lettuce than the traditional method of farming.					
5. The system produces real-time data coming from the sensors.					
6. The web-based monitoring system has an excellent user interface and user experience design.					
7. The actuator has a fast response time.					
8. The sensors have accurate and precise readings.					
9. The system can handle and store a large amount of data.					
10. The system performs the task required.					

Comments and suggestions:____

Appendix C: Expected Outputs

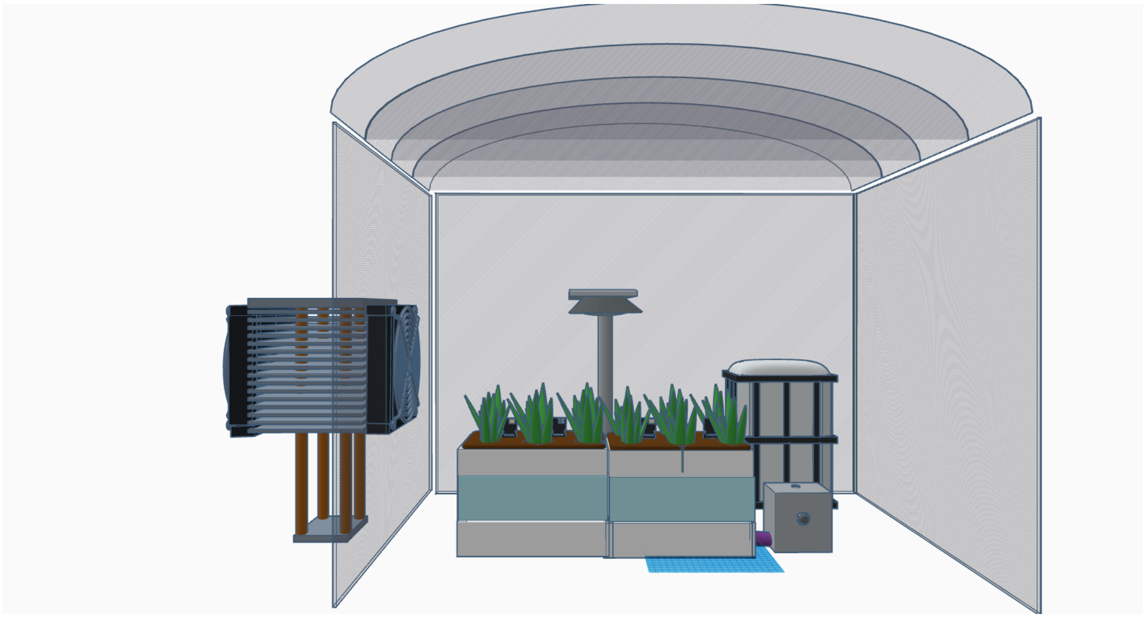

Figure 37: System Model (Front View).

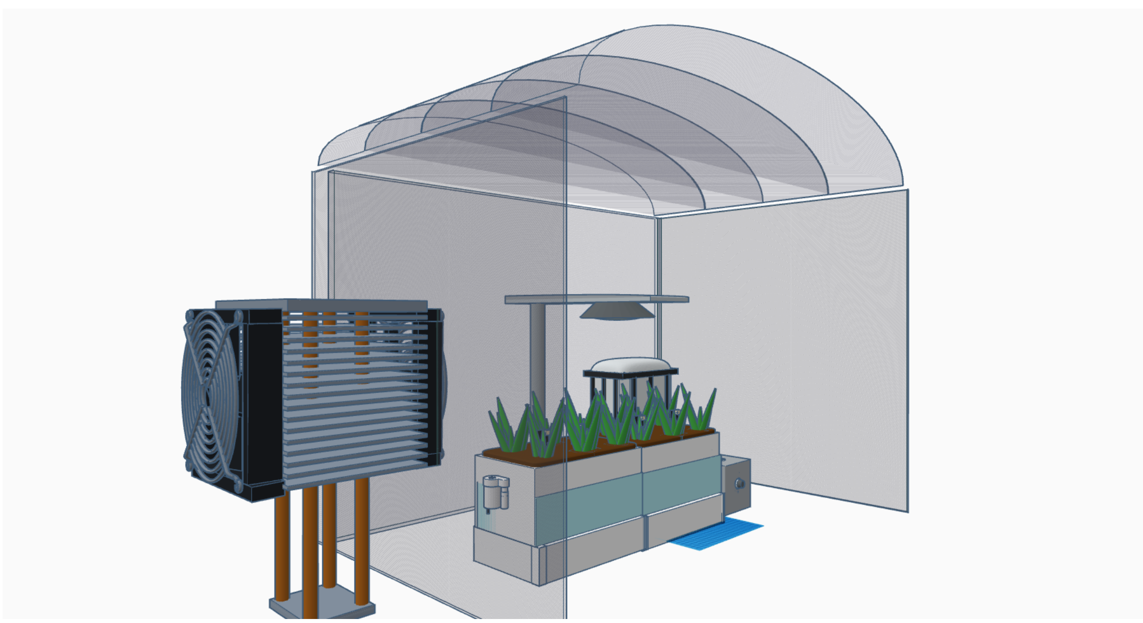

Figure 38: System Model (Oblique View).

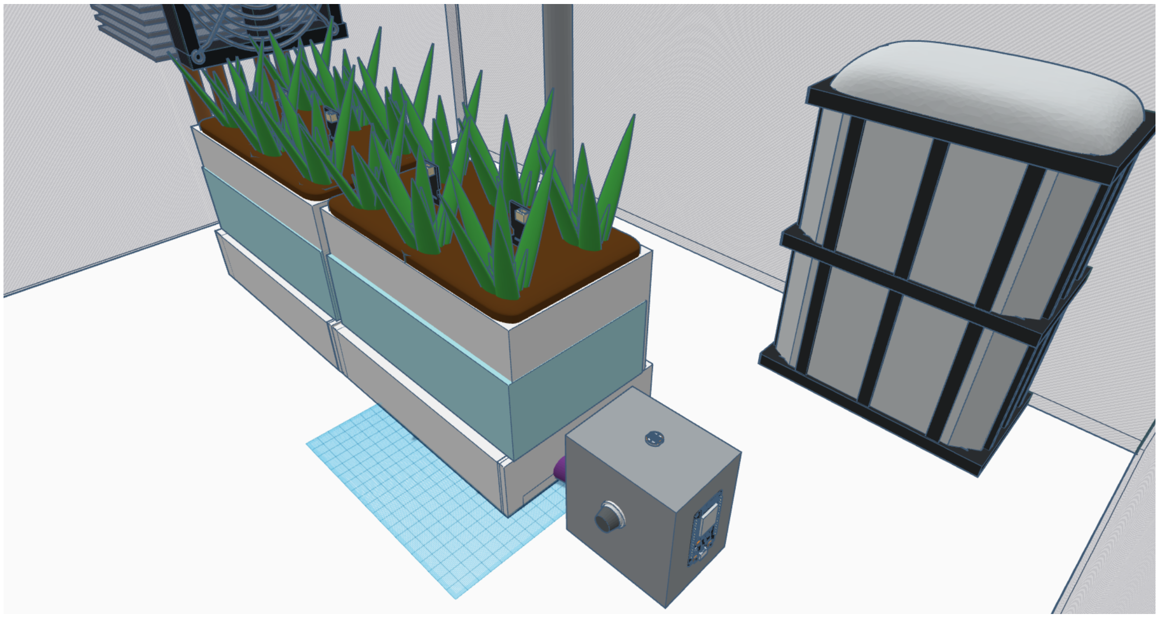

Figure 39: Sensor Casing (Top View).

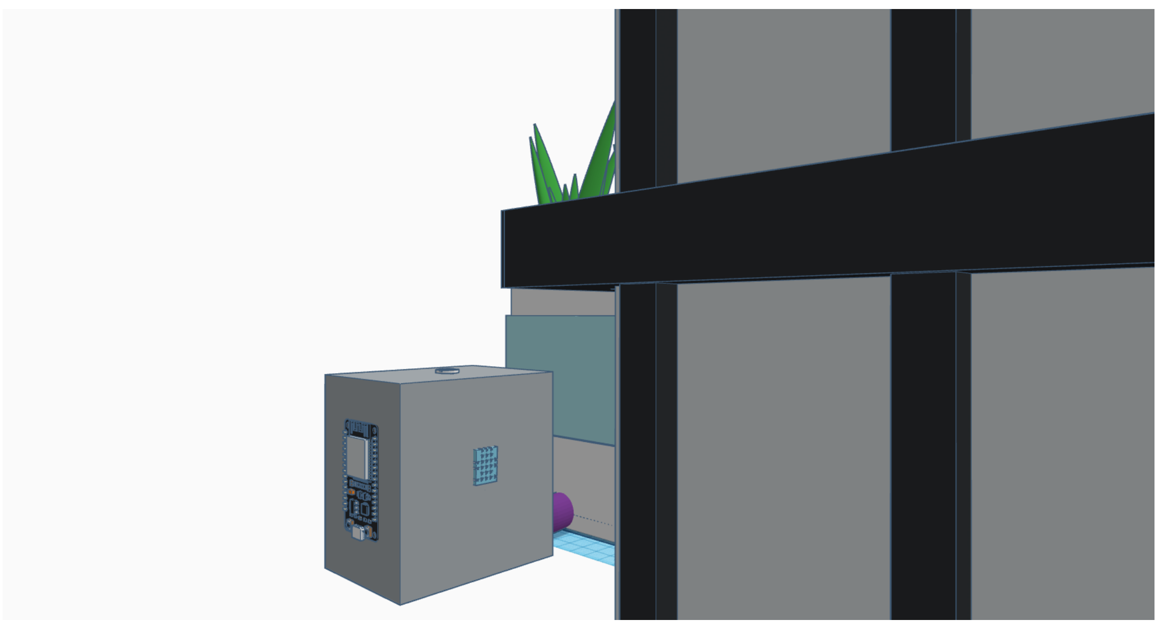

Figure 40: Sensor Casing (Side View).

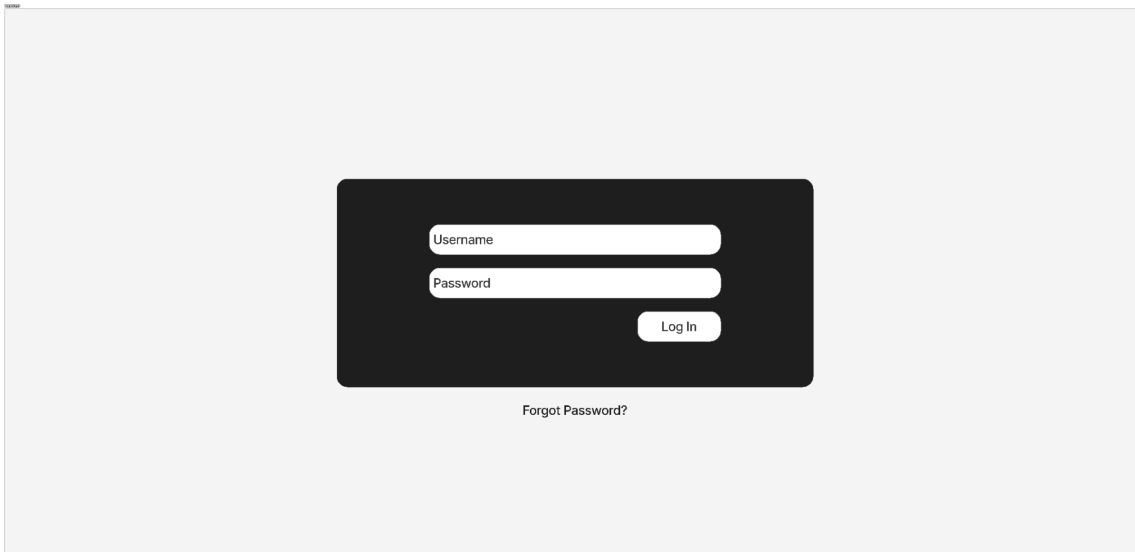

Figure 41: Log-In Page.

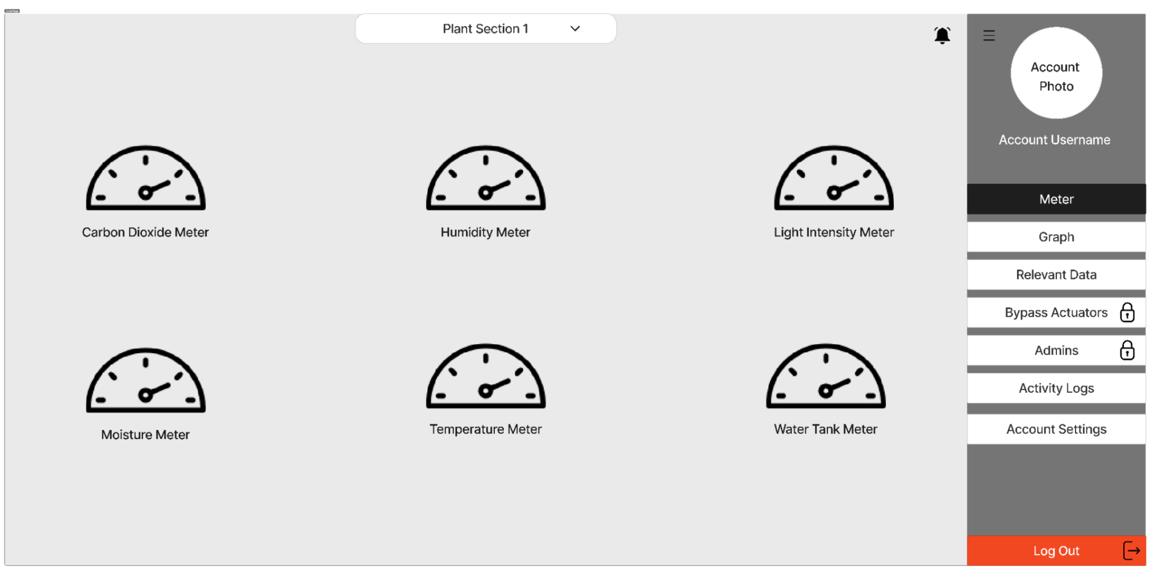

Figure 42: Meter Page.

Figure 43: Graph Page.

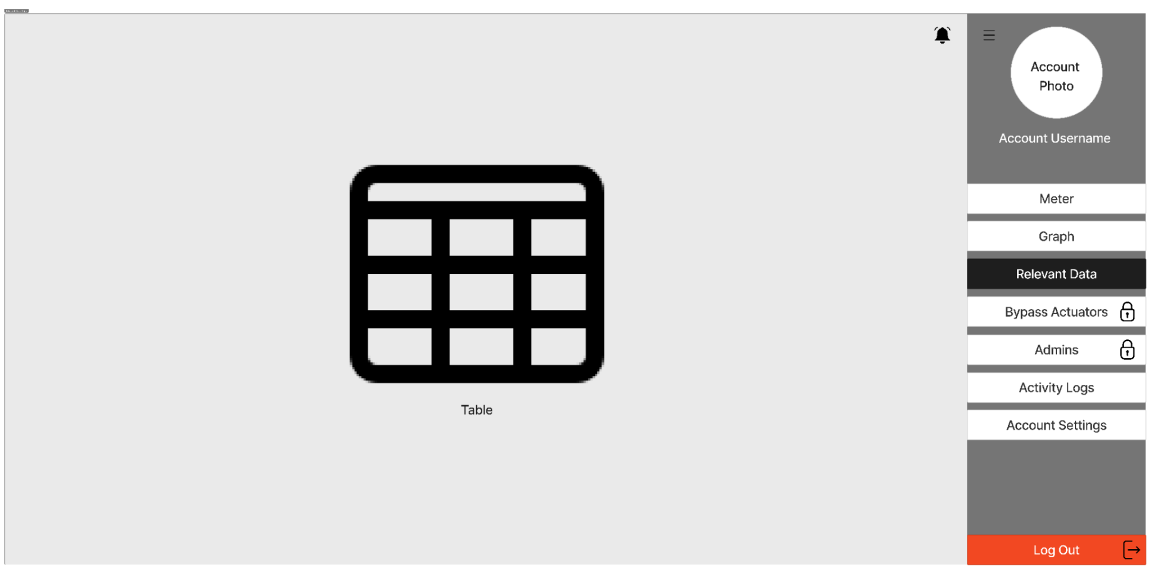

Figure 44: Relevant Data Page.

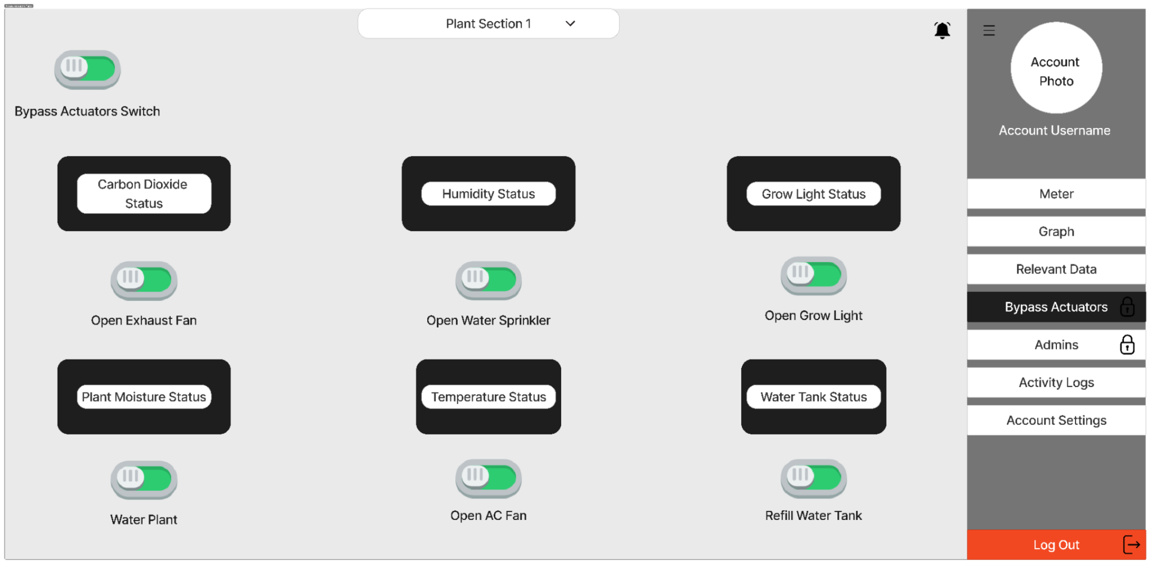

Figure 45: Bypass Actuators Page.

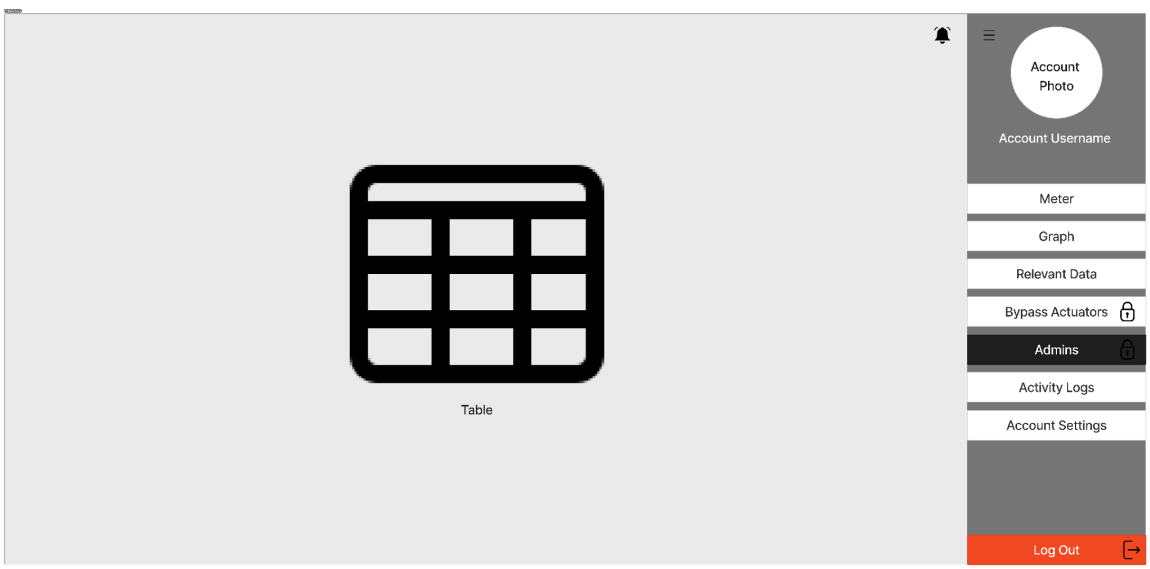

Figure 46: Admins Page.

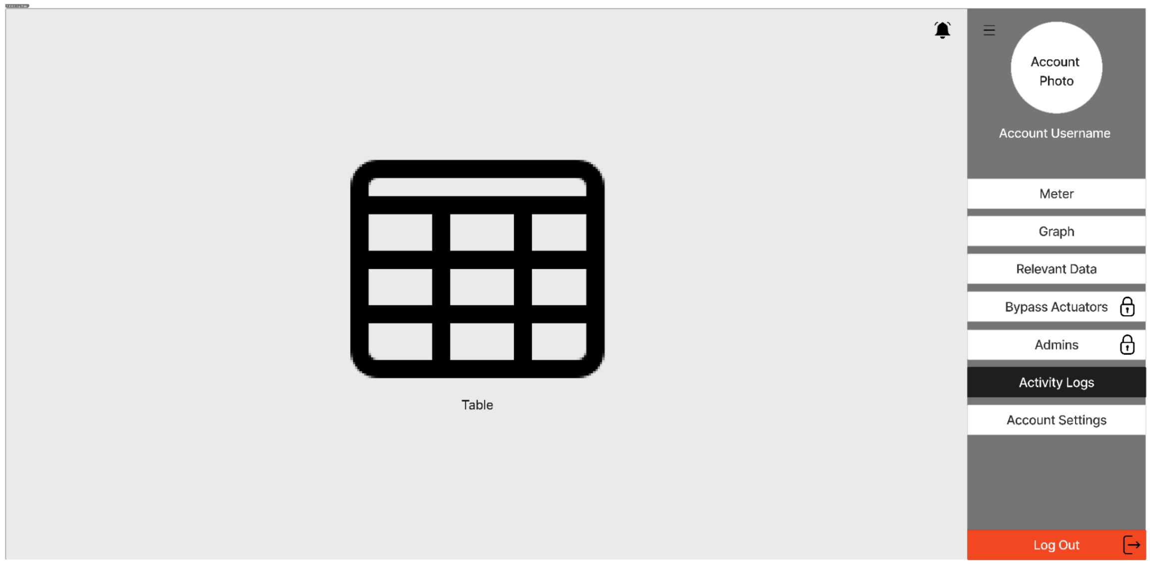

Figure 47: Activity Logs Page.

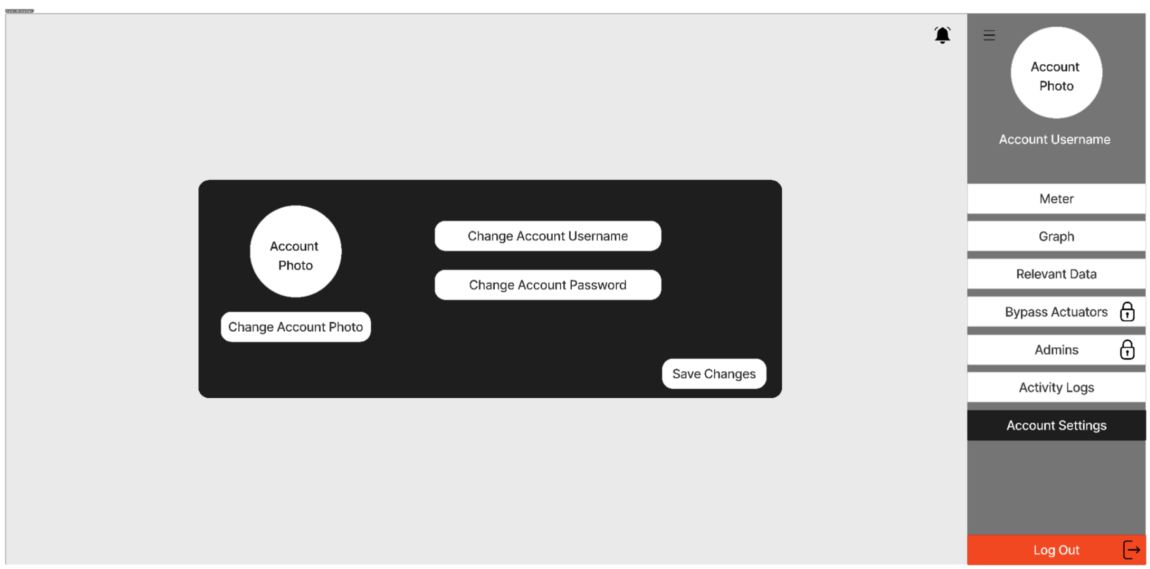

Figure 48: Account Settings Page.

In figure 41, The monitoring system will begin with a log-in page, requesting authentication from the end user or admin before accessing the system. Figures 42 and 43, the end-user can access near real-time data displayed in meters as the default main page with a tab available to access different pages after successfully logging in. A notification will also alert the end-user of new notices about the system; furthermore, the graph page will display the same near real-time data using line charts. Figure 44, the relevant data page will display the said data in a tabular form to provide the end-user with detailed information on any measurements outside the set conditions or threshold of the system with timestamps. Figure 45, The bypass actuators page will allow the end-user to access and control the actuators system manually. The page also includes a bypass actuator switch that acts as the master switch. In figures 46 and 47, the admins' page will add, update, delete, and display a table of end-users who can access the web-based monitoring system. The table will include several data or information about the admins of the system; moreover, the activity logs page will log and display any actions performed by the end-user and the system itself; it will include timestamps to indicate when the action has occurred. In figure 48, the account settings page will allow the end-user to modify their account; the end-user will be able to change the account username, password, and photo. Lastly, Before bypassing the actuators or accessing the admins' page, the system will ask for another authentication for a more secure system.

Appendix D: Actual Outputs

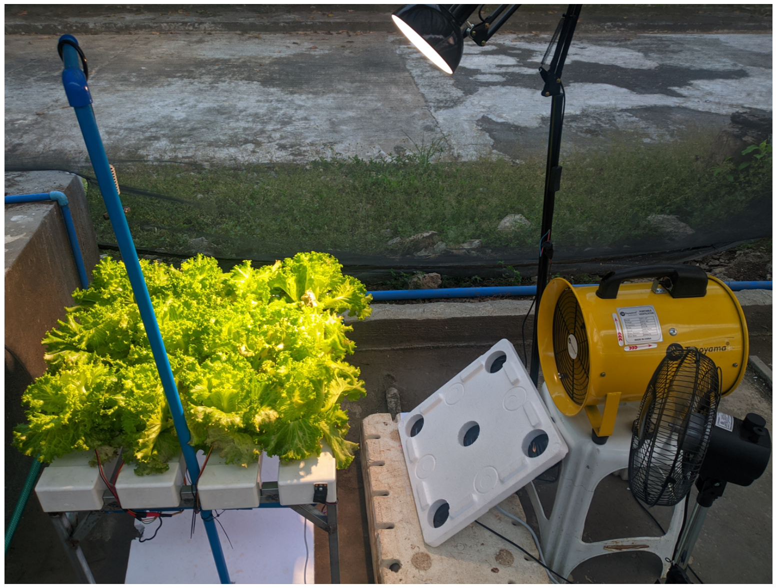

Figure 49. Prototype View 1.

Figure 50. Prototype View 2.

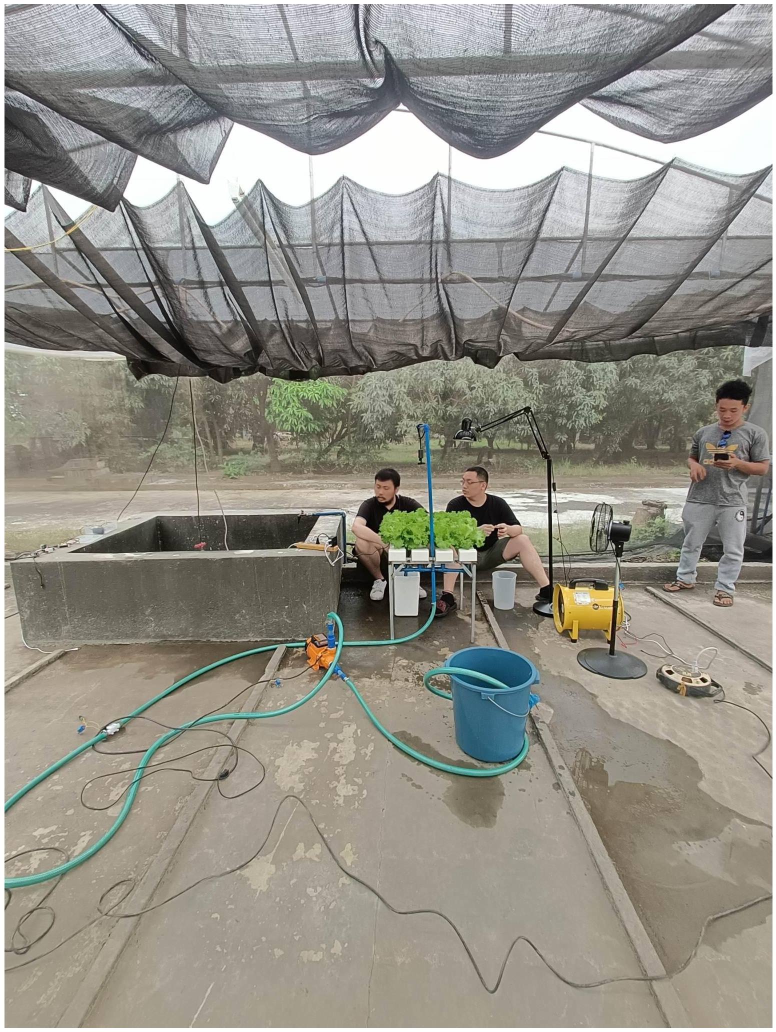
Figure 51. Prototype View 3.

Figure 52. Prototype View 4.

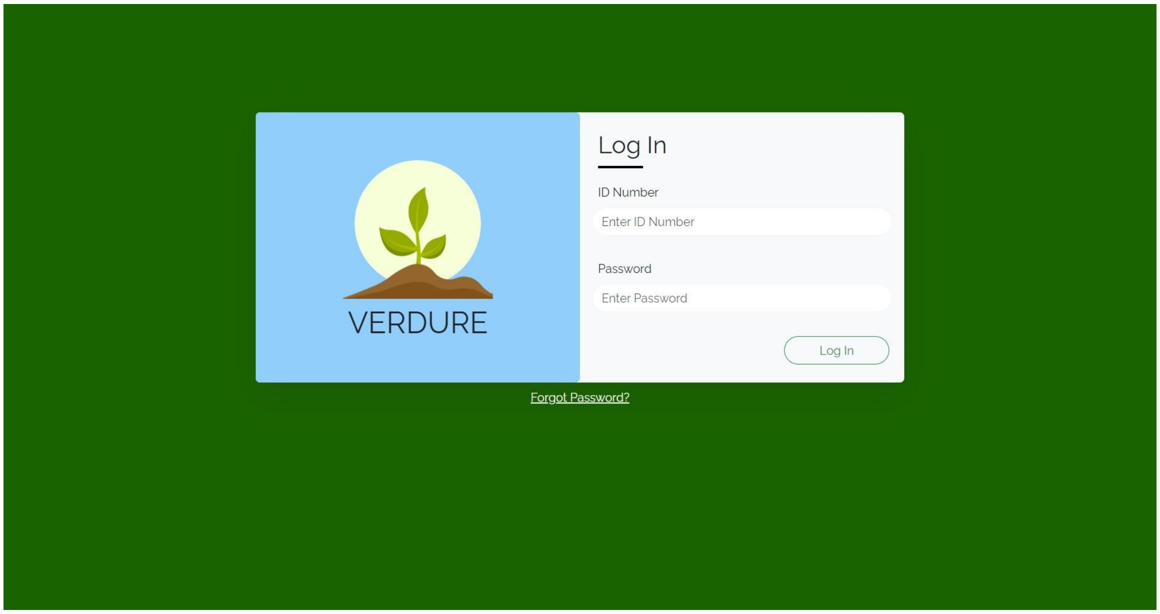

Figure 53. Log In Page.

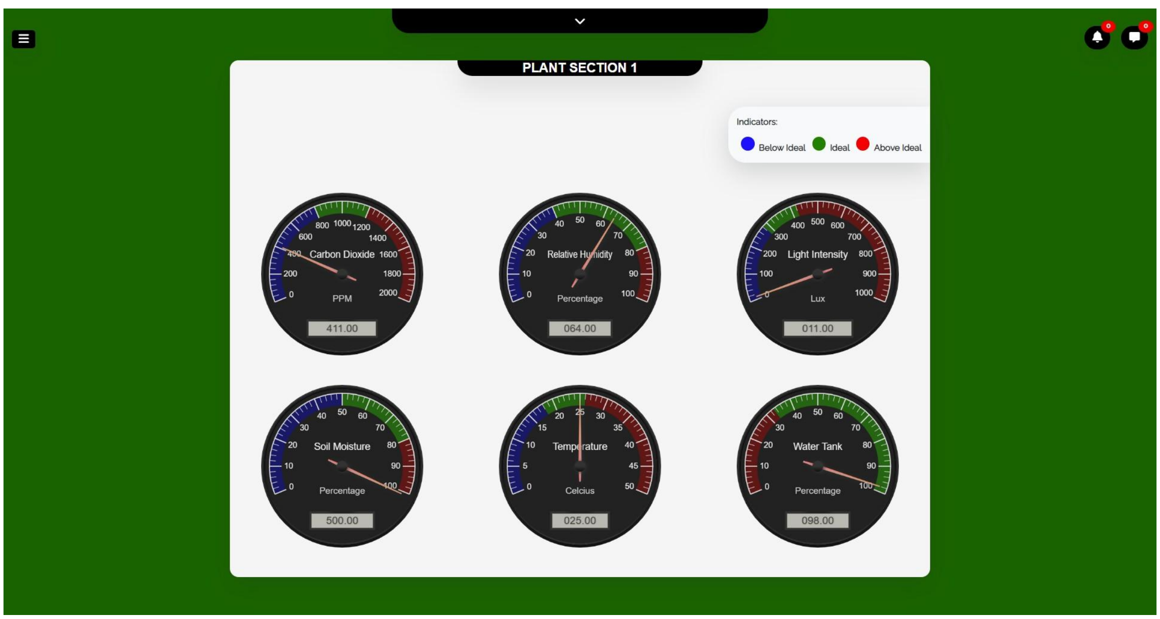

Figure 54. Meter Page.

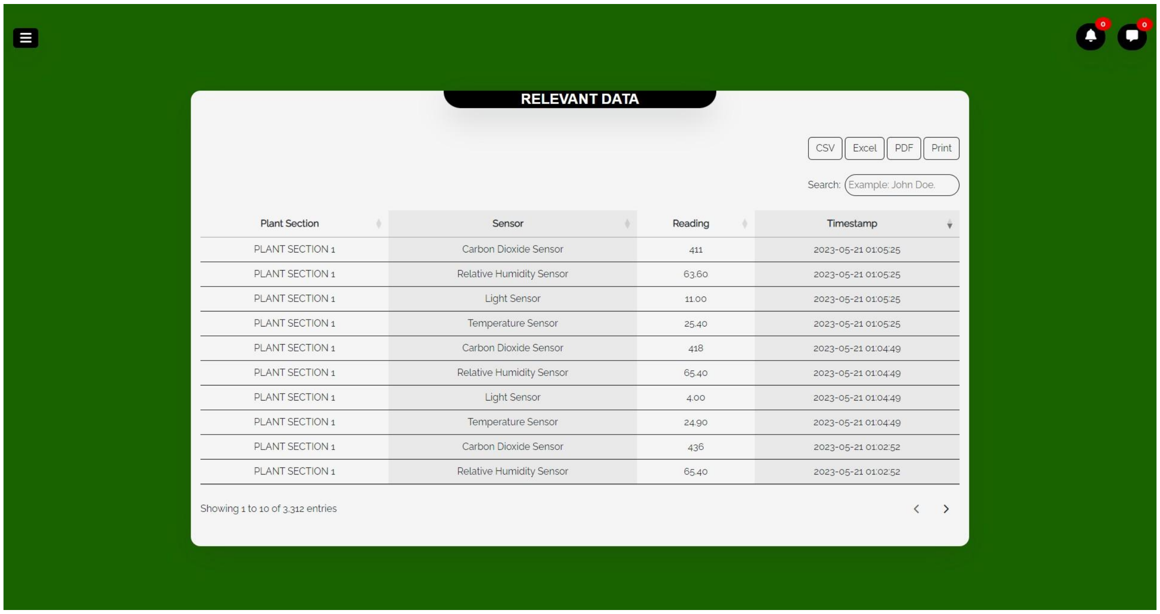

Figure 55. Meter Page.

Figure 56. Bypass Actuators Page.

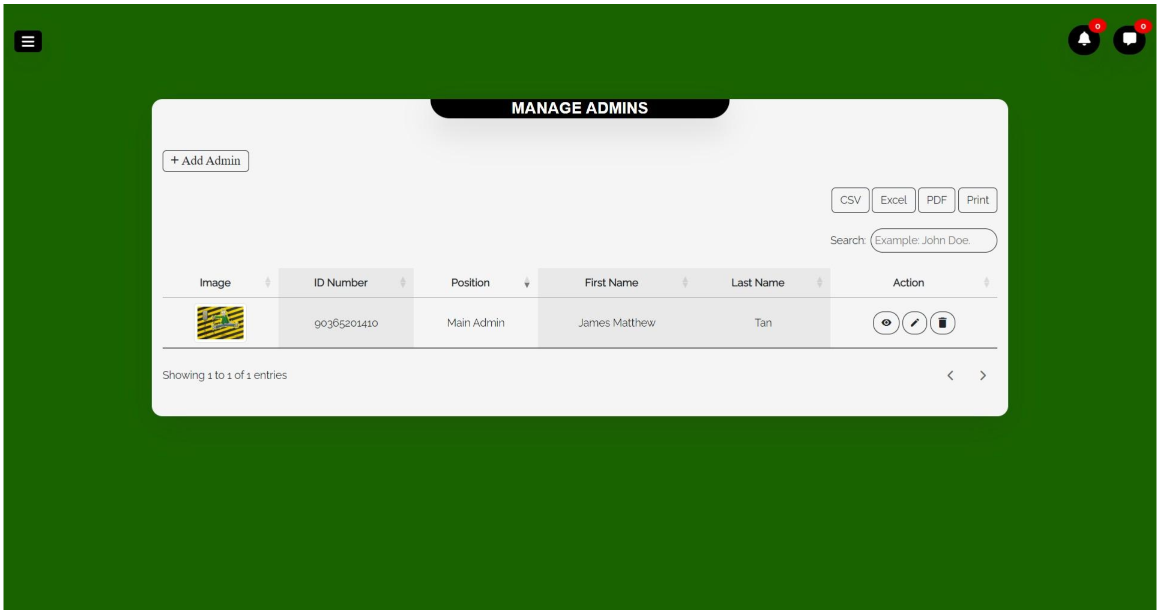

Figure 57. Manage Admins Page.

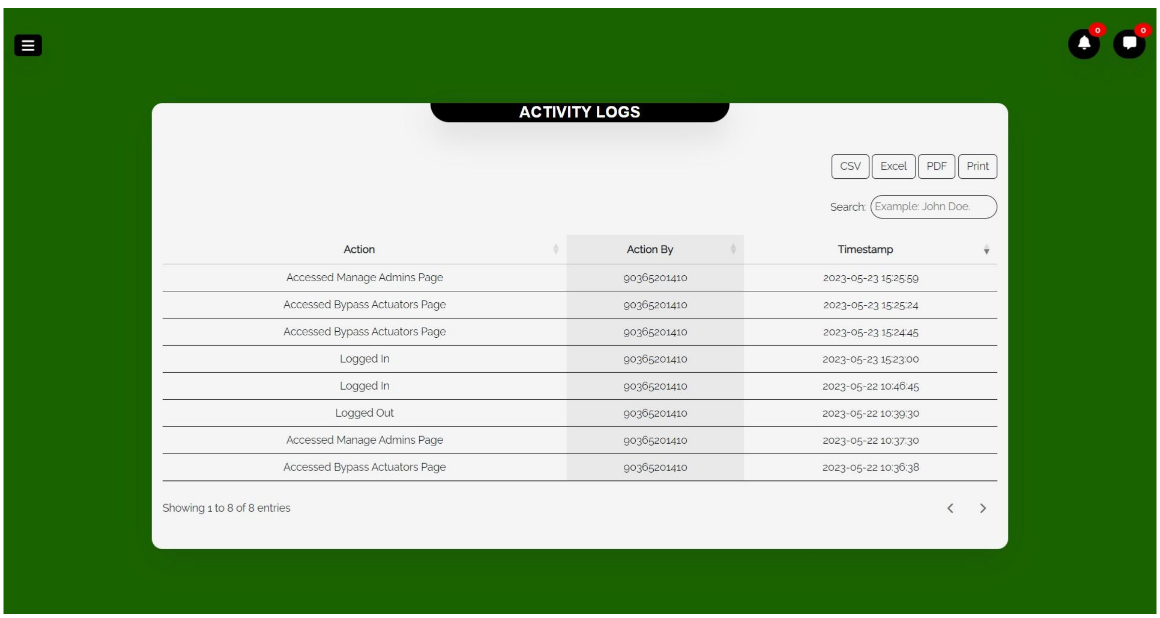

Figure 58. Activity Logs Page.

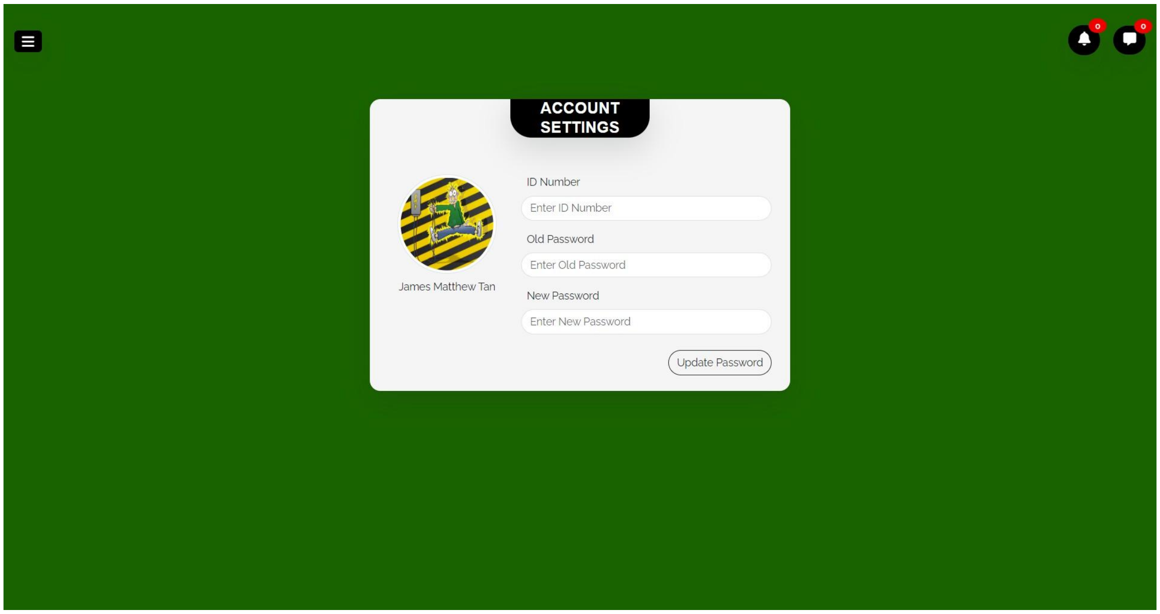

Figure 59. Account Settings Page.

Appendix E: Sample Testing Images

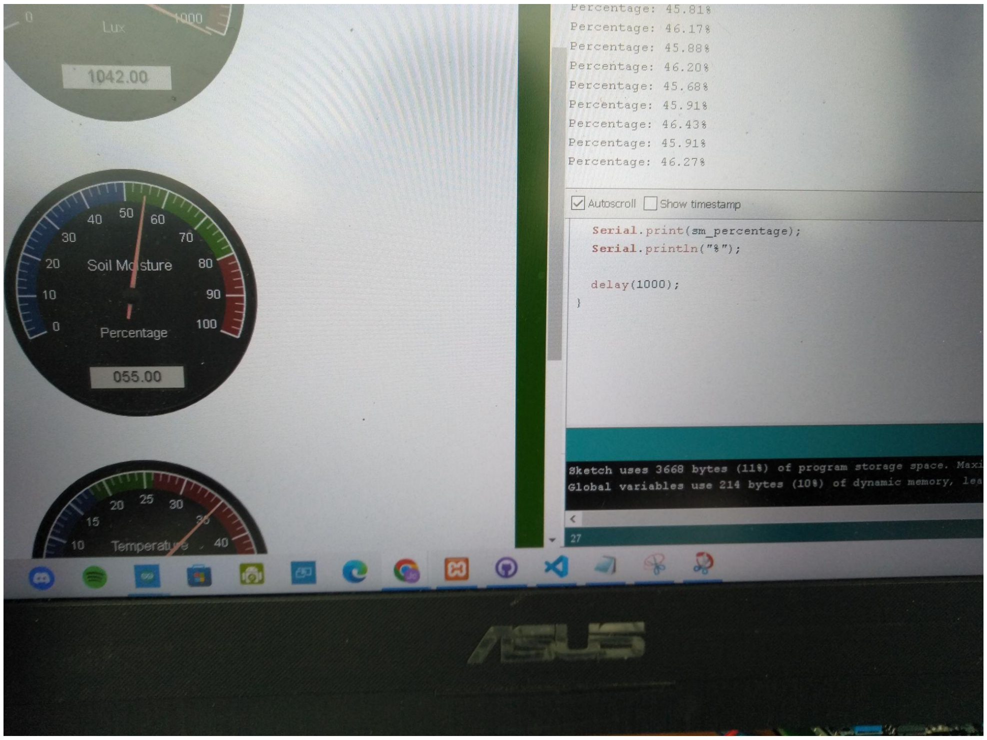

Figure 60. Testing 1 for the Accuracy of the Soil Moisture Sensor.

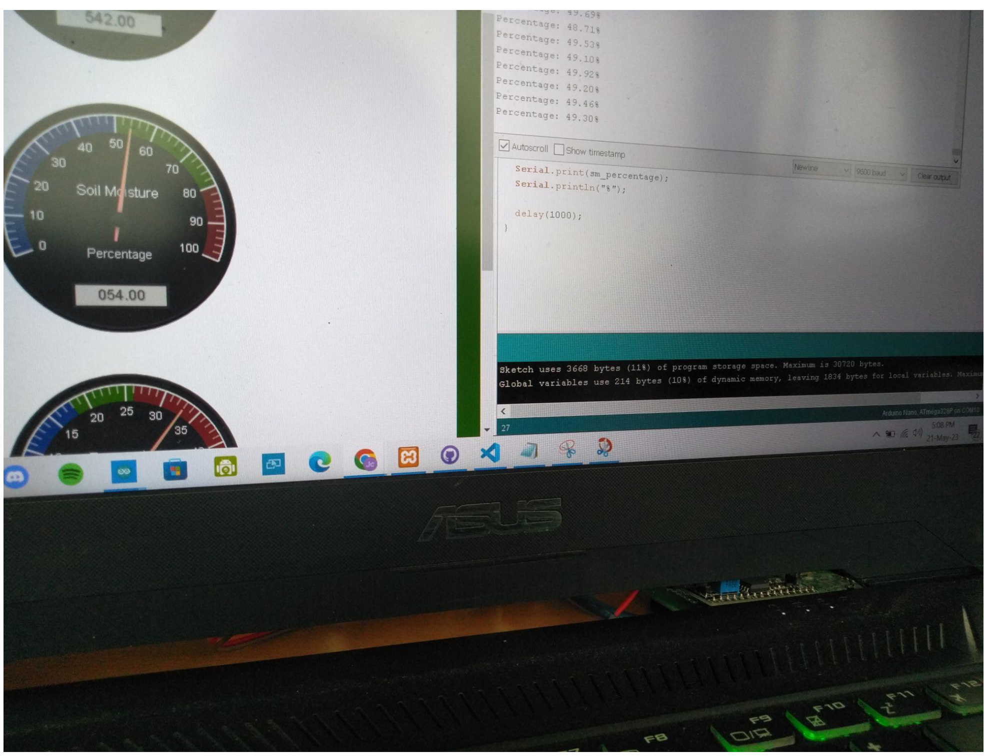

Figure 61. Testing 2 for the Accuracy of the Soil Moisture Sensor.

Figure 62. Testing 1 for the Accuracy of the Light Sensor.

Figure 63. Testing 2 for the Accuracy of the Light Sensor.

Figure 64. Testing 1 for the Accuracy of the Humidty Sensor.

Figure 65. Testing 2 for the Accuracy of the Humidty Sensor.

Figure 66. Testing 1 for the Accuracy of the Carbon Dioxide Sensor.

Figure 67. Testing 2 for the Accuracy of the Carbon Dioxide Sensor.

Figure 68. Testing 1 for the Accuracy of the Temperature Sensor.

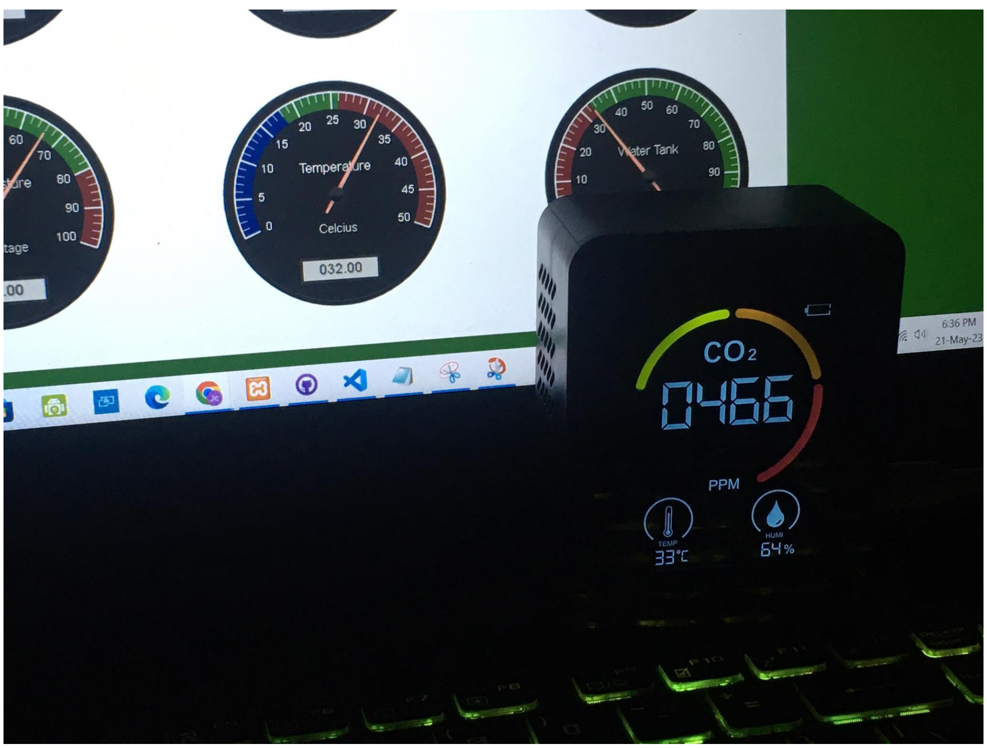

Figure 69. Testing 2 for the Accuracy of the Carbon Dioxide.

Appendix F: Schematic Diagrams

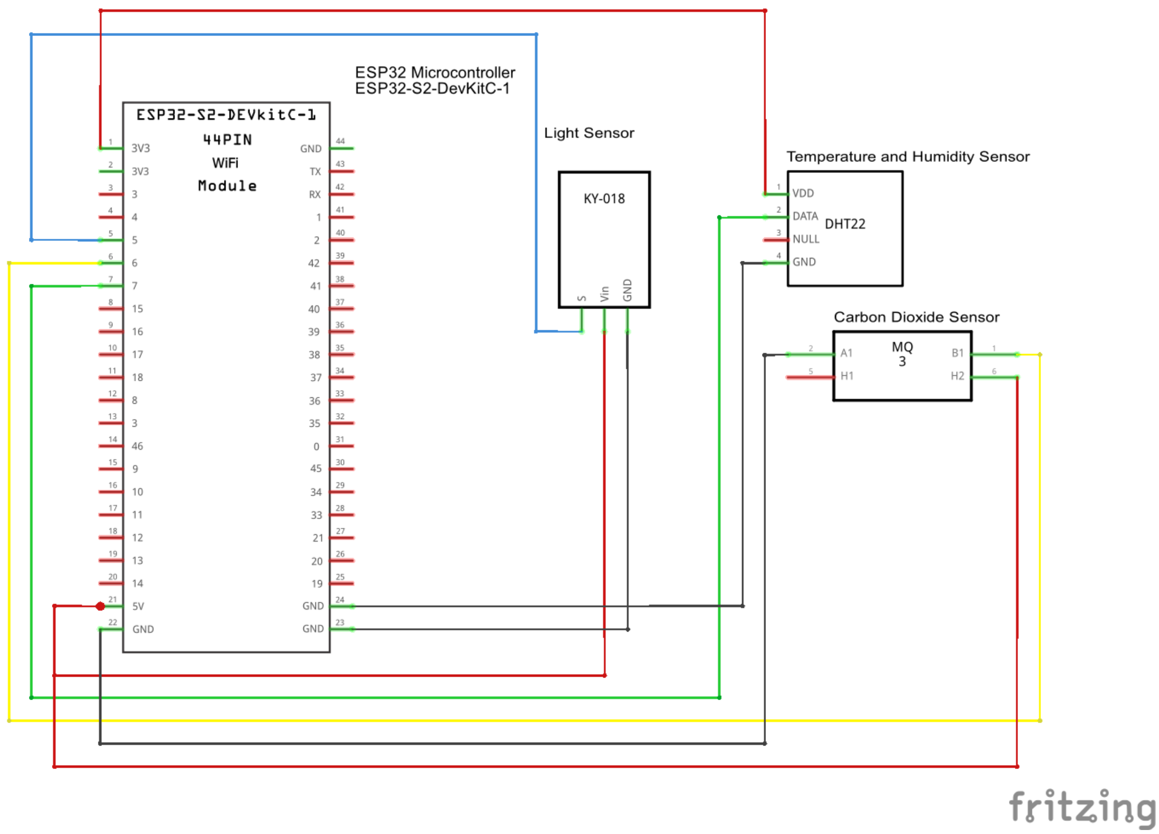

Figure 70. Schematic Diagram for the ESP32 (Carbon Dioxide, Humidity, Light, Temperature Sensor).

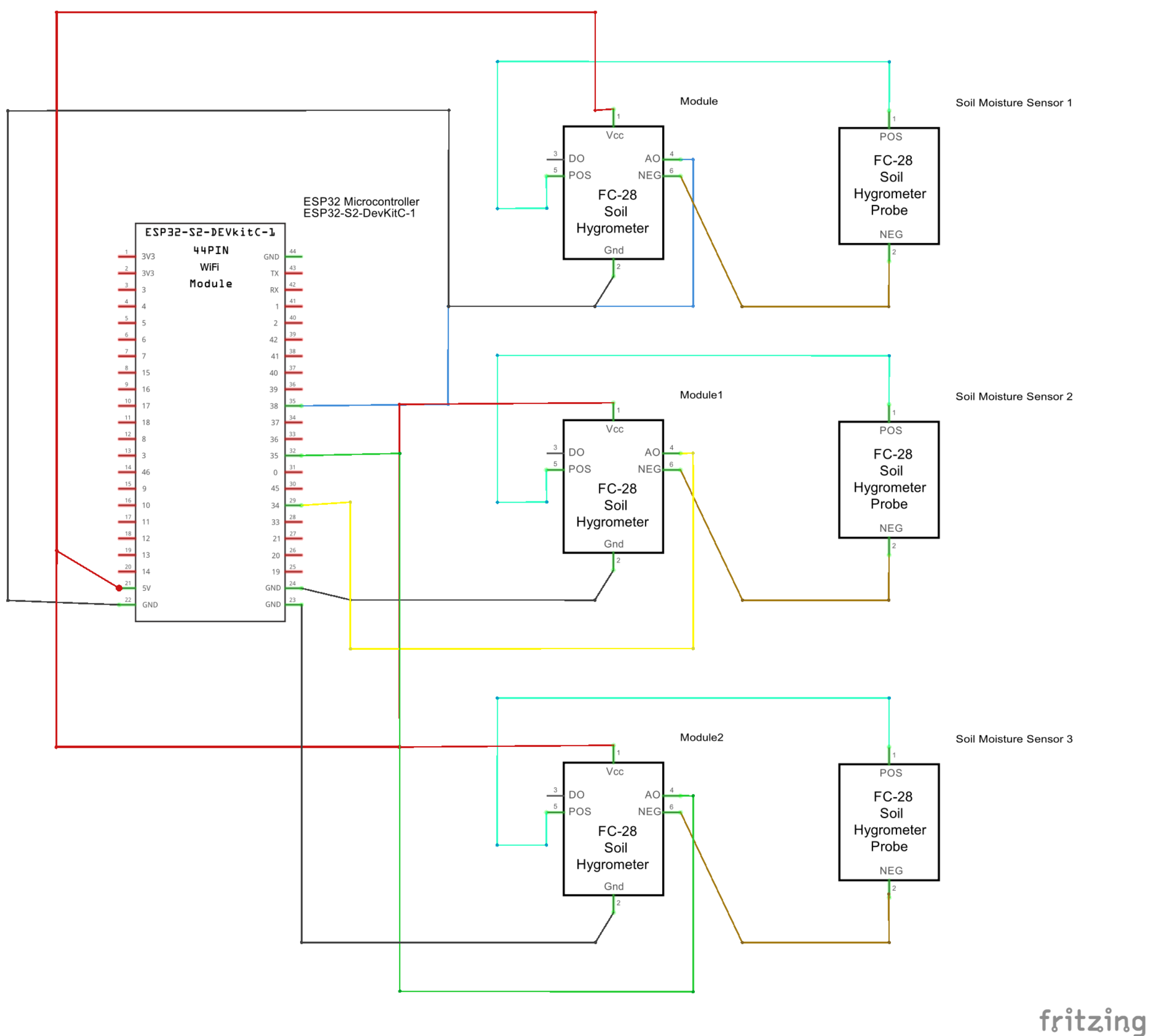

Figure 71. Schematic Diagram for the ESP32 (Soil Moisture Sensor).

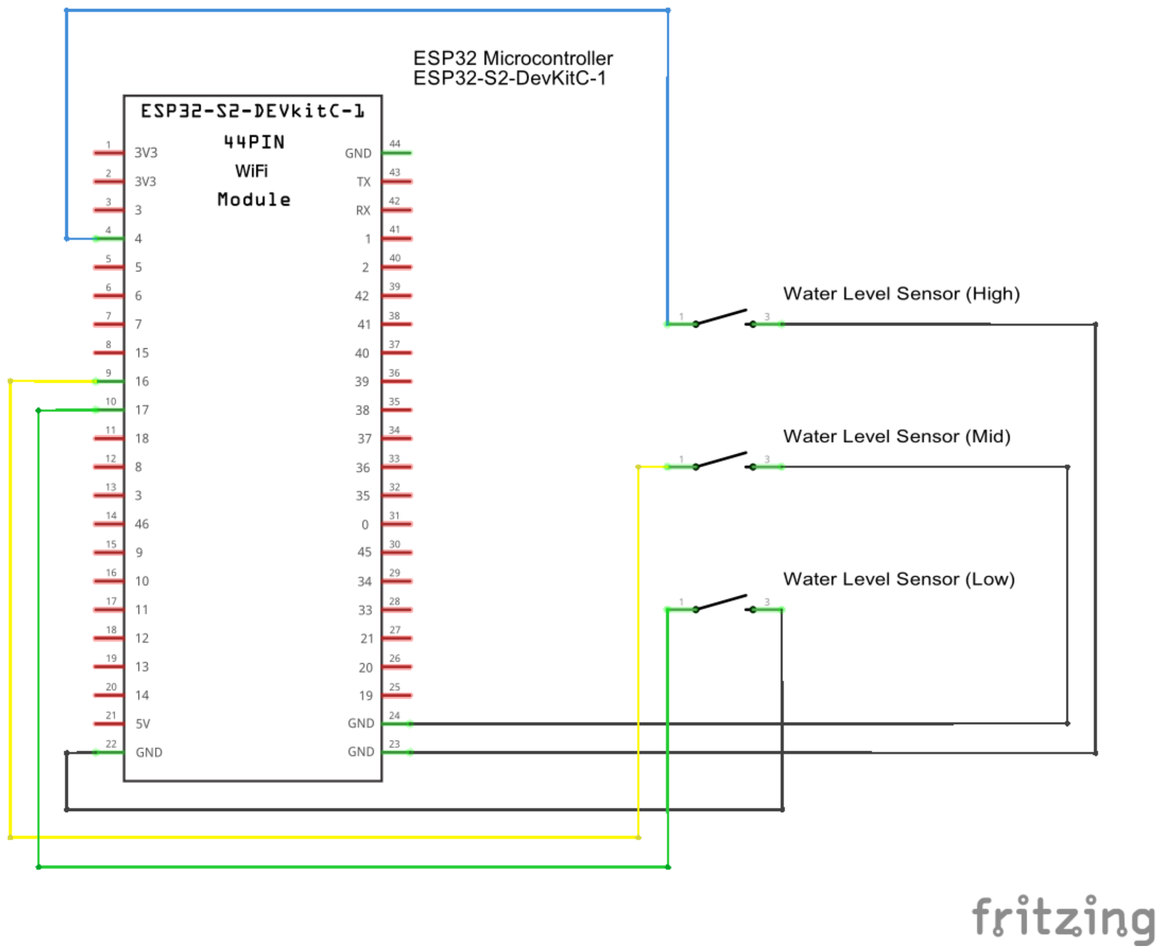

Figure 72. Schematic Diagram for the ESP32 (Water Level Sensor).

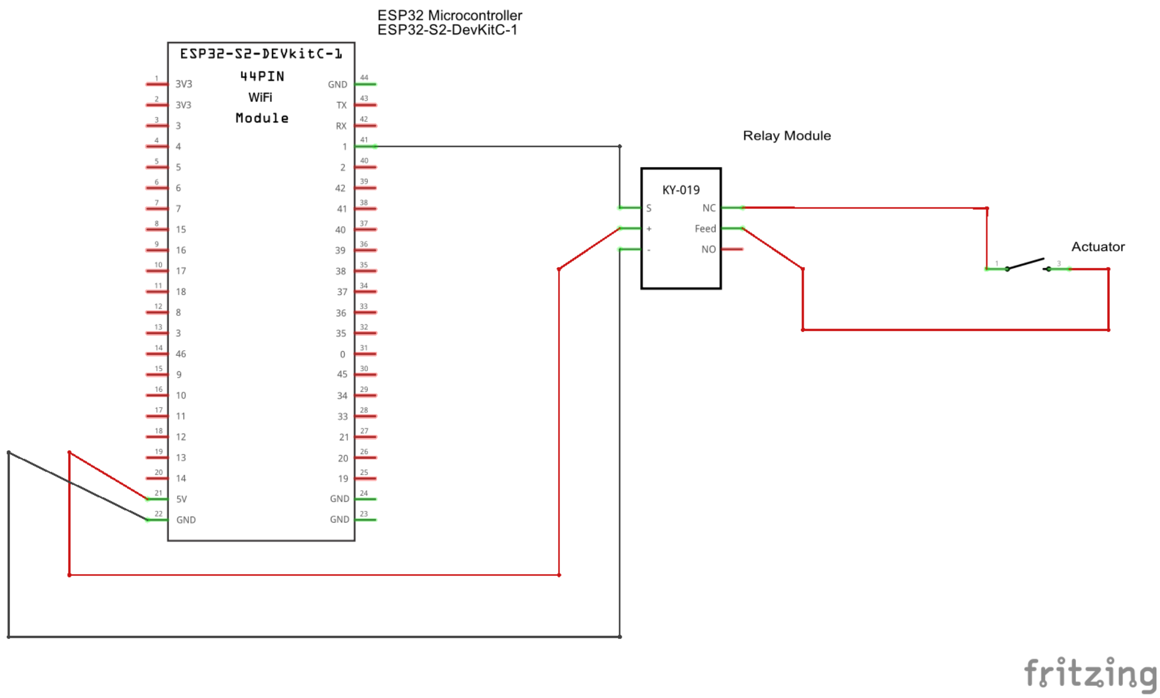

Figure 73. Schematic Diagram for the ESP32 (All Actuators).

Appendix G: Sample Source Codes

```
void loop() {
  // Read humidity and temperature from DHT sensor
  float humidity = dht.readHumidity();
  float temperature = dht.readTemperature();

  // Read lux value from light sensor
  float ls = analogRead(lightSensor);
  ls = (6000/4095) * ls;
  Serial.println(ls);

  // Read carbon dioxide value from MQ135
  float cd = gasSensor.getPPM();

  //Check WiFi connection status
  if(WiFi.status()== WL_CONNECTED){

    WiFiClient client;
    HTTPClient http;

    // Your Domain name with URL path or IP address with path
    http.begin(client, serverName);

    // Specify content-type header
    http.addHeader("Content-Type", "application/x-www-form-urlencoded");

    // Prepare your HTTP POST request data
    String httpRequestData = "api_key=" + apiKeyValue + "&location=" + location + "&carbon_dioxide_value=" +
String(cd) + "&relative_humidity_value=" + String(humidity) + "&light_intensity_value=" + String(ls) +
"&temperature_value=" + String(temperature) + "";
    Serial.print("httpRequestData: ");
    Serial.println(httpRequestData);

    // Send HTTP POST request
    int httpResponseCode = http.POST(httpRequestData);

    if (httpResponseCode>0) {
      Serial.print("HTTP Response code: ");
      Serial.println(httpResponseCode);
    }
    else {
      Serial.print("Error code: ");
      Serial.println(httpResponseCode);
    }
    // Free resources
    http.end();
  }
  else {
    Serial.println("WiFi Disconnected");
  }
  //Send an HTTP POST request every 35 seconds
  delay(35000);
}
```

Figure 74. Sending the Humidity, Temperature, Light, Carbon Dioxide Sensor Data to the MySQL Database.

```
void loop() {
  // Read the values from all three sensors
  float sm_1 = analogRead(sm_one);
  float sm_2 = analogRead(sm_two);
  float sm_3 = analogRead(sm_three);

  // Calculate the average sensor value
  float averageValue = (sm_1 + sm_2 + sm_3) / 3.0;

  // Calculate the percentage value
  float sm_percentage = (averageValue / maxSensorReading) * 100.0;

  // Print the percentage value to the serial monitor
  Serial.print("Percentage: ");
  Serial.print(sm_percentage);
  Serial.println("%");

  //Check WiFi connection status
  if(WiFi.status()== WL_CONNECTED){

    WiFiClient client;
    HTTPClient http;

    // Your Domain name with URL path or IP address with path
    http.begin(client, serverName);

    // Specify content-type header
    http.addHeader("Content-Type", "application/x-www-form-urlencoded");

    // Prepare your HTTP POST request data
    String httpRequestData = "api_key=" + apiKeyValue + "&location=" + location + "&soil_moisture_value=" +
String(sm_percentage) + "";
    Serial.print("httpRequestData: ");
    Serial.println(httpRequestData);

    // Send HTTP POST request
    int httpResponseCode = http.POST(httpRequestData);

    if (httpResponseCode>0) {
      Serial.print("HTTP Response code: ");
      Serial.println(httpResponseCode);
    }
    else {
      Serial.print("Error code: ");
      Serial.println(httpResponseCode);
    }
    // Free resources
    http.end();
  }
  else {
    Serial.println("WiFi Disconnected");
  }
  //Send an HTTP POST request every 35 seconds
  delay(35000);
}
```

Figure 75. Sending the Soil Moisture Sensor Data to the MySQL Database.

```
void loop() {
  // Read the values from all three sensors
  int wls_1 = digitalRead(wls_one);
  int wls_2 = digitalRead(wls_two);
  int wls_3 = digitalRead(wls_three);

  float wl_averageValue = (float)(wls_1 + wls_2 + wls_3) / 3.0;

  // Scale the average value to a percentage range
  float wl_percentage = (((wl_averageValue - minSensorReading) / (maxSensorReading - minSensorReading)) * 100);
  wl_percentage = (round(wl_percentage * 1000));
  // Print the percentage value to the serial monitor
  Serial.print("Water Level Percentage: ");
  Serial.print(wl_percentage);
  Serial.println("%");

  //Check WiFi connection status
  if(WiFi.status()== WL_CONNECTED){

    WiFiClient client;
    HTTPClient http;

    // Your Domain name with URL path or IP address with path
    http.begin(client, serverName);

    // Specify content-type header
    http.addHeader("Content-Type", "application/x-www-form-urlencoded");

    // Prepare your HTTP POST request data
    String httpRequestData = "api_key=" + apiKeyValue + "&location=" + location + "&water_level_value=" +
String(wl_percentage) + "";
    Serial.print("httpRequestData: ");
    Serial.println(httpRequestData);

    // Send HTTP POST request
    int httpResponseCode = http.POST(httpRequestData);

    if (httpResponseCode>0) {
      Serial.print("HTTP Response code: ");
      Serial.println(httpResponseCode);
    }
    else {
      Serial.print("Error code: ");
      Serial.println(httpResponseCode);
    }
    // Free resources
    http.end();
  }
  else {
    Serial.println("WiFi Disconnected");
  }
  //Send an HTTP POST request every 3 seconds
  delay(3000);
}
```

Figure 76. Sending the Water Level Sensor Data to the MySQL Database.

```
void loop() {
  // Send HTTP request to server
  HTTPClient http;
  http.begin(serverName);
  int httpCode = http.GET();
  String payload = http.getString();
  Serial.println(payload);
  http.end();

  // Parse JSON data
  const size_t capacity = JSON_OBJECT_SIZE(15) * 2 + 2*JSON_OBJECT_SIZE(15) + 180;
  DynamicJsonDocument doc(capacity);
  DeserializationError error = deserializeJson(doc, payload);
  if (error) {
    Serial.println("Failed to parse JSON data");
    Serial.println(error.c_str());
    return;
  }

  int bypass_actuators_switch_value = doc["bypass_actuators_switch_value"];
  int actuators_disabled = doc["actuators_disabled"];
  int grow_lights_switch_value = doc["grow_lights_switch_value"];
  int light_intensity_sensor_value = doc["light_intensity_sensor_value"];
  int current_time = doc["current_time"];

if(bypass_actuators_switch_value && !actuators_disabled){
  if(grow_lights_switch_value == 1){
    digitalWrite(grow_lights, HIGH);
  } else if(grow_lights_switch_value == 0) {
    digitalWrite(grow_lights, LOW);
  }
}

else if(!bypass_actuators_switch_value && actuators_disabled){
  if(current_time >= 6 && current_time <= 17){
    if(light_intensity_sensor_value < 400){
      digitalWrite(grow_lights, HIGH);
    } else if (light_intensity_sensor_value >= 400) {
      digitalWrite(grow_lights, LOW);
    }
  }
  else if(current_time > 17 || current_time < 6){
    digitalWrite(grow_lights, LOW);
  }
}

  // Wait for 25 seconds before sending another request
  delay(25000);
}
```

Figure 77. Receiving the Light Sensor Data from the MySQL Database to Automate and Control the Grow Light.

```
void loop() {
  // Send HTTP request to server
  HTTPClient http;
  http.begin(serverName);
  int httpCode = http.GET();
  String payload = http.getString();
  Serial.println(payload);
  http.end();

  // Parse JSON data
  const size_t capacity = JSON_OBJECT_SIZE(15) * 2 + 2*JSON_OBJECT_SIZE(15) + 180;
  DynamicJsonDocument doc(capacity);
  DeserializationError error = deserializeJson(doc, payload);
  if (error) {
    Serial.println("Failed to parse JSON data");
    Serial.println(error.c_str());
    return;
  }

  int bypass_actuators_switch_value = doc["bypass_actuators_switch_value"];
  int actuators_disabled = doc["actuators_disabled"];
  int evaporative_air_cooler_switch_value = doc["evaporative_air_cooler_switch_value"];
  int temperature_sensor_value = doc["temperature_sensor_value"];
  int current_time = doc["current_time"];

if(bypass_actuators_switch_value && !actuators_disabled){
  if(evaporative_air_cooler_switch_value == 1){
    digitalWrite(evaporative_air_cooler, HIGH);
  } else if(evaporative_air_cooler_switch_value == 0) {
    digitalWrite(evaporative_air_cooler, LOW);
  }
}

else if(!bypass_actuators_switch_value && actuators_disabled){
  if(current_time >= 6 && current_time <= 17){
    if(temperature_sensor_value > 26){
      digitalWrite(evaporative_air_cooler, HIGH);
    } else if (temperature_sensor_value <= 26) {
      digitalWrite(evaporative_air_cooler, LOW);
    }
  }
  else if(current_time > 17 || current_time < 6){
    digitalWrite(evaporative_air_cooler, LOW);
  }
}

  // Wait for 25 seconds before sending another request
  delay(25000);
}
```

Figure 78. Receiving the Temperature Sensor Data from the MySQL Database to Automate and Control the Fan.

```
void loop() {
  // Send HTTP request to server
  HTTPClient http;
  http.begin(serverName);
  int httpCode = http.GET();
  String payload = http.getString();
  Serial.println(payload);
  http.end();

  // Parse JSON data
  const size_t capacity = JSON_OBJECT_SIZE(15) * 2 + 2*JSON_OBJECT_SIZE(15) + 180;
  DynamicJsonDocument doc(capacity);
  DeserializationError error = deserializeJson(doc, payload);
  if (error) {
    Serial.println("Failed to parse JSON data");
    Serial.println(error.c_str());
    return;
  }

  int bypass_actuators_switch_value = doc["bypass_actuators_switch_value"];
  int actuators_disabled = doc["actuators_disabled"];
  int water_sprinkler_switch_value = doc["water_sprinkler_switch_value"];
  int relative_humidity_sensor_value = doc["relative_humidity_sensor_value"];
  int current_time = doc["current_time"];

if(bypass_actuators_switch_value && !actuators_disabled){
  if(water_sprinkler_switch_value == 1){
    digitalWrite(water_sprinkler, HIGH);
  } else if(water_sprinkler_switch_value == 0) {
    digitalWrite(water_sprinkler, LOW);
  }
}

else if(!bypass_actuators_switch_value && actuators_disabled){
  if(current_time >= 6 && current_time <= 17){
    if(relative_humidity_sensor_value < 40){
      digitalWrite(water_sprinkler, HIGH);
    } else if (relative_humidity_sensor_value >= 40) {
      digitalWrite(water_sprinkler, LOW);
    }
  }
  else if(current_time > 17 || current_time < 6){
    digitalWrite(water_sprinkler, LOW);
  }
}

  // Wait for 25 seconds before sending another request
  delay(25000);
}
```

Figure 79. Receiving the Humidity Sensor Data from the MySQL Database to Automate and Control the Sprinkler.

```php
<?php

include "Database.php";

$sql = "SELECT * FROM bypass_actuators";
$result = $mysqli->query($sql);

$sql_two = "SELECT * FROM sensor_data";
$result_two = $mysqli->query($sql_two);

date_default_timezone_set('Asia/Manila'); // Set timezone to Philippines

$current_time = date('H'); // Get current time in 24-hour format

if ($result->num_rows > 0 && $result_two->num_rows > 0) {

    // Initialize an empty array to store the combined data
    $data = array();

    // Loop through rows in bypass_actuators table
    while ($row = $result->fetch_assoc()) {
        $data['bypass_actuators_switch_value'] = $row["Bypass_Actuators_Switch"];
        $data['actuators_disabled'] = $row["Actuators_Disabled"];
        $data['exhaust_fan_switch_value'] = $row["Exhaust_Fan_Switch"];
        $data['water_sprinkler_switch_value'] = $row["Water_Sprinkler_Switch"];
        $data['grow_lights_switch_value'] = $row["Grow_Lights_Switch"];
        $data['water_hose_switch_value'] = $row["Water_Hose_Switch"];
        $data['evaporative_air_cooler_switch_value'] = $row["Evaporative_Air_Cooler_Switch"];
        $data['water_hose_two_switch_value'] = $row["Water_Hose_Two_Switch"];
    }

    // Loop through rows in sensor_data table
    while ($row = $result_two->fetch_assoc()) {
        $data['carbon_dioxide_sensor_value'] = $row["Carbon_Dioxide_Sensor"];
        $data['relative_humidity_sensor_value'] = $row["Relative_Humidity_Sensor"];
        $data['light_intensity_sensor_value'] = $row["Light_Intensity_Sensor"];
        $data['soil_moisture_sensor_value'] = $row["Soil_Moisture_Sensor"];
        $data['temperature_sensor_value'] = $row["Temperature_Sensor"];
        $data['water_level_sensor_value'] = $row["Water_Level_Sensor"];
    }

    $data['current_time'] = $current_time;

    // Encode the combined data as JSON and output it
    header('Content-Type: application/json');
    echo json_encode($data);

} else {

    echo "0 results";

}

?>
```

Figure 80. Sending the Data from MySQL Database to the ESP32 to Automate and Control the Actuators.

```php
<?php

include "Database.php";

// Keep this API Key value to be compatible with the ESP32 code provided in the project page.
// If you change this value, the ESP32 sketch needs to match
$api_key_value = "tPmAT5Ab3j7F9";

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $api_key = sensor_data($_POST["api_key"]);
    if ($api_key == $api_key_value) {
        $location = sensor_data($_POST["location"]);
        $carbon_dioxide_value = sensor_data($_POST["carbon_dioxide_value"]);
        $relative_humidity_value = sensor_data($_POST["relative_humidity_value"]);
        $light_intensity_value = sensor_data($_POST["light_intensity_value"]);
        $temperature_value = sensor_data($_POST["temperature_value"]);

        $sql = "UPDATE sensor_data SET Location = '$location', Carbon_Dioxide_Sensor = '$carbon_dioxide_value',
Relative_Humidity_Sensor = '$relative_humidity_value', Light_Intensity_Sensor = '$light_intensity_value',
Temperature_Sensor = '$temperature_value' WHERE Location = '$location'";
        $query_run = mysqli_query($mysqli, $sql);

        if ($query_run) {
            $sql_two = "INSERT INTO relevant_data (Plant_Section, Sensor, Reading) VALUES ('$location', 'Carbon
Dioxide Sensor', '$carbon_dioxide_value')";
            $query_two = mysqli_query($mysqli, $sql_two);
            $sql_nine = "UPDATE relevant_data_notification SET Unread = Unread + 1";
            $query_run_nine = mysqli_query($mysqli, $sql_nine);

            $sql_three = "INSERT INTO relevant_data (Plant_Section, Sensor, Reading) VALUES ('$location', 'Relative
Humidity Sensor', '$relative_humidity_value')";
            $query_three = mysqli_query($mysqli, $sql_three);
            $sql_ten = "UPDATE relevant_data_notification SET Unread = Unread + 1";
            $query_run_ten = mysqli_query($mysqli, $sql_ten);

            $sql_four = "INSERT INTO relevant_data (Plant_Section, Sensor, Reading) VALUES ('$location', 'Light
Sensor', '$light_intensity_value')";
            $query_four = mysqli_query($mysqli, $sql_four);
            $sql_eleven = "UPDATE relevant_data_notification SET Unread = Unread + 1";
            $query_run_eleven = mysqli_query($mysqli, $sql_eleven);

            $sql_six = "INSERT INTO relevant_data (Plant_Section, Sensor, Reading) VALUES ('$location', 'Temperature
Sensor', '$temperature_value')";
            $query_six = mysqli_query($mysqli, $sql_six);
            $sql_thirteen = "UPDATE relevant_data_notification SET Unread = Unread + 1";
            $query_run_thirteen = mysqli_query($mysqli, $sql_thirteen);

            $mysqli->query("SET @num := 0;");
            $mysqli->query("UPDATE sensor_data SET id = @num := (@num + 1);");
            $mysqli->query("ALTER TABLE sensor_data AUTO_INCREMENT = 1;");

            $mysqli->query("SET @num := 0;");
            $mysqli->query("UPDATE relevant_data SET id = @num := (@num + 1);");
            $mysqli->query("ALTER TABLE relevant_data AUTO_INCREMENT = 1;");

            echo "New record created successfully";
        } else {
            echo "Error: " . $sql . "<br>" . $conn->error;
        }
    } else {
        echo "Wrong API Key provided.";
    }
} else {
    echo "No data posted with HTTP POST.";
}

function sensor_data($data)
{
    $data = trim($data);
    $data = stripslashes($data);
    $data = htmlspecialchars($data);
    return $data;
}

?>
```

Figure 81. Insert the Sensor Data coming from the ESP32 Microcontroller into the MySQL Database.

```
<script>

  function sensors_data() {
    var xhr = new XMLHttpRequest();
    xhr.open('GET', 'http://192.168.1.50:1912/Verdure/Github/PHPMySQL/Display_Sensor_Data_In_Meter_Page.php', true);
    xhr.onreadystatechange = function() {
      if (xhr.readyState === XMLHttpRequest.DONE && xhr.status === 200) {
        var data = JSON.parse(xhr.responseText);
        <?php
        $sql = "SELECT * FROM sensor_data";
        $result = $mysqli->query($sql);

        if ($result->num_rows > 0) {
          // output data of each row
          while ($row = $result->fetch_assoc()) {
            $id = $row["ID"];
            $data = $row["Data"];
        ?>
            // update the location div with the first location value in the data
            document.getElementById('carbon_dioxide_gauge_<?php echo $id ?>').setAttribute("data-value", data[<?php
echo $data ?>].carbon_dioxide_sensor_value);
            document.getElementById('relative_humidity_gauge_<?php echo $id ?>').setAttribute("data-value", data[<?
php echo $data ?>].relative_humidity_sensor_value);
            document.getElementById('light_intensity_gauge_<?php echo $id ?>').setAttribute("data-value", data[<?php
echo $data ?>].light_intensity_sensor_value);
            document.getElementById('soil_moisture_gauge_<?php echo $id ?>').setAttribute("data-value", data[<?php
echo $data ?>].soil_moisture_sensor_value);
            document.getElementById('temperature_gauge_<?php echo $id ?>').setAttribute("data-value", data[<?php
echo $data ?>].temperature_sensor_value);
            document.getElementById('water_level_gauge_<?php echo $id ?>').setAttribute("data-value", data[<?php
echo $data ?>].water_level_sensor_value);
        <?php
          }
        } else {
          echo "0 results";
        }
        ?>
      }
    };
    xhr.send();
  }

  setInterval(sensors_data, 25000);

</script>
```

Figure 82. Display the Sensor Data to the Meter Page.

```php
<?php

include "Database.php";

$sql = "SELECT * FROM sensor_data";
$result = $mysqli->query($sql);

$data = [];

if ($result->num_rows > 0) {
    // output data of each row
    while ($row = $result->fetch_assoc()) {
        $location = $row["Location"];
        $carbon_dioxide_sensor_value = $row["Carbon_Dioxide_Sensor"];
        $relative_humidity_sensor_value = $row["Relative_Humidity_Sensor"];
        $light_intensity_sensor_value = $row["Light_Intensity_Sensor"];
        $soil_moisture_sensor_value = $row["Soil_Moisture_Sensor"];
        $temperature_sensor_value = $row["Temperature_Sensor"];
        $water_level_sensor_value = $row["Water_Level_Sensor"];

        $data[] = [
            'location' => $location,
            'carbon_dioxide_sensor_value' => $carbon_dioxide_sensor_value,
            'relative_humidity_sensor_value' => $relative_humidity_sensor_value,
            'light_intensity_sensor_value' => $light_intensity_sensor_value,
            'soil_moisture_sensor_value' => $soil_moisture_sensor_value,
            'temperature_sensor_value' => $temperature_sensor_value,
            'water_level_sensor_value' => $water_level_sensor_value
        ];
    }
} else {
    echo "0 results";
}

echo json_encode($data);

?>
```

Figure 83. Pulling the Data from the MySQL Database to Display in Meter Page.

Appendix H: Sensors Accuracy Tables

Table 4: Accuracy of the Carbon Dioxide Sensor

	System Sensor (PPM)	**Third Party Sensor (PPM)**	**Percentage Error (%)**
1	475	460	3.26
2	421	560	24.82
3	442	519	14.84
4	481	479	0.42
5	491	467	5.14
6	415	521	20.35
7	487	494	1.42
8	450	504	10.71
9	481	489	1.64
10	427	450	5.11
11	492	559	11.99
12	430	565	23.89
13	488	510	4.31
14	469	511	8.22
15	405	536	24.44
16	474	450	5.33
17	493	460	7.17
18	470	453	3.75
19	440	463	4.97
20	406	449	9.58

21	469	499	6.01
22	498	455	9.45
23	406	473	14.16
24	419	474	11.60
25	429	490	12.45
26	405	505	19.80
27	487	494	1.42
28	431	474	9.07
29	420	468	10.26
30	463	458	1.09
Average	452.13	489.63	9.56

Table 5: Accuracy of the Humidity Sensor.

	System Sensors (%)	**Third Party Sensors (%)**	**Percentage Error (%)**
1	63	47	34.04
2	64	49	30.61
3	63	49	28.57
4	64	50	28.00
5	63	47	34.04
6	62	46	34.78
7	63	50	26.00
8	62	48	29.17

9	63	49	28.57
10	62	49	26.53
11	63	47	34.04
12	62	48	29.17
13	62	46	34.78
14	63	48	31.25
15	64	49	30.61
16	65	48	35.42
17	66	50	32.00
18	67	50	34.00
19	66	49	34.69
20	65	50	30.00
21	64	49	30.61
22	65	49	32.65
23	64	49	30.61
24	65	50	30.00
25	67	51	31.37
26	68	51	33.33
27	67	51	31.37
28	68	52	30.77
29	69	50	38.00
30	68	53	28.30
Average	64.57	49.13	31.44

Table 6: Accuracy of the Temperature Sensor

	System Sensor (℃)	Third Party Sensor (℃)	Percentage Error (℃)
1	32	31	3.23
2	32	32	0.00
3	33	32	3.13
4	31	33	6.06
5	32	32	0.00
6	32	32	0.00
7	32	33	3.03
8	31	33	6.06
9	32	34	5.88
10	33	34	2.94
11	31	33	6.06
12	33	33	0.00
13	33	34	2.94
14	31	34	8.82
15	32	34	5.88
16	33	33	0.00
17	32	34	5.88
18	32	34	5.88
19	33	34	2.94
20	30	34	11.76

21	31	34	8.82
22	34	34	0.00
23	31	35	11.43
24	32	35	8.57
25	31	34	8.82
26	32	34	5.88
27	31	34	8.82
28	32	34	5.88
29	31	34	8.82
30	33	34	2.94
Average	31.93	33.5	5.02

Table 7: Accuracy of the Light Sensor

	System Sensor (Lux)	**Third Party Sensor (Lux)**	**Percentage Error (%)**
1	2731	3043	10.25
2	2932	2997	2.17
3	2847	2909	2.13
4	2709	2857	5.18
5	2686	2801	4.11
6	2624	2637	0.49
7	2516	2586	2.71
8	2366	2473	4.33

9	2141	2370	9.66
10	2202	2138	2.99
11	2059	1981	3.94
12	1988	1922	3.43
13	1836	1729	6.19
14	1156	1296	10.80
15	1165	1272	8.41
16	1083	1284	15.65
17	1039	1286	19.21
18	1042	1175	11.32
19	1077	1058	1.80
20	974	1051	7.33
21	926	1115	16.95
22	1229	1158	6.13
23	913	1164	21.56
24	1018	1200	15.17
25	1179	1197	1.50
26	297	220	35.00
27	223	223	0.00
28	251	252	0.40
29	218	275	20.73
30	240	261	8.05
Average	1522.23	1597.67	8.59

Table 8: Accuracy of the Soil Moisture.

	System Sensor (%)	**Third Party Sensor (%)**	**Percentage Error (%)**
1	54	45.19	19.50
2	54	45.13	19.65
3	53	45.94	15.37
4	53	46.01	15.19
5	54	45.91	17.62
6	53	45.45	16.61
7	54	45.65	18.29
8	54	45.72	18.11
9	54	46.17	16.96
10	55	46.27	18.87
11	55	46.86	17.37
12	53	46.53	13.91
13	54	46.63	15.81
14	53	47.93	10.58
15	54	48.19	12.06
16	53	49.04	8.08
17	54	48.55	11.23
18	53	48.81	8.58
19	53	48.81	8.58
20	54	49.14	9.89

21	54	49.3	9.53
22	54	49.22	9.71
23	54	49.53	9.02
24	54	49.17	9.82
25	54	49.43	9.25
26	54	49.33	9.47
27	54	49.59	8.89
28	54	49.53	9.02
29	54	49.43	9.25
30	54	49.59	8.89
Average	53.8	47.74	12.84