Pi-Droponics: An Automated Hydroponics System with Notification System using Convolutional Neural Network

A Project Proposal Presented to the Faculty of

Electronics Engineering Department

College of Engineering

Technological University of the Philippines

In Partial Fulfilment of the Subject Requirements for the Degree of

Bachelor of Science in Electronics Engineering

Submitted by:

MARACHA, Curt Johann M.

BALAIS, Blessie Mari S.

BANSIL, Roden Leo Q.

CASTILLO, Kaye Celene S.

LIBANG, Gemmilyn T.

Adviser:

Engr. Mark P. Melegrito

August 2023

APPROVAL SHEET

This project study entitled **"Pi-Hydroponics: An Automated Hydroponics System with Notification System using Convolutional Neural Network"**, has been prepared and submitted by the following proponents:

Balais, Blessie Mari S.
Libang, Gemmilyn T.
Bansil, Roden Leo Q.
Maracha, Curt Johann M.
Castillo, Kaye Celene S.

In partial fulfillment of the requirements for the **Degree of Bachelor of Science in Electronics Engineering is hereby recommended for approval.**

Engr. Mark P. Melegrito
Project Adviser

Engr. Jessica S. Velasco
Panel Chair

Engr. Jomer V. Catipon
Panel Member

Engr. August Thio-Ac
Panel Member

Accepted and approved in partial fulfillment of the requirements for the **Degree of Bachelor of Science in Electronics Engineering**

Engr. Timothy M. Amado
Head, ECE Department

Engr. Nilo M. Arago
Dean, College of Engineering

ABSTRACT

Hydroponics is a soilless farming method that utilizes technology to regulate the growth environment. A two-layer vertical hydroponic system was constructed in a greenhouse utilizing the Nutrient Film Technique, including sensors that keep track of the light intensity, air temperature, TDS, and pH levels. The objective is to develop a fully automated hydroponic system with alerts for monitoring plant growth, detecting diseases, and estimating the harvest date. The sensor data is sent to the NodeMCU esp8266 and saved in Firebase, while a CNN algorithm is used to create a plant disease detection model, which is stored in a Raspberry Pi 4B. The app retrieves data from Firebase and sends real-time updates on system conditions, plants' health status, and notifications when sensor values are outside the threshold range or diseases are detected. The trials demonstrated the sensor readings to compare with those industrial-grade devices. The testing showed average percentage errors of 5.38% for pH sensor vs. pH meter, 2.6% for TDS sensor vs. TDS meter, 2.96% for light intensity sensor vs. lux meter, 3.84% for air temperature sensor vs. hygrometer, and 7.85% for humidity sensor vs. hygrometer. Accurate pH and TDS readings effectively monitor the greenhouse environment for the hydroponics system. Air humidity, temperature, and light intensity sensors indicate no significant difference between the greenhouse and the outside environment. A disease detection model using CNNs was developed and successfully tested on external plant leaves. The healthy state of the system's plants indirectly validates the model's effectiveness, although direct testing with the system's plants was not conducted.

ACKNOWLEDGEMENT

The proponents of the project "Pi-Hydroponics: An Automated Hydroponics System with Notification System using Convolutional Neural Network," also known as team HyPi IV, would like to express our heartfelt gratitude to God Almighty for His guidance and provisions that made this study possible despite the challenges posed by a partially online setup during the COVID-19 pandemic, the pressure of managing our academic workload, exams, and other personal commitments.

First and foremost, we extend our special gratitude to Engr. Glenn C. Virrey and Engr. Mark P. Melegrito, our dedicated Project Study Advisers, for their unwavering support and dedication throughout the project. We are immensely grateful for the time they invested and the invaluable assistance they provided. We would also like to express our appreciation to Engr. Nilo Arago, our esteemed Project Study Professor and College Dean, for consistently offering his time, experience, and knowledge to guide us on this journey.

We would like to convey our sincere appreciation to the ECE Faculty and our esteemed panelists for their invaluable assistance and unwavering support during every step of our project defense. We are truly grateful for the opportunity they gave us to express our ideas and for their continuous suggestions, which significantly contributed to the enhancement of our work throughout the completion process. Their insightful thoughts and expertise have played a crucial role in the development of our project.

Furthermore, we would like to express our heartfelt gratitude to the Barangay officials of Barangay 786 Zone 86, Sta. Ana Manila, for granting us permission to deploy the project on the rooftop of the barangay hall. We are also deeply grateful to our families for their support during the prototype construction stage. Their financial assistance and generous accommodation, allowing us to stay and work on the project study, are sincerely appreciated. Additionally, we would like to express our gratitude to all those who contributed to the successful completion of this study, even if not explicitly mentioned, your support has been invaluable to us.

TABLE OF CONTENTS

Approval Sheet.. ii
Abstract.. iii
Acknowledgment.. iv
Table of Contents.. vi
a. List of Figures.. ix
b. List of Tables.. xii
a. List of Annexes.. xiii

CHAPTER 1: THE PROBLEM AND ITS SETTING 1
1.1 Introduction..1
1.2 Background of the Study..3
1.3 Research Gap..12
1.4 Research Objectives 13
1.5 Significance of the Study 14
1.6 Scope and Limitations..16
1.1 Definition of Terms..17
CHAPTER 2: REVIEW OF RELATED LITERATURE 19
2.1 Hydroponics System 19
2.1.1 Vertical Farming..20
2.1.2 Nutrient Film Technique 20
2.2 Hydroponics Vegetables 24
2.2.1 Spinach 24
2.3 Growth Condition..25
2.3.1 Total Dissolved Solids..26
2.3.2 Light Requirement..26
2.3.3 pH Level..27
2.3.4 Temperature..27
2.4 Nutrient Solutions 30

2.5 Growing Sponge....31
2.6 Deep Learning....32
2.6.1 Convolutional Neural Network....32
2.6.1.1 Convolutional Neural Network for Plant Diseases....33
2.7 Python....36
2.8 USB Camera....36
2.9 Raspberry Pi....37
2.10 Wi-Fi Module....38
2.10.1 NodeMCU ESP8266....38
2.10.2 Google Firebase....39
CHAPTER 3: METHODOLOGY....41
3.1 Research Deign....41
3.1.1 Input / Process / Output....42
3.1.2 Block Diagram....43
3.2 Research Process Flow....45
3.2.1 Hardware Flowchart of the Automated Hydroponics System....47
3.2.1.1 Temperature Parameter Process Flowchart....48
3.2.1.2 pH Parameter Flowchart....49
3.2.1.3 Total Dissolve Sensor (TDS) Parameter Process Flowchart....50
3.2.1.4 Light Intensity Parameter Process Flowchart....51
3.2.2 Software Flowchart of the Automated Hydroponics System....52
3.3 Hardware Development....53
3.3.1 Materials and Equipment....54
3.3.1.1 PVC Pipes....54
3.3.1.2 DFRobot pH Sensor v1.1....55
3.3.1.3 Water Level Sensor v1.1....56
3.3.1.4 DFRobot Analog Total Dissolved Solid Sensor....57
3.3.1.5 DHT22....58

3.3.1.6 GY-30 BH1750FVI Module Digital Light Intensity Illumination Sensor 59
3.3.1.7 Arduino Mega 2560 R3 60
3.3.1.8 Actuators 61
3.3.1.8.1 12V DC Fan 61
3.3.1.8.2 Led Grow Lights 62
3.3.1.8.3 Peristaltic Pump 63
3.3.2 Hardware Construction 63
3.3.2.1 Greenhouse Design 63
3.3.2.2 Vertical Hydroponics System Design 64
3.3.3 Calibration of Sensors and Actuators 64
3.3.3.1 DHT22 Calibration 65
3.3.3.2 Light Intensity Sensor Calibration 65
3.3.3.3 pH Sensor (DFRobot Analog pH Sensor) Calibration 66
3.3.3.4 TDS Sensor Calibration 67
3.3.3.5 Water Level Sensor Calibration 67
3.3.4 Positioning of the Sensors and USB Cameras 68
3.3.5 Automations of the System 68
3.3.5.1 Actuators and Relay 70
3.4 Software Development 71
3.4.1 Development of the Android Mobile Application 72
3.4.1.1 Notification System Development 73
3.4.2 Collection of the Data Sets 74
3.4.3 Development of the Model for Plant Disease Detection 74
3.6 System Assessment 77
3.6.1 Comparison between Conventional Method and Hydroponics System 77
3.6.2 Testing of the Functionality 78
3.6.3 Testing of Reliability and Efficiency 78
3.7 Statistical Analysis 78

3.8 Research Instrument 80
3.9 Project Work Plan (Gantt Chart) 85
CHAPTER 4: RESULTS AND DISCUSSION 88
4.1 Project Technical Description 88
4.2 Project Structural Organization 90
4.2.1 The Greenhouse and Hydroponics Set-up 90
4.2.2 Mobile Application 93
4.3 Project Limitations and Capabilities 92
4.4 Project Evaluation 92
4.4.1 Accuracy of the trained model for Disease Detection 93
4.4.2 Disease Detection of Mobile Application 94
4.4.3 Air Temperature and Humidity 95
4.4.4 Light Intensity 97
4.4.5 pH Level 99
4.4.6 Total Dissolve Solid Sensor 101
4.4.7 Comparison of Spinach Growth between Hydroponics and Conventional Solid Farming 103
4.4.7.1 Length of the Spinach Plant 103
4.4.7.2 Height of the Spinach Plant 106
4.4.7.3 Number of Leaves of the Spinach Plant 108
4.4.7.4 Color of the Spinach Plant 110
4.4.8 User's Evaluation 111
CHAPTER 5: CONCLUSION AND RECCOMENDATIONS 113
5.1 Summary of Findings 113
5.2 Conclusion 114
5.3 Recommendations 115
References 117

LIST OF FIGURES

Figure 3.1.1 Input/ Process/ Output 42
Figure 3.1.2 Block Diagram 43
Figure 3.2 Research Process Flow 45

Figure 3.2.1 Hardware Flowchart of the Pi-Droponics System..47
Figure 3.2.1.1 Flowchart of the System Temperature Correction 48
Figure 3.2.1.2 Flowchart of the System pH Correction..49
Figure 3.2.1.3 Flowchart of the System TDS Correction 50
Figure 3.2.1.4 Flowchart of the System Light Intensity Correction 51
Figure 3.2.2 Software flowchart of the Pi-Droponics System 52
Figure 3.3.1.1 PVC Pipes 54
Figure 3.3.1.2 pH Sensor 55
Figure 3.3.1.3 Water Level Sensor 56
Figure 3.3.1.4 Total Dissolved Solid Sensor (TDS) 57
Figure 3.3.1.5 Air Temperature and Humidity Sensor 58
Figure 3.3.1.6 Light Intensity 59
Figure 3.3.1.7 Arduino Mega 2560 R3 60
Figure 3.3.1.8.1 DC Fan 61
Figure 3.3.1.8.2 Grow Lights..62
Figure 3.3.1.8.3 Peristaltic Pump 63
Figure 3.3.2.2 Hydroponics Design 64
Figure 3.3.2.2 Calibration of DHT22 Sensor 65
Figure 3.3.3.2 Calibration of Light Intensity Sensor 65
Figure 3.3.3.3 Calibration of pH Sensor 66
Figure 3.3.3.4 Calibration of TDS Sensor 67
Figure 3.3.3.5 Calibration of Water Level Sensor 67
Figure 3.3.4 USB Camera 68
Figure 3.3.5 Circuit Diagram of Sensors of Arduino Mega 2560 ESP8266 NodeMCU...69
Figure 3.3.5.1 Connection of Actuators to Relay 70
Figure 3.3.5.1.2 Schematic Diagram of Actuators and Relay 70
Figure 3.4 Network Architecture 68

Figure 3.4.1 MIT App Inverter 72
Figure 3.32 Notification System 73
Figure 3.4.3.1 Google Colab 74
Figure 3.4.3.2 Testing of Pre-trained Model for Disease Detection 75
Figure 3.4.3.3 Test, Train, and Validation of Datasets for Disease Detection 76
Figure 3.4.3.4 Naming 6 Classes for Disease Detection 77
Figure 3.9 Project Gantt Chart 85
Figure 4.2.1 The Greenhouse and Hydroponics System 88
Figure 4.2.2 The connection of Arduino Mega 2560 R3 to ESP8266 NodeMCU 89
Figure 4.2.3 The connection of Actuators to Relay 90
Figure 4.2.4 Connection of Cameras to Raspberry Pi 4 Model B 90
Figure 4.2.2 Pi-Droponics Mobile Application 91
Figure 4.4.1 Accuracy of the trained model for Disease Detection 93
Figure 4.4.2 Disease Detection of Mobile Application 94
Figure 4.4.3.1 Comparison between Digital Thermometer vs. DHT22 Sensor 95
Figure 4.4.4.1 Comparison between Light Intensity Meter vs. GY-30 BH1750FVI Module Digital Light Intensity Illumination Sensor 97
Figure 4.4.5.1 Comparison between pH Meter vs. DFRobot pH Sensor v1.1.................100
Figure 4.4.6.1 Comparison between TDS Meter vs. DFRobot Analog Total Dissolved Solid Sensor 102
Figure 4.4.7.1 Length of the Spinach in Conventional Method vs. Hydroponics System 104
Figure 4.4.7.4 Height of the Spinach in Conventional Method vs. Hydroponics System 107
Figure 4.4.7.6 Number of Leaves of Spinach in Conventional Method and Hydroponics System 109
Figure 4.4.7.7 Color of the Leaves of Spinach in Conventional Method and Hydroponics System 110

LIST OF TABLES

Table 2.1 Hydroponics System 23
Table 2.2 Hydroponics Vegetables...25
Table 2.3.1 Total Dissolved Solids 26
Table 2.3.2 Light Requirement 27

Table 2.3.3 pH Level 27
Table 2.3.4 Temperature 28
Table 2.3 Growth Conditions 29
Table 2.4 Nutrient Solution 30
Table 2.5 Growing Sponge 32
Table 2.6 Deep Learning 35
Table 2.7 Python...36
Table 2.8 USB Camera 37
Table 2.9 Raspberry Pi 38
Table 2.10 Wi-Fi Module 40
Table 3.7 Two-tailed t-test Formula 79
Table 3.8 Statistical Analysis for User's Evaluation 80
Table 4.4.3.1 Collected Data from Experiments using Digital Thermometer and DHT22 Sensor..95
Table 4.4.3.2 Statistical Comparison between Digital Thermometer vs. DHT22 Sensor 96
Table 4.4.4.1 Collected Data from Experiments using Light Intensity Meter and GY-30 BH1750FVI Module Digital Light Intensity Illumination Sensor....................................97
Table 4.4.4.2 Statistical Comparison between Light Intensity Meter vs. GY-30 BH1750FVI Module Digital Light Intensity Illumination Sensor 98
Table 4.4.5.1 Collected Data from Experiments using pH Meter and DFRobot pH Sensor v1.1..94

Table 4.4.5.2 Statistical Comparison between pH Meter vs. DFRobot pH Sensor v1.1 ...95
Table 4.4.6.1 Collected Data from Experiments using TDS Meter and DFRobot Analog Total Dissolved Solid Sensor 96
Table 4.4.6.2 Statistical Comparison between TDS Meter VS DFRobot Analog Total Dissolved Solid Sensor 97
Table 4.4.7.1 Comparison of Length of the Spinach in Conventional Method and Hydroponics System 103
Table 4.4.7.2 Statistical Comparison of Length of the Spinach in Conventional Method and Hydroponics System 105
Table 4.4.7.3 Comparison of Height of the Spinach in Conventional Method and Hydroponics System 106
Table 4.4.7.4 Statistical Comparison of Height of the Spinach in Conventional Method and Hydroponics System 106
Table 4.4.7.5 Number of Leaves of Spinach in Conventional Method and Hydroponics System 108
Table 4.4.7.6 Statistical Comparison of Number of Leaves of the Spinach in Conventional Method and Hydroponics System 108
Table 4.4.7.7 Color of the Leaves of Spinach in Conventional Method and Hydroponics System 110
Table 4.4.8 User's Evaluation from the Barangay Barangay786 Zone 86, Sta. Ana, Manila Residents 107

ANNEXES

ANNEX I Bill of Materials 124
ANNEX II Survey Forms 126
ANNEX III Pi-Droponics Codes 137
ANNEX IV Project Documentation....................161
ANNEX V User's Manual 170
ANNEX VI Manual for Duplication of Prototype 175
ANNEX VII Proponents' Information 184

CHAPTER 1

THE PROBLEM AND ITS BACKGROUND

Chapter 1 provides an overview of the study, including the introduction, background information, research gap, research objectives, the significance of the study, scope and limitations, and the definition of key terms. This chapter sets the foundation for the research and establishes its importance in addressing the identified research gap. It also outlines the specific objectives and clarifies the boundaries within which the study will operate.

1.1 Introduction

Agriculture cultivates land to produce food, medicines, and other products to sustain life. It has been associated with the production of basic food crops for decades. Agriculture and farming are not currently commercialized. However, as the economy grew, many more farming-related occupations were recognized as part of agriculture. Agricultural lands are being converted into buildings in modern society, resulting in a scarcity of farmlands on which to plant crops, and due to the economic, social, and a lot of opportunities, people continue to convey to cities to acquire. Nevertheless, urbanization shows major significant challenges. For 49% of Filipinos' livelihood in the city area, urbanization in the Philippines promptly grew. The anticipated increase of the percentage is 77% that will happen in 2030. Mostly the cities of Baguio, Cebu, Davao, and Metro Manila caused this much urbanization that more than 12 million populaces live inside crowded group with huge constructions of buildings. The current agriculture system is up to a huge task: in 2050, about 70% of food production needs to increase to meet the caloric needs of a population of 9.8 billion people. (Canlas, 2020).

According to Sandre (2013), for most plants, the loam is a better readily accessible flourishing method. However, soil can sometimes hinder plant growth due to disease-causing organisms and nematodes, inappropriate soil reactivity, unfavorable soil compaction, poor drainage, and erosion degradation. Hydroponics is a farming system wherein plants are raised in nutrient-rich water rather than soil. They were worrying about a growing human population on how to nourish them in a climate change that has led scientists to hope that hydroponic technology could help alleviate impending food shortages. Some devices will monitor the number of nutrients, pH, water temperature, and even the light the plants receive in today's hydroponic systems. The three basic types of hydroponic systems are the nutrition film technology, the Ebb and Flow System, and the Wick system.

Canlas (2020) stated that vertical farming in city centers on a wide scale has the potential to provide enough food in a sustainable manner to comfortably feed the whole human population for the foreseeable future; to restore significant swaths of land to their natural state, restoring ecological functions and services; to use the organic element of human and agricultural waste to generate energy through methane generation while reducing vermin populations (e.g., rats, cockroaches) in a safe and effective manner; to remediate black water by developing a much-needed new plan for water conservation; make use of empty and abandoned urban sites; to disrupt the disease-carrying agents' transmission cycle in a fecally-contaminated environment; to allow for year-round food production without crop waste owing to greenhouse effect or weather-related occurrences; to obviate the need for insecticides and herbicides on a big scale; to produce an environment that motivates sustainable urban life, advocating a good healthy lifestyle for the people that choose to live in urban cities; to give a meaningful role for agrochemical industries.

1.2 Background of the Study

Vertical farming is one of the most cutting-edge agricultural solutions for reducing land use. Growing plants and vegetables on steeply inclined surfaces are known as vertical gardening. Instead of producing vegetables or meals on a single level, this type of planting produces items in vertically stacked layers. This farming strategy yields increased crop production in the same growing space. The monitoring method is crucial in vertical farming since it allows you to keep track of how much nutrients you're giving your plants. (Lagomarsino, 2019).

Furthermore, image processing may be used to classify and detect plant diseases, critical for successful crop production. Picture pre-processing, image acquisition, feature extraction, disease detection, and classifying are necessary for disease detection utilizing image processing. Enhanced photos are clearer and more high-quality than the original.

Plant disease detection with an automated technique is advantageous because it decreases the workforce required to monitor large crop farms and discovers disease symptoms early. (Michael, G., Tay, F., & Then, Y., 2021). This section presents recent studies on hydroponics monitoring systems and crop disease detection.

The study of Siregar, B., Efendi, S., Pranoto, H., Ginting, R. B., Andayani, U., & Fahmi, F. last 2017, proposed a monitoring system that includes a pH sensor, electrical conductivity sensor, water temperature sensor, air temperature sensor, light sensor, GSM / GPRS, Open Garden Shield, and Open Garden Hydroponic, and Arduino Uno as the main board or microprocessor. Arduino and the designated sensor network were installed on a hydroponic plant container. The information used came directly from the sensor network. The sensor network will send data to Arduino. Then, using the GSM Shield, Arduino will send the data to the server that has been given. It was determined whether the plant's state was normal once the system received the values of all sensors showing crop conditions. The system also displayed a graph of each sensor, which was refreshed every 10 minutes.

A temperature and humidity monitoring in agricultural fields using sensors based on the CC3200 single chip. The camera is linked to the CC3200, which captures images and sends data via MMS to the farmers' cell phones via Wi-Fi. Sensors are used to monitor the condition of crop fields, which is critical. The MT9D111 camera sensor connects the camera to the

CC3200 camera booster pack via a PCB. This is used to capture real-time images of a specific field, which are then transmitted to the farmer via GPRS. This agricultural monitoring system is a reliable and effective method for taking corrective action. The improved system will benefit farmers because it is more efficient. It sends information to the farmer via MMS if the temperature and humidity of the air in an agricultural field fall outside of the ideal range. Implementing such a technology in the field has the potential to significantly accelerate crop harvesting and global production. This system can be improved in the future by incorporating modern techniques such as irrigation and solar power. (Prathibha, S., Hongal, A., & Jyothi, M., 2017).

The analysis of the color spaces RGB, HSV, CIELab, and YCbCr in the smart farm setup for the identification of lettuce growth stages. The color space in the chosen arrangement will be investigated using K-nearest neighbors through Euclidean distance. This technique for segmenting images is simple but efficient. K-Nearest Neighbor recognizes objects based on the learning data that is closest to the object. Using a supervised learning algorithm (k), KNN is a technique that combines the results of the calculations with the training data in the selected range to provide predictions of classification by the majority. Next, the Euclidean distances between those pixels and each color marker were calculated using the designated color space. When the distance is shortest, the pixel most closely resembles that color marker. The lettuce growth stage was identified using both top-performing color spaces, CIELAB and YCbCr. 30 pictures of lettuce were utilized in the experiment, with 10 pictures representing sowing, 10 pictures representing vegetative growth, and 10 pictures representing harvest. The method has an average accuracy of up to 90% for CIELab. The CIELAB color space has the most influence on distinguishing the background from the lettuce image, according to the study's findings. Using the CIELab color space on the smart farm setup, a high level of accuracy was reached in predicting lettuce growth. (*Color Space Analysis Using KNN for Lettuce Crop Stages Identification in Smart Farm Setup, 2018).*

The study of Patil, Uttekar, & Suryawanshi in 2020 demonstrated a Node MCU-based hydroponic system. The system's most notable characteristics are a water-driven agriculture system that eliminates soil needs. The crops are supplied with water and nutrients in this automated hydroponic system based on sensor feedback such as temperature and humidity (DHT 11), pH sensors, and electric conductivity circuits. All the sensors were connected to an open-source Node-MCU (ESP12) microcontroller. A 3.3 V power supply is also connected to this microcontroller. The Node-MCU controls the valves, pumps, and dispensers to ensure the system runs smoothly. All this data is delivered to a smartphone application (MQTT). The developed system proved successful in cost-effectively producing plants/crops by minimizing the amount of water, nutrients, and farming space required. The goal was to capture all data related to the working system so that the IoT could monitor and regulate it.

The data collected by the developed system is uploaded to Thing Speak and presented on the "IoT MQTT Panel" mobile application, which allows the user to easily monitor and control the state of the plants.

The overall purpose of this study is to construct a temperature-controlled greenhouse and plant growth monitoring system for smart aquaponics using an Android IoT application. The goal of this study was to develop a Python program on the Raspberry Pi that uses image processing to monitor and assess the plant's surface area, as well as compare the growth of plants (surface area of the leaf) grown in traditional soil-based farming to the proposed smart aquaponics system using the developed Python-based image processing program. Each plant's growth is carefully captured using a web camera connected to the Raspberry Pi module. Image processing of the collected images is used to determine the plant surface areas. To evaluate the correctness of the Python Image Processing Program for the monitoring system of the plant growth, a software provides a precise, free, and fast tool that is made for the comparison for estimating leaf area from digital images. The Python image processing tool was compared to the easy demonstration software. Between the average, there are no significant differences statistically between the surface area acquired from the Python image processing program and the Easy Leaf Area software because the value is greater than 0.01. This shows that the Python image processing program is accurate enough. (Tolentino, 2020).

The study of Pratama, Wahab, & Alaydrus (n.d.) compared the performance by dividing the ratio of training and validation dataset into 3 groups, i.e., 78/9, 70/17, and 61/26 with the standard testing ratio for all categories being 13 percent, and used Deep Learning as objection detection to spot the disease in Hydroponic vegetables. For 3-dimensional object detection, segmentation, and other applications, faster R-CNN is often applied. Feature Network, Regional Proposal Network, and Detection Network are the three neural networks that makeup Faster R-CNN in concept. Convolutional networks' complexity can be simplified using the module Inception V2. This module allows the convolution network to grow deeper and wider. From a dataset of 412 images with 53 testing images and a default learning rate setting of 0.0002, the results of this study show that a ratio of 78/9 has a better performance with Accuracy of 70 percent, Precision of 97 percent, recall of 68 percent, and F1 Score 80 percent, while ratio 61/26 has the worst performance with Accuracy 40 percent, Precision 24 percent, recall 100 percent, and F1 Score 38.5 percent. As a result, a set of images associated with ailments discovered in hydroponic lettuce were used to train the Faster RCNN Inception V2 algorithms, which were selected as efficient deep learning algorithms.

Bala Murugan et al. presented work in 2021 that uses the Internet of Things and deep learning to integrate agricultural management with leaf disease diagnosis. This research analyzes agricultural field metrics in a distant cloud environment using the Internet of Things and remote sensing.

To construct a smart illness prediction, a cloud-deployed modified Resnet model was used. The Internet of Things Smart Monitoring System is intended to monitor plant development factors such as light intensity, soil moisture, and other vital environmental variables automatically. The Bolt IoT Wi-Fi module, Sensors, and the Cloud are the system's main components. This model can forecast soil moisture and light levels in the future. From basic water need pattern analysis to complicated ensemble weather forecast models, this has a wide range of uses. This may be linked to the model for identifying plant ailments, and an application can be developed to incorporate these components into a comprehensive smart agricultural and irrigation system. (Murugan, M., Rajagopal, M., & Roy, D., 2021).

The study of Yang, R., Wu, Z., Wentai, F., Zhang, H., Wang, W., Fu, L., Majeed, Y., Li, R., & Cui, Y. last 2023 demonstrate the viability of using machine learning models to identify yellow and decaying lettuce leaves, i.e., K-Nearest Neighbor (KNN), Multiple Linear Regression (MLR), and Support Vector Machines (SVM). One-way analysis of variance was employed to lessen the presence of RGB, HSV, and L*a*b* features in images of lettuce grown hydroponically. An image of lettuce grown hydroponically was segmented using the techniques of image binarization, image masking, and image filling in order to evaluate several models. The performance of KNN, MLR, and SVM was evaluated in terms of accuracy and detection speed. The measurement for the detection accuracy was 1 less than the result of the error calculation, which was the absolute value of subtracting the ratio of the abnormal area acquired from the ground truth from the ratio of the abnormal area predicted by KNN, MLR, and SVM. KNN took significantly longer than MLR (0.61 s) and SVM (1.98 s) to recognize a 3 024 4032-pixel image, taking roughly 20.25 seconds. In comparison to SVM's 98.33 percent and 97.91 percent, respectively; MLR achieved detection accuracies of 89.48 percent for yellow leaves and 99.29 percent for rotten leaves. With just one camera and hydroponic lettuce, SVM was able to detect abnormal leaves. As a result, machine learning methods might be able to spot abnormal hydroponic lettuce leaves.

The development of an automated system for a nutrient film technique (NFT) hydroponic system in 2021. The recommended system controls and monitors the pH level, electrical conductivity (EC), dissolved oxygen (DO), water temperature, water flow rate, and water level of the nutrient solution in the Hydroponics nutrient solution irrigation system suited for a given plant. The core system consists of sensors, microcontrollers, actuators, and data recorders. It was validated by the sensor and actuator testing, as well as a field trial. The system incorporates a touch screen LCD for monitoring nutritional environment factors and manually encoding the crop's pH, EC, DO, and water temperature needs for the hydroponic system. All identified nutritional environment factors are recorded on a MicroSD card by the data logger module. The experiment findings revealed that the developed system is capable of monitoring and adjusting the nutrient solution parameters in a hydroponic setting. (Kuncoro, C., Asyikin, M., & Amaris, A.,2021).

In 2021, Tolentino et al. developed a vertical farming smart hydroponics system with LoRaWAN based WSN for data monitoring transfer. The system is self-contained and regulates the nutrients, water, light, and air that the plants require without the need for human intervention. Acquiring the exact data in real-time was done using the Long-Range Wide Area Network (LoRaWAN) technology, which transmits vital lettuce parameters across the internet to the Application device. Metrics were monitored using this Wireless Sensor Network technology regardless of the distance from the system. This technology is beneficial for Internet of Things (IoT) applications such as plant growth monitoring and autonomous nutrient supply. As a result, the user will be able to monitor the smart hydroponics via the IoT gateway from any location and at any time.

The study of Arshad, J., Aziz, M., Al-Huqail, A., Zaman, M., Husnain, M., Rehman, A., & Shafiq, M. last 2022 created GPS-based LoRaWAN sensor nodes in 2022 that can provide real-time data to a central server/cloud via an IoT gateway. The application's architecture allows users to save and retrieve data at any time from any remote place. Crop yield data may be gathered through monitoring, which assists in discovering the root reason for poor yield output in agriculture. The sensor system, which comprises soil moisture, humidity, NPK, temperature, and pH sensors, provides data to the cloud through the internet. Users may monitor real-time data from any place and at any time using an Android application. The findings and comparison verify the originality of the concept by merging smart irrigation with smart control and smart decision-making based on reliable real-time field data. Because it delivers data using LoRa, an open-source communication technology with a transmission range of several kilometers, it outperforms conventional systems. The sensor nodes contributed to increased crop output, resulting in the accomplishment of inclusive and sustainable economic goals.

According to previous studies, majority of these systems do not include all the sensors required for controlling and monitoring the nutrient solution in a hydroponics system. In some experiments, real-time data transmission to a central server or cloud via an IoT gateway or access to the data through monitoring applications has been accomplished using technologies including LoRaWAN, MMS, and GPRS. In the hydroponics studies found, there are no incorporated disease detection and notification systems. Similar studies of disease detection and notification systems have been found but do not necessarily utilize cameras to capture photos of a specific leafy vegetable that was planted in order to detect disease, instead they relied on readings from sensor modules. Although some studies used cameras, they were only for status checking to identify whether the plants are ready for harvest or not, and these studies do not have a notification system for the said purpose. As a result, the proponents will conduct a study that will include a notification system for monitoring and control as well as for disease detection and harvesting.

1.3 Research Gap

An optimal hydroponics system can regulate plant illumination and temperature while monitoring and adjusting air temperature, light intensity, TDS, and pH levels. Previous studies include a monitoring system for automated hydroponics systems that could be accessed via mobile application. However, only studies of standard farming applying machine learning algorithms to detect plant diseases and plant maturity are existing. And all-important parameters needed in hydroponics farming were not usually being considered from the previous studies. Furthermore, there are still no studies on hydroponics systems with plant disease detection that utilizes machine learning. In this study, a camera module will be incorporated into the system and be utilized for machine learning using the convolutional neural network algorithm. This study will focus on building a hydroponics system with all the necessary sensors, as well as developing a mobile application that will serve as the notification system capable of sending notification to the user whenever the sensor reading falls out of threshold range, whenever a plant becomes unhealthy as a result of a disease observed and whenever the plants are deemed harvestable because they have reached the required days for maturity. The edge of this study over the previous hydroponics studies is that the researchers will implement a mobile application for monitoring of sensors as well as the notification system for the plant disease detection and harvesting.

1.4 Research Objectives

The general objective of this research is to develop an automated vertical hydroponics system with monitoring and control that will include a real-time disease detection system for leafy plants and vegetables and a notification system that will send information whenever the system is out of specification parameters, as well as the plants that have detected disease and if it is ready to harvest.

Specifically, it aims to:

1. To construct and develop a circuit with integrated actuators and sensors, including a light intensity sensor, total dissolved solids sensor, pH sensor, temperature sensor, humidity sensor, water level sensor, and camera module, using the Raspberry Pi IV model B as a sensor node, to control and monitor plant growth factors in an outdoor vertical hydroponics system.
2. To have a notification system that sends information to the user whenever the system is out of specification parameters, as well as for plants that have detected disease and for plants that are ready to harvest based on the estimated number of days required for maturity.
3. To develop a model for disease detection using convolutional neural networks, as well as an android mobile application for monitoring.
4. To test the functionality, reliability, and efficiency of the system.

1.5 Significance of the Study

The significance of this study is to develop a vertical hydroponics system with image processing as a notification that will monitor the growth of the plants and notify the user when the plant is ready to harvest or not with the help of the mobile application. This study could significantly contribute to the following Sustainable Development Goals (SDG) of the United Nations: (2) Zero Hunger, (8) Decent Work and Economic Growth, and (9) Industry, Innovation, and Infrastructure.

Furthermore, this study aims to contribute to Section III: Agriculture, Aquatic and Natural Resources (AANR) under Annex 7 for Coconuts and Annex 13 for Vegetables. The AANR encourages the use of innovative and emerging technologies, comparable to our research on electronics and automation, with the goal of developing a product that will have a significant impact on the industry.

This study represents the application of electronics engineering to the environment. The researchers wanted to create and include something that could benefit the environment in order to reduce or limit the non-perishable waste of the country. In the environment, the coconut husk is a sufficiently eco-friendly product with a positive impact on the environment, as it will never cause environmental damage and is, therefore, free of plant irritation. It is a planting medium that helps plants retain moisture and resist fungal growth. The Urban Farmers / People, will allow the urban farmers to help solve their problems without excessive use of space or water and produce leafy vegetables with high nutrient content, and produce vegetables faster than traditional or conventional growing methods.

Additionally, the community will gain from the study in terms of solving their concerns with planting without wasting too much space or water. It takes less time to cultivate the crops and does not require soil. In the future, this form of agriculture will be a primary source of higher-quality food items. The proposed project will allow the researchers to put their expertise and abilities to work in the development of technology that will benefit a lot of people as well as the environment. Thus, the researchers want to present, share and provide additional knowledge about the study for the forthcoming fellow researchers to be a contribution to future references.

1.6 Scope and Limitations

The aim of the research is to create a completely automated vertical hydroponics system using the Convolutional Neural Network algorithm. The project's control system can alter critical parameters for plant growth, disease detection, and notification system. The device has five sensors: a pH sensor, a water level sensor, a TDS sensor, an air temperature and humidity sensor, and a light intensity sensor. The required number of hours per day can be met by setting the grow lights to switch on and off at specific periods during the day. The study was restricted based on the suggestion of the barangay chairman to meet the needs of the community in Sta. Manila. In the discussion, the research study will focus on three (3) leafy vegetables since different plants have various growing requirements and need to satisfy the nutrient requirements for maximum production, which differ from what is required for building a large, tight leafy heart. The researchers decided to plant the leafy vegetable one at a time since growing two different plants side by side or in the same system can be complicated in hydroponics. In addition, the mobile application of the research project is compatible up to Android version 13.

1.7 Definition of Terms

For the purpose of clarification, the following were the definition of words used in the paper:

- Nutrient Film Technique - is a hydroponic technique in which a very shallow stream of water containing all of the dissolved nutrients essential for plant growth is re-circulated past the bare roots of plants in a watertight gully, also known as channels. (DBpedia, n.d.).
- Alternaria Leaf Spot Disease - is a foliar disease that affects brassica crops and is caused by the fungus *Alternaria brassicicola*. Many brassica crops, including cabbage, cauliflower, kale, brussels sprouts, and broccoli, can be affected by the disease. Even minor infections can result in an unmarketable crop. (Cornell, n.d.).
- Anthracnose Disease - Anthracnose is a broad term for a group of related fungal diseases that cause dark lesions on

leaves. It can also cause sunken lesions and cankers on twigs and stems in severe cases. (UCIPM, n.d.).

- Downy Mildew Disease - is a plant disease caused by a fungus-like (Oomycete) organism. Airborne spores spread it from plant to plant. It is a wet weather disease because infection is aided by prolonged leaf wetness. Downy mildews typically have limited host ranges that include only a few closely related plants (UCIPM, n.d.).
- White Rust Disease - initially, white rust symptoms appear as white, blister-like pustules on the lower surface of leaves. A yellowish spot may appear opposite these pustules on the upper surface of the leaf. (Bugwoodwiki, n.d.).

CHAPTER 2

REVIEW OF RELATED LITERATURE AND STUDIES

Chapter 2 focuses on providing a thorough review of related literature, presenting a comprehensive analysis of existing research, studies, and scholarly works directly relevant to the topic of the study. This chapter establishes a theoretical framework and contextual understanding by critically evaluating and synthesizing various authoritative sources' findings, theories, and concepts. This chapter contributes to the existing knowledge base through the review and sets the stage for the subsequent research chapters.

2.1 Hydroponics System

Hydroponics is a practice of growing in nutrient solutions even without the assistance of a physical support medium such as gravel, vermiculite, Rockwool, peat moss, saw dust, coir dust, or coconut fiber. Most hydroponic systems were already automated to regulate the amount of water, nutrients, and illumination based on the needs of various plants. (Pramono, 2020).

Soilless system developments, such as aquaponics and hydroponics, can assist farmers save money on resources. Gashgari et al. previously investigated the difference between growing plants in a hydroponics system and a soil-based system. In terms of plant height growth, the findings of this study demonstrated that hydroponic planting outperforms traditional soil planting. The planting approach, on the other hand, has no influence on leaf length. Furthermore, seed type or the connection between seed type and planting technique have little effect on plant development. (Gashgari, n.d.).

2.1.1 Vertical Farming System

Vertical farming systems (VFS) have been suggested as an engineering approach for increasing agricultural productivity per unit area of Vertical farming systems (VFS) have been suggested as an engineering approach for increasing agricultural productivity per unit area of cropland by expanding crop yields vertically. Touliatos et al. compared a vertical farming system to a standard horizontal hydroponic system (HHS) using lettuce as a model crop to see if this strategy could be a viable alternative to horizontal crop production systems. It was concluded that vertical column-based VFS offers a potential alternative to traditional horizontal growth systems by maximizing the utilization of growing space and yielding more crops per unit area. To counteract the observed PPFD gradient, additional yield increases could be achieved by adding artificial lighting within the VFS. (Touliatos, D., Dodd, I., & McAinsh, M., 2016).

2.1.2 Nutrient Film Technique

Dr. Alan Cooper invented NFT in the mid-1960s in England to overcome the inadequacies of the ebb and flow system, according to Domingues et al. Water or a nutrient solution circulates throughout the system and enters the growing tray via a water pump that does not have a timer. The system is sloped slightly to allow the nutritional solution to flow through the roots and back into the reservoir. Plants with roots floating in a hydroponic solution are placed in a channel or tube. Because roots are constantly submerged or nutrients, they are prone to fungal infestation. (DBpedia, n.d.).

Puno et al. discussed the design and implementation of a lettuce hydroponics system using nutrient film technology. The design comprises a rack (24" x 48" x 72") with three layers of three inch-diameter polyvinyl chloride (PVC) pipes that will be utilized as gully or channel. The PVC pipes are spaced 4.5 inches apart. The system's reservoir is positioned on the bottom tier of the rack, as is the pump. The flow meter and the valve are connected to the pump via a flexible hose. The water flow from the nutrient tank will be controlled by the valve. In the nutrient tank, there will be two inlets and one output pipe that can be controlled. The nutrient concentrate tank and freshwater tank have inlet pipes, while the nutrient tank drain has an outlet pipe. One PVC pipe contains 15 slots with a hole diameter of 2 inches and a spacing of 6 inches from center to center. A total of 135 spaces will be available over the entire arrangement. For LED light support, an extension will be made to the sides of the rack. The range of the parameters was maintained using fuzzy logic control. The fuzzy logic will manage the pumping of fresh water and nutrient concentration reservoir, as well as the drain of the mixing tank, using data from the sensors that measure electrical conductivity, pH, and the water level in the mixing tank. PVC pipes are successfully used as a conduit in the nutrient film technique hydroponics system. This resulted in the creation of an automated NFT hydroponics system utilizing fuzzy logic. (Carlo et al., 2020).

Table 2.1 Hydroponics System

Author	Year	Title	Relevant Finding	Significance to our Study
Sharma, N. et al.	2018	Hydroponics as an advanced technique for vegetable production: An overview	Discussed the hydroponic constructions including the functions of the wick, ebb and flow, drip, deep water culture, and Nutrient Film Technique (NFT) systems, as well as their advantages and disadvantages.	Helped in further understanding the various hydroponics techniques and establish the type of hydroponics system to be used.
Gashgari, R. et al.	2018	Comparison between Growing Plants in Hydroponic System and Soil Based System	The hydroponic planting system outperforms the traditional soil approach by allowing plants to grow to their full height faster. The planting strategy, on the other hand, has no effect on the length of the leaves. Furthermore, plant growth is unaffected by seed type or the interplay between seed type and planting system.	It assisted Pi-Droponics in establishing that hydroponic planting has an advantage over soil-based planting and is appropriate for urban farmers today.
Touliatos, et al.	2016	Vertical farming increases lettuce yield per unit area compared to conventional horizontal hydroponics	Proved that the vertical farming system is a viable alternative to conventional horizontal growth systems by optimizing growing space use efficiency, thereby producing more crops per unit area.	The vertical farming system will be implemented by the Pi-Droponics study. This study showed that vertical farming is more efficient than horizontal farming.
Touliatos, et al.	2016	Vertical farming increases lettuce yield per unit area compared to conventional horizontal hydroponics	Proved that the vertical farming system is a viable alternative to conventional horizontal growth systems by optimizing growing space use efficiency, thereby producing more crops per unit area.	The vertical farming system will be implemented by the Pi-Droponics study. This study showed that vertical farming is more efficient than horizontal farming.
Puno, J. et al.	2020	Design of A Nutrient Film Technique Hydroponics System with Fuzzy Logic Control	A fuzzy logic-based automated NFT hydroponics system was successfully created.	Pi-Droponics will use the Nutrient Film Technique (NFT) method for the hydroponics system. This study helped in conceptualizing the design of the system and proper positioning of important materials and equipment.

2.2 Hydroponics Vegetables

A hydroponics system can grow a vast number of plants, crops, and vegetables. The resulting products' quality, flavor, and nutritional content are often superior to those grown in natural soil. Leafy greens (lettuce, spinach, parsley, celery, etc.) can be grown successfully and easily in hydroponic systems, according to several experimental findings. Because of their increased growth and nutrient uptake capacity, lettuce and spinach are the most promising species to grow in integrated hydroponics and aquaculture systems. (DBpedia, n.d.).

2.2.1 Spinach

Green vegetables, particularly spinach plants, were studied since they are high in nutrients but have a short lifespan. The storage life of spinach leaves could be prolonged to a maximum of 5-7 days under regular storage circumstances. (Aiswarya et al. 2019). Because of its increased growth and nutrient uptake capacity, spinach is one of the most desirable species for integrated hydroponics and aquaculture systems. Ranawade et al. examined spinach yields in hydroponic, aquaponic, and traditional systems using perlite (aquaponics) and sphagnum moss (hydroponics). Aquaponically grown spinach had a slightly higher yield than hydroponically grown spinach. (DBpedia, n.d.).

Table 2.2 Hydroponics Vegetables

Author	Year	Title	Relevant Finding	Significance to our Study
Engeseth, J. et al	2021	Comparison of growth characteristics, functional qualities, and texture of hydroponically grown and soil grown lettuce	This is the first thorough side-by-side growth study to show that lettuce grown hydroponically is different from lettuce grown in soil.	A hydroponics system may grow a wide variety of plants, crops, and vegetables, according to this research. It was determined that the quality, flavor, and the nutritional content of hydroponics goods are frequently superior to those grown in normal soil.
Saha, S. et al.	2016	Growth, yield, plant quality and nutrition of basil (Ocimum basilicum L.) under soilless agricultural systems	The study wants to see the growth, yield, plant quality and nutrition of basil under soilless agricultural system	Pi-Droponics will use Basil as its leafy vegetables that have a fast growth in the hydroponics system.

2.3 Growth Conditions

Nutrient management is a crucial phase in hydroponics. The four major criteria to focus on for nutrition management in soilless culture are total salt concentration, pH, alkalinity, and nutrient concentration ratio. (Maboko, 2019).

2.3.1 Total Dissolved Solids

Total Dissolved Solids (TDS) refers to the strength of a nutrient solution. It is measured in MicroSiemens (mS or S) and expressed in Parts Per Million (PPM) or Electrical Conductivity (EC). PPM is commonly used in the US, while EC is more common globally. Although PPM and EC meters may display the readouts differently, both methods accurately measure nutrient strength. PPM and EC meters measure the flow of electrical current between two metallic probes to measure nutrient strength. As the salt concentration increases, the nutrient solution becomes more conductive. (Singh, 2016).

Table 2.3.1 Total Dissolved Solids

Parameter	Spinach
TDS (ppm)	1260 - 1610

2.3.2 Light Requirement

One of the things that affect hydroponic development is sunlight. Sunlight contains a spectrum of colors that can help hydroponic plants thrive by stimulating their photosynthetic process. However, nutrients for hydroponic plants from illumination may be inadequate due to some unfavorable climatic conditions. Prasetia et al. devised a solution to tackle this problem, in which the sun's nutrients are substituted by light-emitting diode (LED) grow lamps that provide a color spectrum similar to indoor sunshine. (Prasetia, Y., Putrada, A., & Rakhmatsyah, A., 2021).

Table 2.3.2 Light Requirement

Parameter	Spinach
Light Requirement (no. of hours)	8

2.3.3 pH level

A solution's pH reveals how acidic or basic it is at the time of measurement. The scale runs from zero to fourteen, with seven signifying neutral. The pH of a nutritional solution must be kept within a certain range because it affects nutrient availability. (Singh, 2016).

Table 2.3.3 pH Level

Parameter	Spinach
pH level	6.0 - 7.0

2.3.4 Temperature

While air temperature is one of the most important environmental factors for influencing secondary metabolism in plants for production, many plants' growth, and chemical composition are also influenced by the temperature at the root zone. One of the advantages of hydroponic gardening is the ability to regulate the temperature of the nutrient solution around the root system by utilizing heaters or cooling spirals to raise or lower the temperature. (Al-Rawahy et al., 2018).

Table 2.3.4 Temperature

Parameter	Spinach
Temperature (° C)	21

Table 2.3 Growth Conditions

Author	Year	Title	Relevant Finding	Significance to our study
Singh, H. et al.	2016	Electrical Conductivity and pH Guide for Hydroponics	Every day, the pH and EC value of the hydroponics system is manually checked using a Combined EC and pH meter to see if the amount of pH and EC received by the plants is accurate.	The pH and EC sensor were used to check the pH and EC value in the hydroponics system.
Prasetia, Y. et al.	2021	Evaluation of IoT Based Grow Light Automation on Hydroponic Plant Growth	This study focused on the impact of LED grow light automation using RTC modules and relays on plant growth rates in a hydroponics system.	The project will implement a LED grow light to help the growth rate of the plants.
Al-Rawahy, M. et al.	2018	Effect of Cooling Root-Zone Temperature on Growth, Yield and Nutrient Uptake in Cucumber Grown in Hydroponic System During Summer Season in Cooled Greenhouse	The root-zone temperature in the hydroponics system affects the growth rate of the plants. The higher the temperature inside the greenhouse the lower the production of the plants yield.	Pi-Droponics will maintain the root-zone temperature within the hydroponics system to enhance the plant growth.

2.4 Nutrient Solutions

An aqueous fertilizer solution for hydroponic systems contains mostly inorganic ions from soluble salts of necessary elements for higher plants. The absence of a crucial element prevents the whole plant life cycle from taking place. Carbon, hydrogen, oxygen, nitrogen, phosphorus, potassium, calcium, magnesium, sulfur, iron, copper, zinc, manganese, molybdenum, boron, chlorine, and nickel are currently considered essential for most plants. The essential components are received from the growth media, except for carbon (C) and oxygen (O), which are provided by the atmosphere. Other elements, such as sodium, silicon, vanadium, selenium, cobalt, aluminum, and iodine, are useful because they can accelerate growth, compensate for hazardous effects, or substitute necessary nutrients in a less defined capacity. Only nitrogen, phosphorus, potassium, calcium, magnesium, and sulfur are included in the most basic nutrient solutions, which are augmented with micronutrients. The electrical conductivity and osmotic potential of a solution are determined by the nutritional makeup. (Pebriana, 2022).

Table 2.4 Nutrient Solutions

Author	Year	Title	Relevant Finding	Significance to our study
Asao, Toshiki	2012	Hydroponics: A Standard Methodology for Plant Biological Researches	According to this book, if important parameters such as temperature, pH, EC, and humidity can be controlled, the plant will increase crop yield and improve crop quality.	Controlling the settings will assist the Pi-Droponics in growing plants in good condition and improve crop quality.

2.5 Growing Sponge

A good planting medium has balanced nutrients for growth, the ability to hold water and nutrients, good aeration and drainage, the ability to keep moisture around the roots, is pest and disease free and is not susceptible. Plant growth is best in porous media because it allows for more flexible water and air movement. The porosity of the media is the amount of open space or pores that allow air and water to circulate through the plant root system. According to studies, sponges have higher porosity and aeration than coco peat and rice husk. Because the sponge hydroponics growing medium is inexpensive and readily available, it is an excellent hydroponic growing medium. Because it is lightweight and easy to work with, a sponge is an ideal hydroponic growing media. (Pebriana, 2022).

This research aimed to ascertain the impact of growing media on the development and production of leafy vegetables grown hydroponically using the Nutrient Film Technique (NFT). This study used a factorial randomized block design (RBD) with two components as its experimental strategy. The first factors were growing media, specifically cocopeat, sponge, and perlite. The second factor was the 30-day harvest of certain crop types, such as lettuce and pakchoi. In the NFT hydroponics system, plants cultivated in cocopeat had the highest plant yield (12.55 g). Plants planted in sponges had the longest plant shoot height (9.69 cm), whereas plants cultivated in perlite had the shortest (8.85 cm). Compared to plants produced in a sponge (4.93 cm) and perlite (4.32 cm) growing media, plants grown in cocopeat (5.54 cm) had the widest plant leaf width. The results of this study demonstrated that growing media such as cocopeat and sponge outperformed perlite. The effects on growth and yield characteristics were negligible when the two components were combined. For optimal growth and yield when growing lettuce and pakchoi hydroponically, cocopeat should be used as the first growing medium, followed by sponges. (Chhetri, S., Dulal, S., Subba, S., & Gurung, K., 2022).

Table 2.5 Growing Sponge

Author	Year	Title	Relevant Finding	Significance to our study
Lestari, YP. et al.	2022	Response of Liquid Organic Fertilizer and Type of Media on Bokchoy (Brassica Rapal.) Production by Wick Hydroponic	The sponge has a higher porosity, which is the proportion of empty space (pores) that is sufficient for air and water circulation around the plant root system.	It will help our research in promoting the growth of lignins through the use of sponge, which is an excellent medium for hydroponic system planting.

2.6 Deep Learning

According to LeCun (2015) deep learning allows for data representations to be learned at various abstraction levels by computational models comprising several processing layers. These technologies have significantly advanced the state-of-the-art in voice recognition, visual object identification, object detection, and various other fields such as drug development and genomics. Deep learning identifies detailed structures in large data sets by utilizing the backpropagation technique to determine how a machine's internal parameters that are used to calculate each layer's representation from the previous layer's representation should be modified.

2.6.1 Convolutional Neural Network

The Convolutional Neural Network is a widely known deep neural network (CNN). It gets its name from the mathematical linear action between matrices known as convolution. CNN comprises several layers, including a convolutional layer, a non-linearity layer, a pooling layer, and a fully connected layer. (Albawi, 2017).

2.6.1.1 Convolutional Neural Network for Plant Diseases

Plant disease detection is essential for preventing agricultural product production and quantity losses. Infectious agents such as fungi, bacteria, and viruses are commonly responsible for plant illnesses. Plant disease signs are obvious indications of infection, while symptoms are the visible repercussions of these diseases. Suresh et al. presented a system with TFLite that includes an end-to-end Android application. The proposed system was chosen to create a plant disease detection Android application. Using a Convolutional Neural Network, it has techniques and models to distinguish species and diseases in crop leaves. To modify source code, the suggested system employs Collab. Training and testing facilities are made up of two types. One is in laboratory settings, while the other is in the field. Because the lighting circumstances and background attributes of the photos are so dissimilar when the samples are taken in the field, the model may provide very low accuracy compared to the accuracy values obtained in the lab. To counteract this, various images were used during the training phase (heterogeneity). Overall, this work is created from scratch and produces reasonable accuracy. (Musa et al, 2022).

A study by Pratama et al. attempted to detect the quality of lettuce hydroponic plants by using disease characteristics in Lettuce plants as plant health indicators. The researchers gathered photos of diseased lettuce (Lactuca sativa) hydroponic plants and utilized them for training the object detection model. These photographs were gathered using crawling techniques from search engine sites using the keywords "hydroponic lettuce diseases," yielding 873 images. Then, any photos with duplicates or unsuitable were removed, leaving only images with lettuce illness images. Consequently, 412 photos were chosen to form the study's dataset. They then used the Labeling program to annotate each image in XML files with the tag "unhealthy lettuce." The dataset was then segmented into training, validation, and testing ratios. The training and validation ratio were split into three groups: Model A (78%) is the most popular, followed by Model B (70%) and Model C (61/26). For the testing ratio the standard testing ratio for all categories was 13 percent or 53 photos. The Faster RCNN Inception V2 and YOLO algorithms were recognized as viable deep learning algorithms in this work, and both were trained using a dataset including images relevant to diseases found in hydroponic lettuce. This is highly advantageous for producers and farmers in identifying good Lettuce goods to deliver to consumers, as well as consumers benefiting from the products' quality. (Pratama et al, 2020).

Table 2.6 Deep Learning

Author	Year	Title	Relevant Finding	Significance to our study
LeCun, Y. et al.	2015	Deep Learning	It allows computational models composed of multiple processing layers to learn data representations with various levels of abstraction.	Deep Learning will be used for image processing using Pi-Droponics machine learning.
Albawi, S. et al.	2017	Understanding of a Convolutional Neural Network	CNN is regarded as a powerful machine learning tool for a variety of applications such as face detection and image processing.	The CNN will be useful in detecting disease and processing images of various variant leafy vegetables.
Suresh, V. et al.	2020	Plant Disease Detection using Image Processing	The proposed system includes an end-to-end Android application built with TFLite, as well as an Android application that detects plant diseases. It employs Convolutional Neural Network algorithms and models to recognize species and diseases in crop leaves.	The system will use image processing and machine learning techniques, as well as a mobile application for monitoring and detecting leafy vegetable disease.
Pratama, I. et al.	2020	Deep Learning for Assessing Unhealthy Lettuce Hydroponic Using Convolutional Neural Network based on Faster R-CNN with Inception V2	Hydroponic project that uses Convolutional Neural Network based on Faster R-CNN for analyzing unhealthy or diseased lettuce and uses 3-dimensional objects, segmentation, and other techniques.	This will support its use of Convolutional Neural Networks to assess leafy plant disease.
Reyes-Yanes, A. et al.	2020	Real-time growth rate and fresh weight estimation for little gem romaine lettuce in aquaponic grow beds	It uses aquaponics for real time growth.	For the real-time growth rate of the leafy vegetables, the system employs image-processing and machine learning techniques.

2.7 Python

Python is a high-level object-oriented programming language with dynamic semantics being interpreted. Its high-level built-in data structures, dynamic typing, and dynamic binding make it particularly interesting to use as a scripting or glue language to connect existing components. It is a programming language used to create websites and applications, as well as to automate processes and do data analysis. Because of its flexibility and beginner-friendliness, it has become one of the most widely used programming languages. It's typically used to create the back end of a website or app bits that users don't see. Python's involvement in web development includes sending data to and from servers, processing data and interacting with databases, URL routing, and guaranteeing security. Python has several web development frameworks. (*Python, 2019).*

Table 2.7 Python

Author	Year	Title	Relevant Finding	Significance to our study
n.a.	n.d.	What is Python? Executive Summary	This article defined Python, its purpose and the advantages of using it.	Python will serve as the programming language of this study.

2.8 USB Camera

A USB camera connects to a computer via a USB cable in order to operate. The camera feeds are sent to the computer, where a software program enables you to see the images and upload them to the internet. Using the FTP, the program you use can be configured to display photos at predetermined intervals (File Transfer Protocol). Additionally, it can be configured to offer a live broadcast. In low-cost USB cameras can be purchased without microphones, which are necessary for audio recording. (Supertek, 2020).

Table 2.8 USB Camera

Author	Year	Title	Relevant Finding	Significance to our study
n.a.	2020	Difference between USB & IP	A camera that connects to a computer by typically being plugged into a USB port on the device is known as a USB webcam. The computer receives the video feed, and a software program allows you to view the images and upload them to the Internet.	The USB camera will be connected to the microcontroller Raspberry Pi 4 Model B for real time monitoring and for video capturing of leafy vegetables.

2.9 Raspberry Pi

Raspberry Pi is a small and powerful minicomputer, just like the size of an atm card. It is constantly upgraded for software and hardware, transforming it into a "Full-Fledged Computer" capable of performing intensive tasks within a specific timeframe. Python, C, C++, BASIC, Perl, and Ruby are among the programming languages supported by Raspberry Pi. (Ghael et al, 2008).

Table 2.9 Raspberry Pi

Author	Year	Title	Relevant Finding	Significance to our study
Ghael, H. et al.	2020	A Review Paper on Raspberry Pi and its Applications	It can support and connect multiple sensors at once, allowing embedded system enthusiasts to create/develop innovative projects and solve complex mathematical problems.	The Raspberry Pi will be our microcomputer in Pi-Droponics, and it is capable of solving complex mathematical problems that will be required in the project.

2.10 Wi-Fi Module

Data transmission and reception over Wi-Fi are handled by Wi-Fi modules or Wi-Fi microcontrollers. It can also acknowledge commands transmitted via Wi-Fi. Wi-Fi modules are utilized for inter-device communication. These are commonly applied to the Internet of things. (ERC, n.d.).

2.10.1 NodeMCU ESP8266

The NodeMCU, also referred to as the Node Microcontroller Unit, is a comprehensive software and hardware development environment that is openly accessible. It is built upon the ESP8266, a System-on-a-Chip (SoC) designed by Espressif Systems. This highly integrated chip encompasses essential components found in a computer, such as a CPU, RAM, Wi-Fi connectivity, a contemporary operating system, and an SDK. The NodeMCU offers an adaptable development platform that supports various programming languages and development tools, making it convenient for developers to create innovative Internet of Things (IoT) applications. Its compatibility with the ESP8266 empowers developers to harness its capabilities and efficiently develop connected products. By utilizing the NodeMCU, individuals from diverse backgrounds can seamlessly plan and implement IoT projects with ease. (*NodeMCU ESP8266 Specifications, Overview and Setting Up*, 2021).

2.10.2 Google Firebase

Google Firebase is a tool for creating, managing, and editing data produced by any Android or iOS application, online service, IoT sensor, or hardware. It is supported by Google. The official Google Firebase documentation is available from Google Firebase if you want to learn more about the Google Firebase Console. (*Send Sensor Data to Android Using Google Firebase & ESP8266, 2020*)

Table 2.10 Wi-Fi Module

Author	Year	Title	Relevant Finding	Significance to our study
Doug Stevenson	2021	ESP8266 NodeMCU: Getting Started with Firebase (Realtime Database)	The NodeMCU ESP8266 to connect to and communicate with the Firebase project, which has a real-time database for storing and reading data. As long as the ESP8266 is connected to the internet, it is possible for it to communicate with the database from any location.	The Pi-Droponics will use the NodeMCU ESP8266 as a Wi-Fi module to transmit data to Google Firebase for real-time monitoring of the condition of the leafy vegetables.

CHAPTER 3

METHODOLOGY

Chapter 3 introduces the methodology employed in this research, outlining the systematic approach used to collect and analyze data. This chapter explains the research design, data collection methods, and data analysis techniques utilized. It also highlights the rationale behind selecting these methods and justifies their suitability for addressing the research objectives. By describing the methodology, this chapter ensures the transparency and rigor of the research process, allowing readers to understand how the study was conducted and the reliability of the results.

3.1 Research Design

The research design that the proponents used in this study was Developmental Research since it focused on the development and innovation of the system and technology which was in line with the goal of this research study. The Automated Vertical Hydroponics System with Image Processing was deployed in Barangay 786 Zone 86, Sta. Ana Manila.

3.1.1 Input/Process/Output

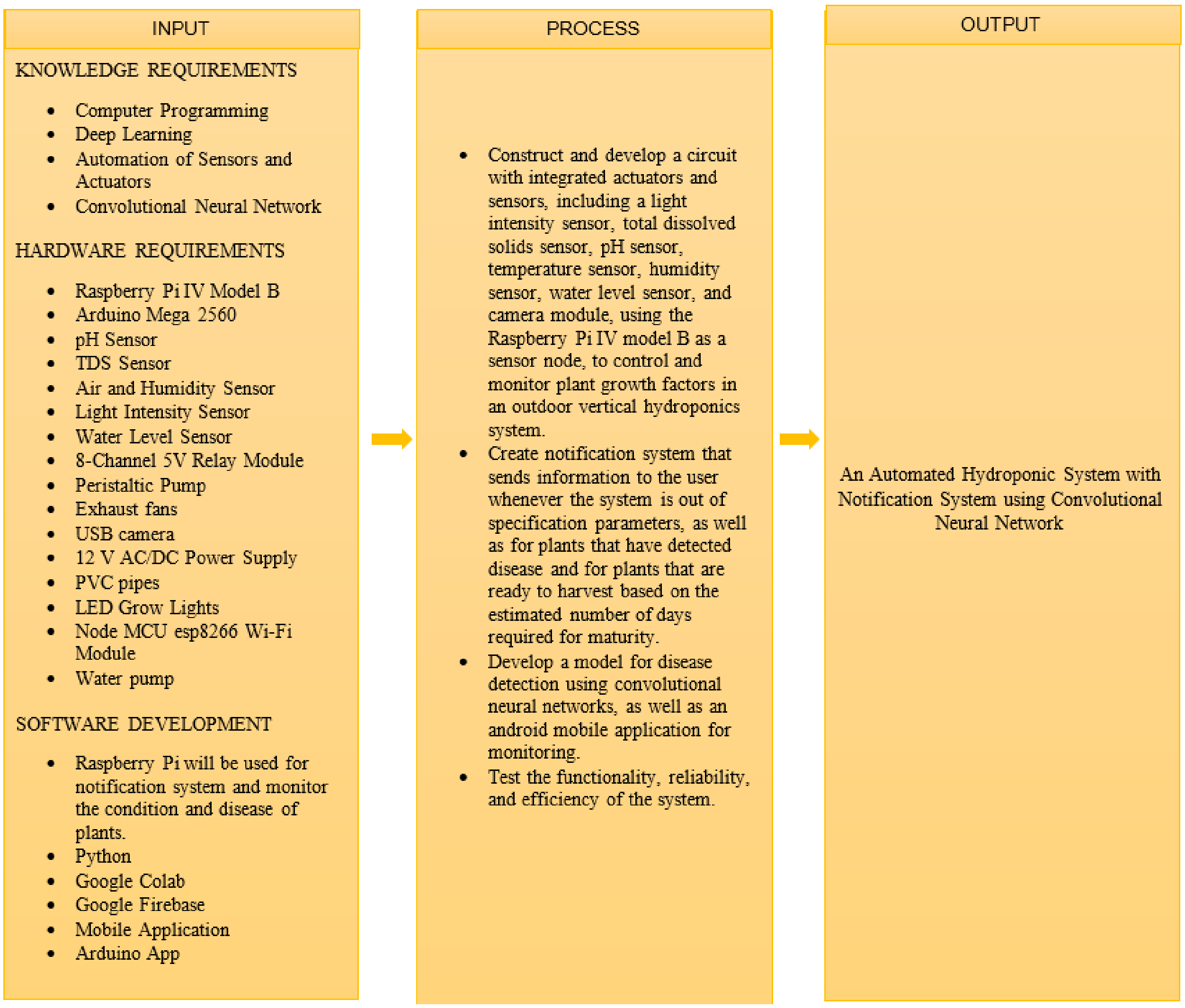


Figure 3.1.1 Input/Process/Output

Figure 3.1.1 illustrated the research design, which included the system's inputs, processes, and outputs. The input of the system included the different sensors and their actuators such as pH sensor, TDS sensor, air and humidity sensor, light intensity sensor, water level sensor, peristaltic pump, exhaust fans, LED grow lights—also the two (2) microcontrollers: the Arduino Mega 2560 and Raspberry Pi IV Model B. The data were sent to the Arduino microcontroller, which monitored and automatically controlled the desired optimum value whenever the parameters were imbalanced. On the other hand, the data gathered from the cameras were transmitted through the Raspberry Pi. In the system process, all the data gathered from two (2) microcontrollers were transmitted through the Firebase and then transferred to Mobile Application.

3.1.2 Block Diagram

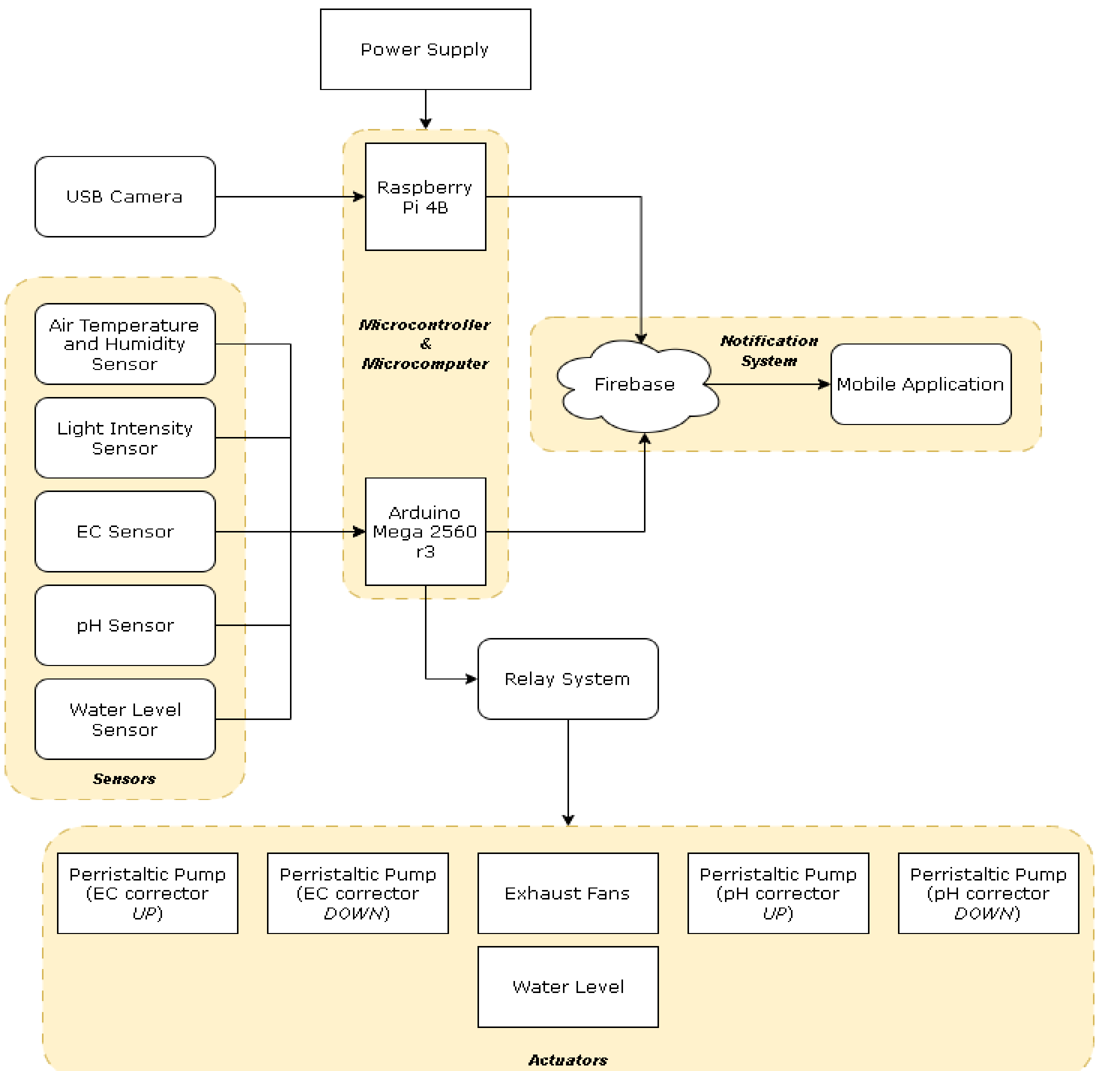


Figure 3.1.2 Block Diagram of the Automated Hydroponics System

Figure 3.1.2 shows the block diagram of a smart hydroponics system using six sensors to measure parameters. These sensors checked the pH levels, TDS (Total Dissolved Solids), air and humidity, light intensity, and water levels. The data from these sensors was sent to a mobile app using Arduino Mega 2560 R3. Additionally, if any diseases were detected by web cameras, that information was sent through Firebase and displayed on the mobile app. Users received notifications on their phones about these detected diseases.

3.2 Research Process Flow

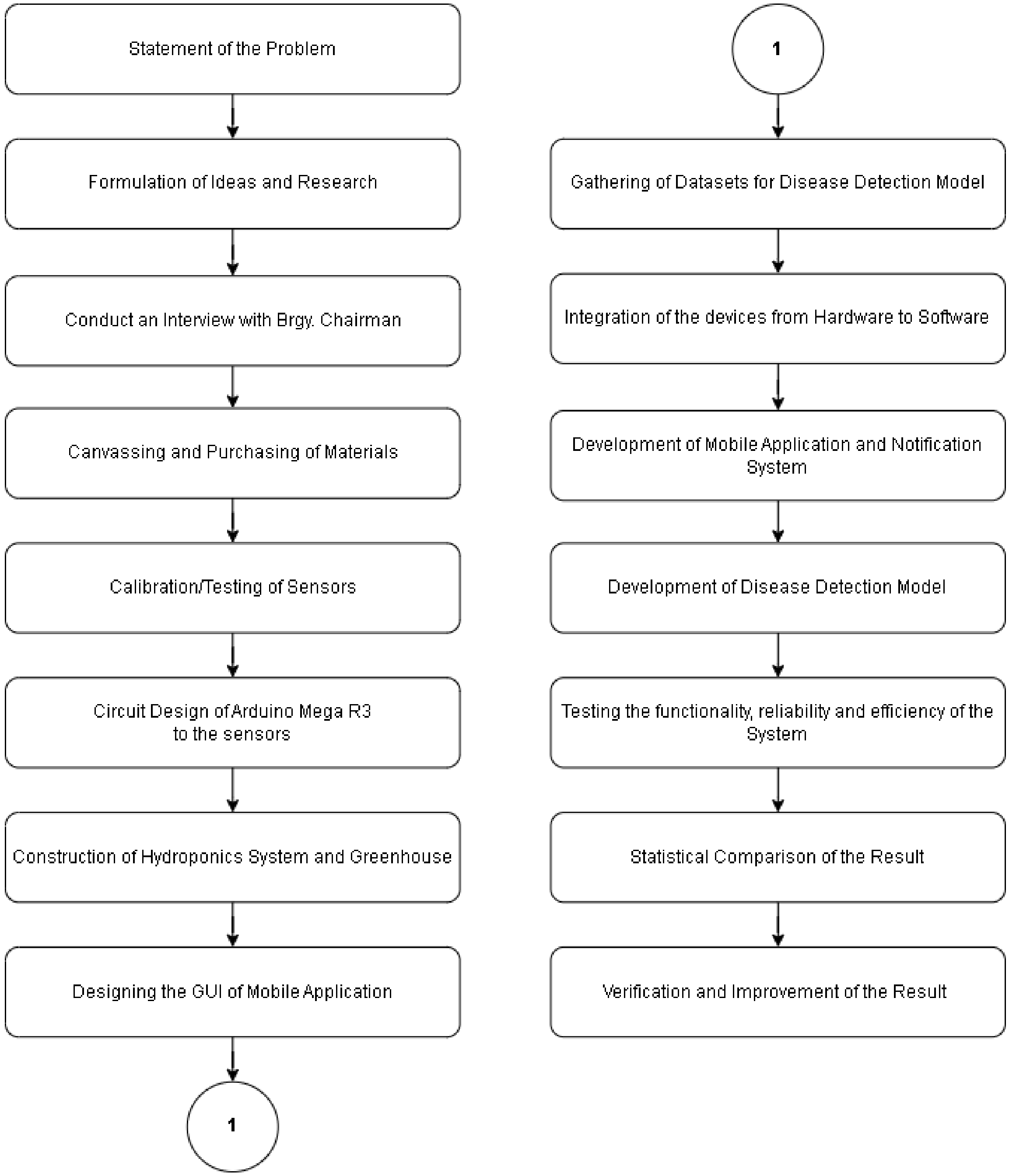


Figure 3.2 Research Process Flow

The research process flow of the proponents' study was illustrated in Figure 3.2. It commenced with the formulation of a "Statement of the Problem" to establish a suitable study title. The subsequent phase involved ideation and research, serving as the primary framework for the study. Following the formulation of ideas and comprehensive research, the proponents conducted an interview with the Barangay Chairman to obtain guidance and insights on suitable plant species for their area. Subsequently, the proponents engaged in material procurement, including sensors and microprocessors required for constructing the prototype. Calibration of the sensors and the design of a hardware circuit ensued after the acquisition of necessary materials. Simultaneously, the construction of the prototype and its accompanying greenhouse commenced. Alongside the physical construction, system monitoring codes were programmed, including parameter settings. In the software part, the proponents embarked on designing the graphical user interface (GUI) for their mobile application and gathering datasets for their disease detection model. The subsequent step involved the integration of software and hardware components. Further development encompassed the mobile application, notification system, and refinement of the disease detection model. Subsequent testing evaluated the functionality, reliability, and efficiency of the system, following which the proponents gathered data for result comparison and analysis. The final phase of the research process involved result verification and analysis. This critical step aided the proponents in achieving the stated objectives of their study. Upon confirming the fully operational status of the prototype, it was ready for deployment.

3.2.1 Hardware Flowchart of the Automated Hydroponics System

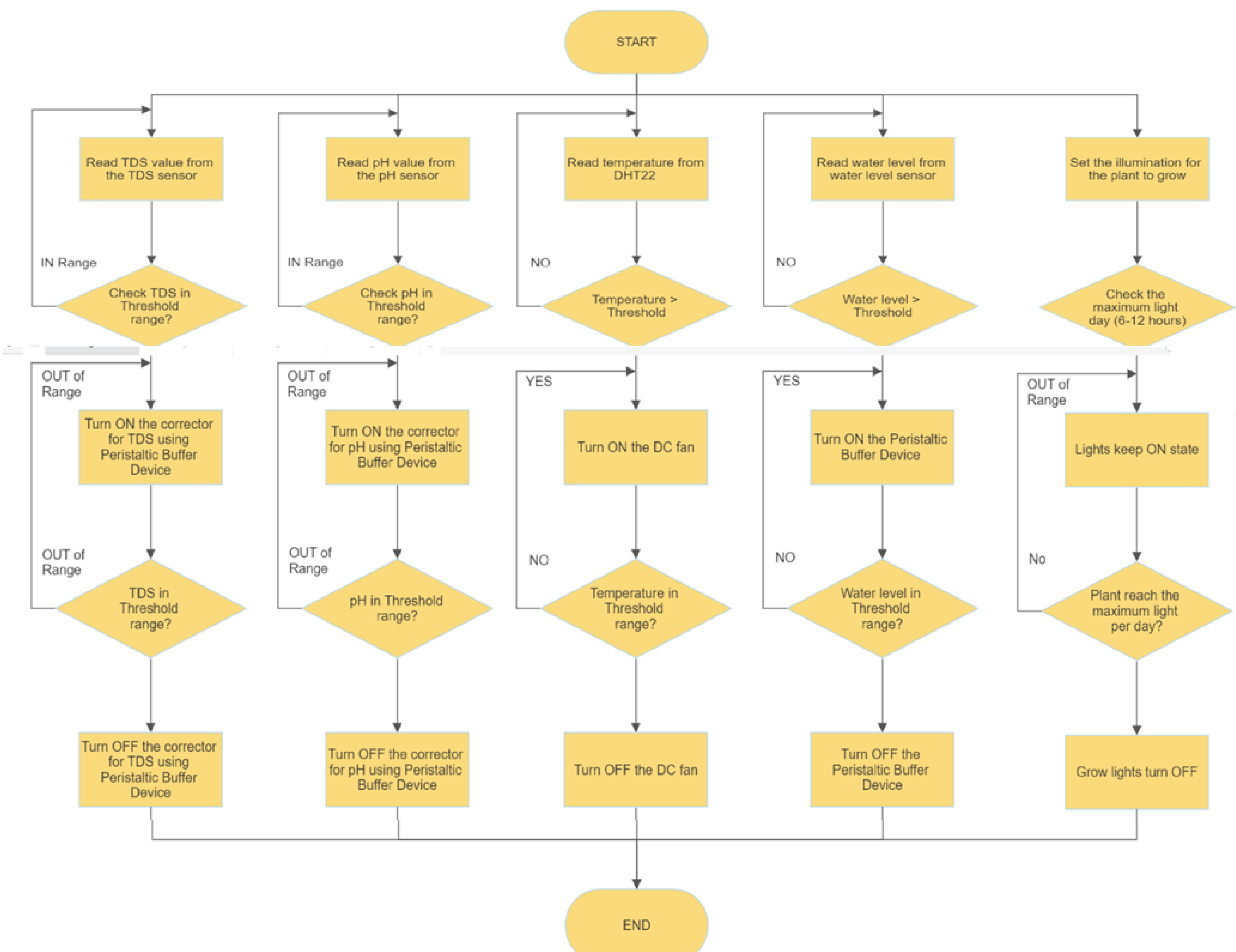


Figure 3.2.1 Hardware Flowchart of the Pi-Droponics System

Figure 3.2.1 shows the flowchart of the process flow for the hardware system of Hydroponics. The system has five (5) sensors that were all connected to the Arduino Mega 2560 R3; which read first all the data value from each sensor, then checked the set threshold for each sensor, and when the sensors were out of threshold's range it turned on the actuators such as the Peristaltic Buffer Device, DC fan, and grow lights. When thresholds were met, it automatically turned off the actuators of each sensor.

3.2.1.1 Temperature Parameter Process Flow Chart

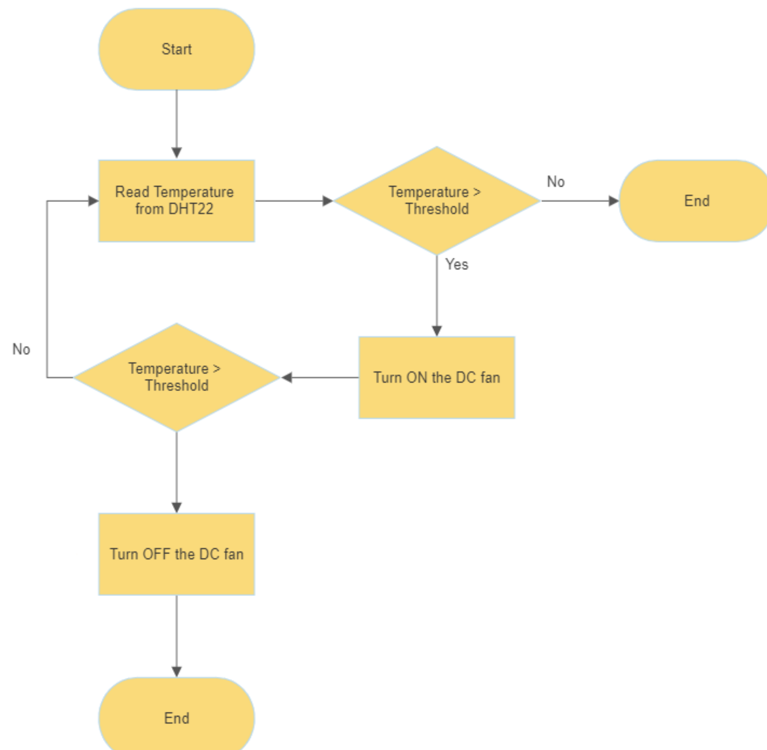


Figure 3.2.1.1 Flowchart of the System Temperature Correction

Figure 3.2.1.1 depicted the flowchart that outlines the process of regulating the air temperature and humidity within this system. This involved obtaining the temperature reading from the DHT22 sensor, which was crucial for the optimal growth of plants. Subsequently, the acquired values from the sensor were checked to determine if they fall within the predefined threshold range. If the values were either below or within the threshold, the system automatically shut off the fan. Moreover, the temperature threshold of the system was adjustable to align with the current season, enabling the system to adapt accordingly. The scale of temperature was fixed to Degrees Celsius (°C). The data acquired from the system were displayed in the HyPi IV Mobile Application.

3.2.1.2 pH Parameter Process Flow Chart

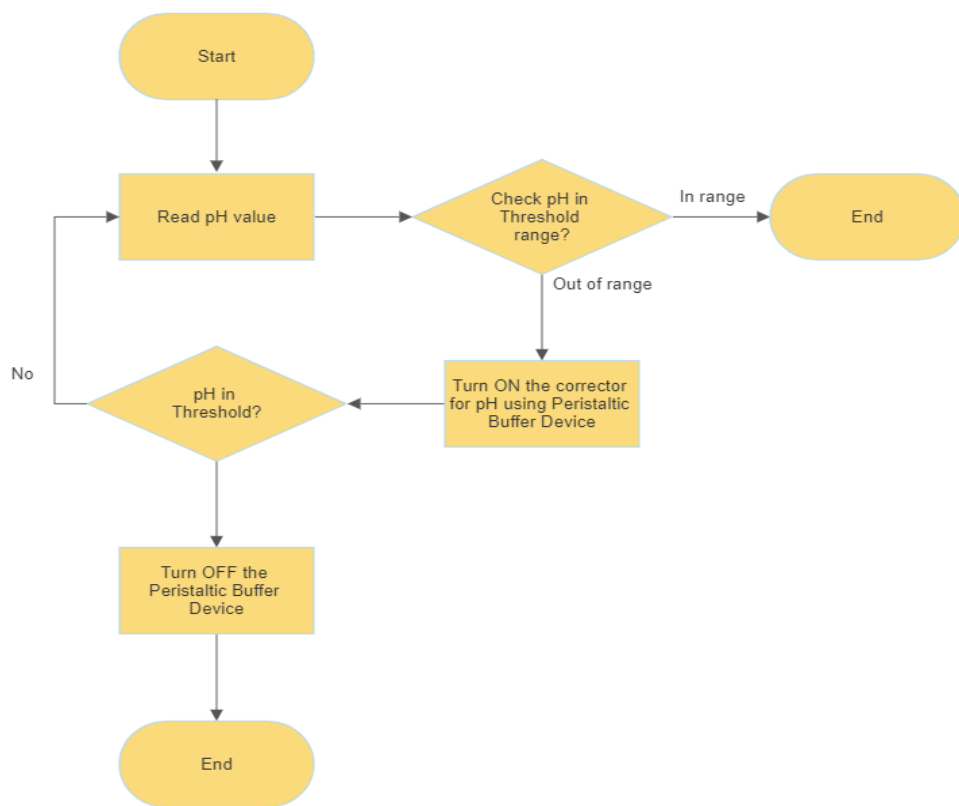


Figure 3.2.1.2 Flowchart of the System pH Correction

Figure 3.2.1.2 illustrated the correction of pH values, accomplished by examining the data obtained from the pH sensor. If the value falls outside the threshold range, the peristaltic buffer device will be activated to introduce a substance that adjusts the pH using pH up or pH down, depending on the specific situation, to decrease or increase the alkalinity of the nutrient solution accordingly. The buffer device remains inactive when the data falls within the predefined threshold range.

3.2.1.3 Total Dissolve Sensor (TDS) Parameter Process Flow Chart

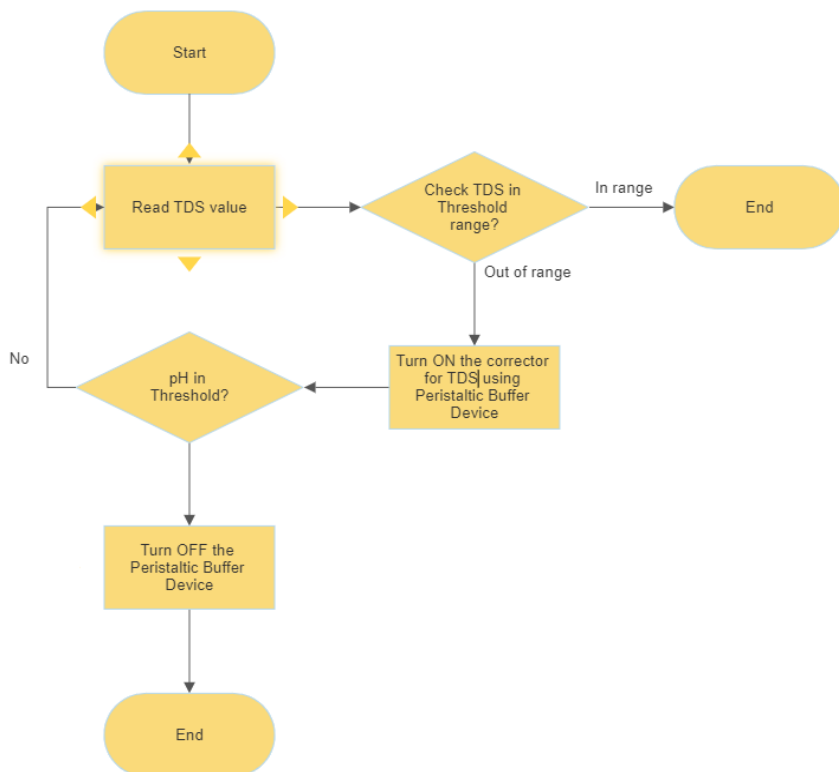


Figure 3.2.1.3 Flowchart of the System TDS Correction

Figure 3.2.1.3depicted the monitoring of TDS (Total Dissolved Solids) values. The process was similar to pH adjustment, as it aimed to regulate the basicity and alkalinity of the nutrient solution. If the TDS value exceeds the predetermined threshold range, the TDS mechanism will add water to dilute the solution. Conversely, if the TDS value falls within the threshold range, the mechanism remains inactive. The TDS parameter was measured in parts per million (ppm).

3.2.1.4 Light Intensity Parameter Process Flow Chart

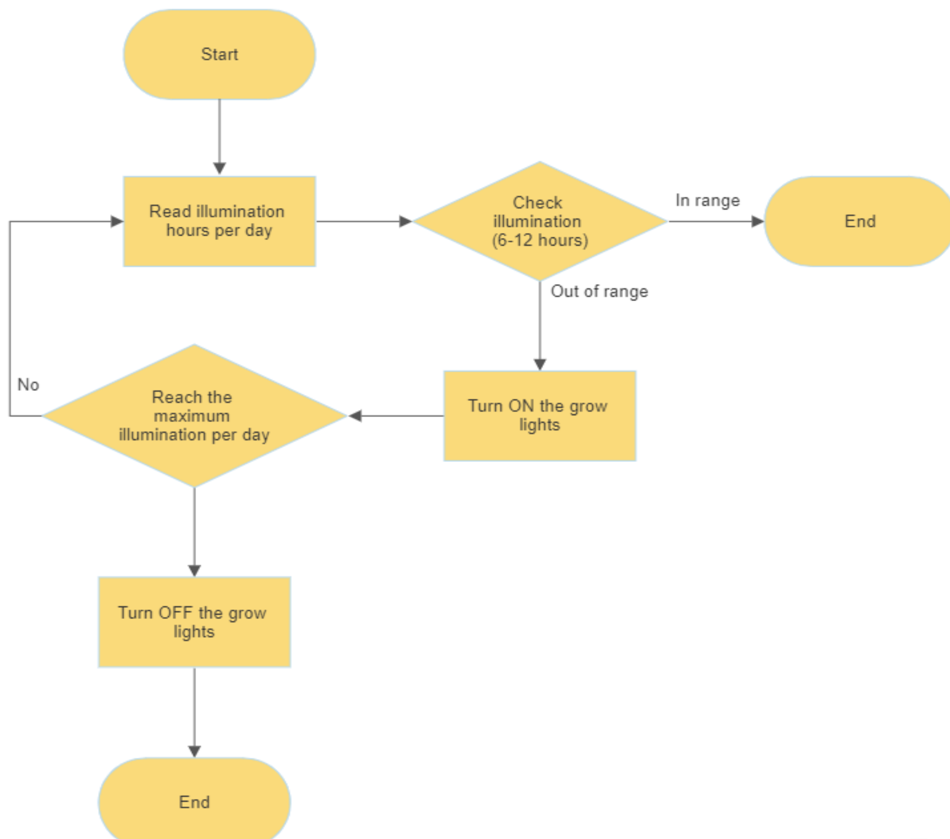


Figure 3.2.1.4 Flowchart of the System Light Intensity Correction

Figure 3.2.1.4 illustrates the system for monitoring and adjusting light intensity. The plants required 6 to 12 hours of illumination per day, with a target light intensity of at least 1000 lux. Once the desired number of hours is set, the system verifies if the maximum required light for the day has been achieved. If the set parameters are not within the specified range, the lights will remain on until the plants receive the necessary illumination. All collected data were transmitted and displayed on the user interface application.

3.2.2 Software Flowchart of the Automated Hydroponics System

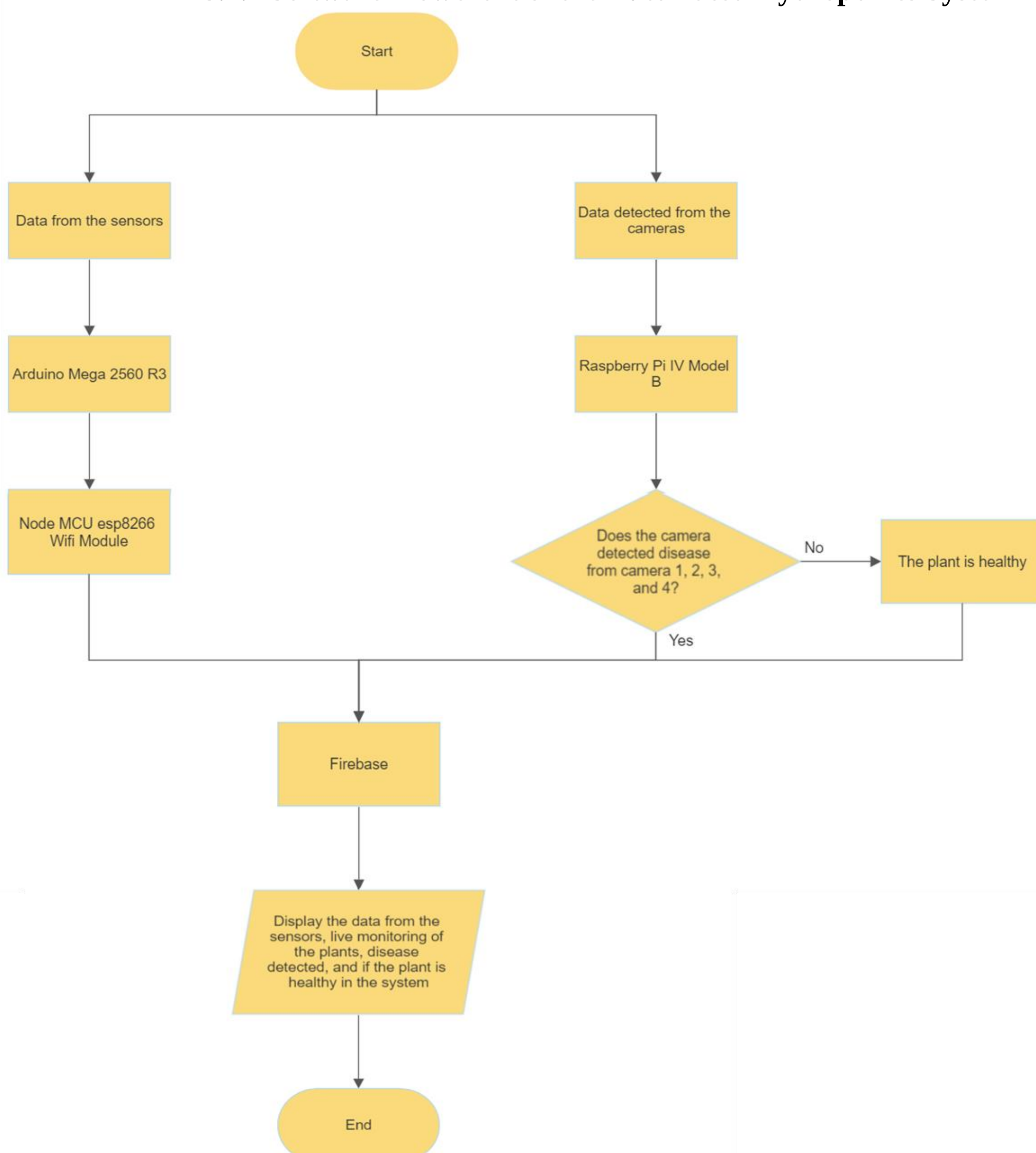


Figure 3.2.2 Software Flowchart of the Pi-Droponics System

Figure 3.2.2 shows the flowchart of the process flow for the software of Hydroponics. The software uses two (2) microcontrollers; the Arduino Mega 2560 R3 was where all the data collected from the sensors were stored which then sent to the Node MCU ESP8266, wirelessly to Firebase, then to the mobile application, and finally displayed the plant monitoring condition. Another microcontroller used was the Raspberry Pi IV Model B, where the data gathered from the cameras were stored. When the cameras detected disease, they sent it through a Firebase. It displayed to the mobile application if the plants were healthy; if not, it would indicate the diseases detected in the plants.

3.3 Hardware Development

The hardware construction and development involved the use of plywood, wood, and PVC pipes for the growing medium. The hardware components consisted of various parameter correctors, including the DC fan, and peristaltic buffer device. The peristaltic buffer devices were connected to the relay system to correct the pH and TDS parameters. Additionally, the hardware included different sensors such as light intensity, pH, TDS, water level, air, and humidity sensors. These sensors were connected to the Arduino Mega 2560 R3 microcontroller, which acted as the sensor node. Another microcontroller used in the system was the Raspberry Pi IV Model B, to which the cameras were connected. The Raspberry Pi was responsible for detecting diseases in the plants.

3.3.1 Materials and Equipment

The proponents acquired various materials and equipment to build the system. It included PVC pipes, pH sensor, water level sensor, TDS, air temperature and humidity sensor, light intensity sensor, Arduino MEGA 2560 R3, ESP8266 NodeMCU, Raspberry Pi IV B, USB camera, LED grow lights, and DC fan.

3.3.1.1 PVC Pipes

Figure 3.3.1.1 PVC Pipes

Figure 3.3.1.1shows the pH sensor version 1.1 that measures a solution's acidity or alkalinity on a scale of 0 to 14, with 7 signifying neutralities. As stated in Table 2.3.3, the pH level of the nutritional solution must be maintained between 6.0 and 7.0. The sensitive glass membrane used to create this industrial pH electrode has a low resistance. It has a quick response time and great temperature stability, making it suitable for many different PH measurements.

3.3.1.2 DFROBOT pH Sensor v1.1

Figure 3.3.1.2 pH Sensor

Figure 3.3.1.2 shows a water level sensor that is simple to use, economical, and capable of recognizing drops of water. It works by measuring the volume of water droplets on a sequence of parallel wires with exposed traces to estimate the water level. To produce the level warning effect, a straightforward conversion of water to an analog signal is required. The output analog values can also be received directly from an Arduino development board. The detecting area is 40mm x 16mm, the operational voltage is DC 3-5V, the operating current is less than 20mA, the operating temperature is 10°C to 30°C, and the operating humidity is 10% to 90% non-condensing.

3.3.1.3 Water Level Sensor

Figure 3.3.1.3 Water Level Sensor

The analog TDS sensor is shown in Figure 3.3.1.3 and is used to measure the total dissolved solids (TDS) in a solution to determine the water quality in hydroponics. The TDS probe is waterproof and may be submerged in water for extended periods of time. This device is compatible with a 5V or 3.3V control system or board since it offers a 3.3 to 5.5V broad voltage input and a 0 to 2.3V analog voltage output. TDS measurement accuracy is 10% F.S., with a range of 0 to 1000 ppm (25 °C).

3.3.1.4 DFRobot Analog Total Dissolved Solid Sensor

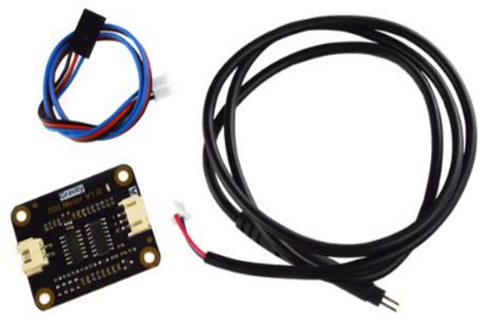


Figure 3.3.1.4 Total Dissolved Solid Sensor (TDS Sensor)

The analog TDS sensor is shown in Figure 3.3.1.4 and is used to measure the total dissolved solids (TDS) in a solution to determine the water quality in hydroponics. The TDS probe is waterproof and may be submerged in water for extended periods of time. This device is compatible with a 5V or 3.3V control system or board since it offers a 3.3 to 5.5V broad voltage input and a 0 to 2.3V analog voltage output. TDS measurement accuracy is 10% F.S., with a range of 0 to 1000 ppm (25 °C).

3.3.1.5 DHT22

Figure 3.3.1.5 Air Temperature and Humidity Sensor

Figure 3.3.1.5 shows the temperature and humidity sensors; these are fundamental, low-cost digital temperature and humidity sensors. It assesses the surrounding air using a capacitive moisture sensor and a thermistor and sends a digital signal to the information processor pin. It has 4 pins and bidirectional serial data output on a single bus for digital signals. The temperature measuring accuracy is +/- 0.5 °C, and the temperature range is -40 to 80 °C. The humidity measurement range is 0 to 100% RH, and the accuracy is 2% RH.

3.3.1.6 GY-30 BH1750FVI Module Digital Light Intensity Illumination Sensor

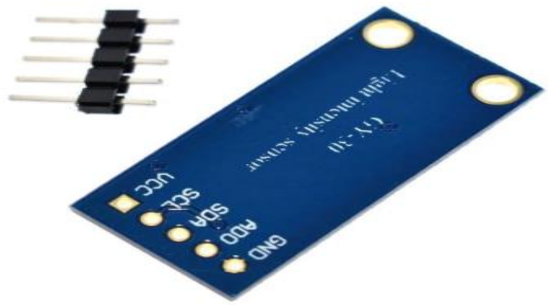


Figure 3.3.1.6 Light Intensity

Figure 3.3.1.6 shows the BH1750, a light intensity sensor with an integrated 16-bit AD converter and digital output that allows for the detection of light density and the reflection of an analog voltage signal back to the computer for Arduino. It measures ambient light intensity in the 0 to 65535 Lux (L) range with direct output in Lux. Basically, it measures intensity based on how much light it receives. The BH1750 uses a voltage range of 2.4 volts to 3.6 volts and only uses 0.12 mA of electricity. However, this module has a built-in voltage regulator that can accept input voltages between 3.3V and 5.0VDC. Since this measurement difference is so small (+/-20%), there are extremely minimal chances of any errors.

3.3.1.7 Arduino Mega 2560 R3

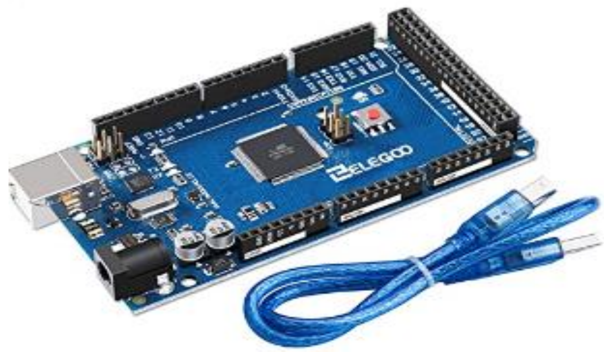


Figure 3.3.1.7 Arduino Mega 2560 R3

The ATmega2560-based Arduino Mega microcontroller board is shown in Figure 3.3.1.7. It contains 16 analog inputs, 4 UARTs (hardware serial ports), a 256k flash memory, a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. Of the 54-digital input/output pins, 14 of them can be utilized as PWM outputs. Along with the AREF, the Mega 2560 R3 also adds SDA and SCL pins. In addition, two new pins have been positioned close to the reset pin. However, one component is the IOREF, which permits the shields to react to the voltage supplied by the board. The other is unrelated and set aside for future use. The Mega 2560 R3 is compatible with all currently available shields and is adaptable to new shields that utilize these extra pins.

3.3.1.8 Actuators

An actuator is a motor that transforms energy into force and subsequently moves or operates a device or a system into which it has been integrated. Both causing and preventing motion are possible with it. Actuators are frequently used in conjunction with a power source and a coupling system. The power unit provides AC or DC power at the designated voltage and current rating. The researchers used peristaltic pumps, DC fans, LED grow lights, pH correctors, and peristaltic pumps as the actuators of the system for the hydroponics research project.

3.3.1.8.1 12V DC Fan

Figure 3.3.1.8.1 DC Fan

The 12V DC Fan is shown in Figure 3.3.1.8.1; it is a cooling fan that rotates by converting electrical energy into electromagnetic energy via DC voltage and electromagnetic induction, then mechanical energy from electromagnetic energy via electromagnetic induction, and finally kinetic energy from kinetic energy. The DC fan has the following specifications: an operating life of 80,000 hours, a current of 0.8 A, a noise level of 46.5 dBa, an air volume of 113CFM, and a speed of 3400 rpm.

3.3.1.8.2 Grow Light

This plant light emits all visible light wavelengths from 380 nanometers to 800 nanometers, just like natural sunlight, whereas LED grow lights serve as one of the actuators. It has been demonstrated to quicken photosynthesis and successfully encourage the growth of houseplants. Hydroponics, aquaponics, and indoor gardening are all possible uses for the LED grow light bar for indoor plants. The sunlight-led strip has a 10W/bar output, a 5V input voltage, 96 LED chips (48 LED chips per bar), and a 50000-hour lifespan.

Figure 3.3.1.8.2 Grow Light

3.3.1.8.3 Peristaltic Pump

Figure 3.3.1.8.3 Peristaltic Pump

This plant light emits all visible light wavelengths from 380 nanometers to 800 nanometers, just like natural sunlight, whereas LED grow lights serve as one of the actuators. It has been demonstrated to quicken photosynthesis and successfully encourage the growth of houseplants. Hydroponics, aquaponics, and indoor gardening are all possible uses for the LED grow light bar for indoor plants. The sunlight-led strip has a 10W/bar output, a 5V input voltage, 96 LED chips (48 LED chips per bar), and a 50000-hour lifespan.

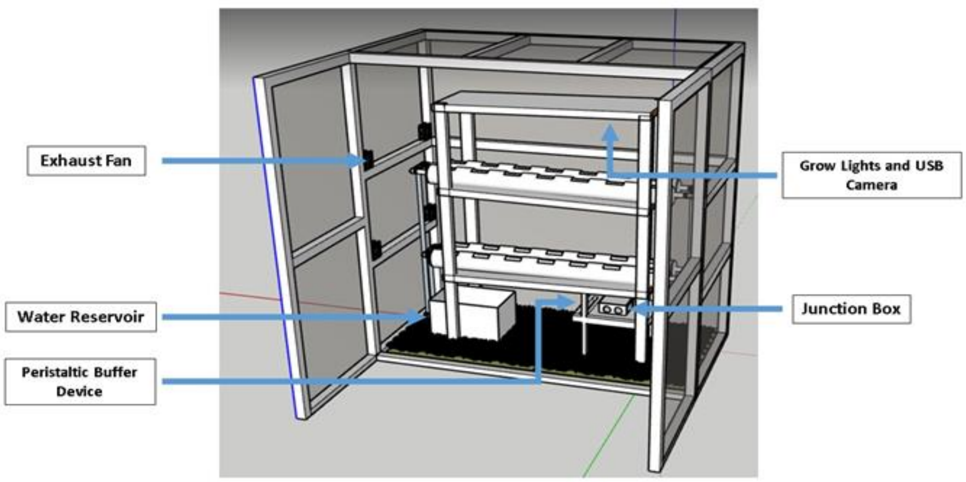
3.3.2 Hardware Construction

3.3.2.1 Greenhouse Design

The greenhouse was made up of low-cost materials, including 2x2 wood wrapped in a UV plastic sheet and filled with weatherproof artificial grass for the flooring. The greenhouse was equipped with a 12V DC fan that acted as the exhaust fan.

3.3.2.2 Vertical Hydroponics System Design

The proponents designed a vertical hydroponics system that used the Nutrient Film Technique (NFT) to reduce salt accumulation in the root region while also ensuring that nutrients flow regularly. The design of the system was made up of two layers, and each layer had two P



s with five holes.

Figure 3.3.2.2 Hydroponics Design

3.3.3 Calibration of Sensors

Before being installed in the system, the sensors and actuator components were calibrated to ensure that each component operates properly. The parameters of each sensor were set depending on the type of plant and its growth condition requirements.

3.3.3.1 DHT22 Calibration

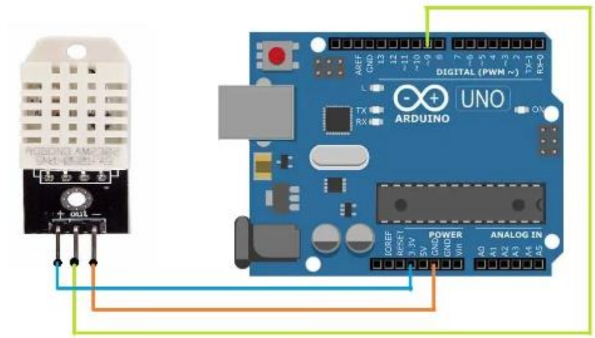


Figure 3.3.3.1 Calibration of DHT22 Sensor

DHT22, the temperature and humidity sensor are fundamental, low-cost digital temperature and humidity sensors. It utilizes a capacitive moisture sensor and a thermistor to assess the surrounding air and sends a digital signal to the information pin. Its use was relatively easy but needs cautious timing to collect information.

3.3.3.2 Light Intensity Sensor Calibration

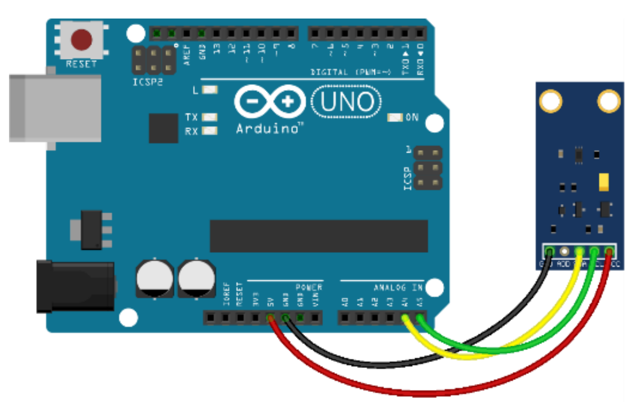


Figure 3.3.3.2 Calibration of Light Intensity Sensor

GY-30 BH1750FVI Light Intensity Sensor is an IC for 12 C bus interface; this light sensor is suited for Arduino or Raspberry Pi and enables you to detect ambient light and reflect the analog voltage signal to the computer for Arduino. High resolution allows for the recognition of a broad range. (1 - 65535 lx).

3.3.3.3 pH Sensor (DFRobot Analog pH Sensor) Calibration

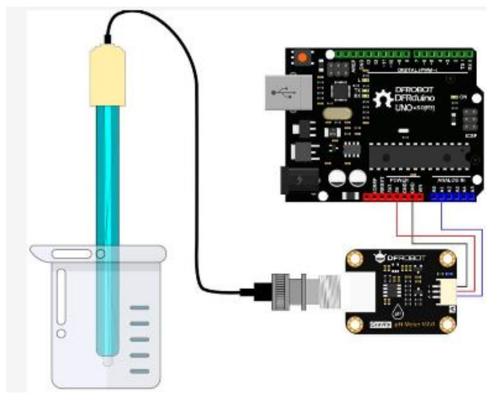


Figure 3.3.3.3 Calibration of pH Sensor

DFRobot Analog pH Sensor is an industrial-grade analog pH sensor that can withstand prolonged water submersion without recalibration. The integrated voltage regulator chip offers a wide voltage supply range of 3.3–5.5V, making it compatible with main control boards that operate at 5V and 3.3V. The sensor's electrode is a sensitive glass membrane with low resistance, making it highly precise.

3.3.3.4 TDS Sensor Calibration

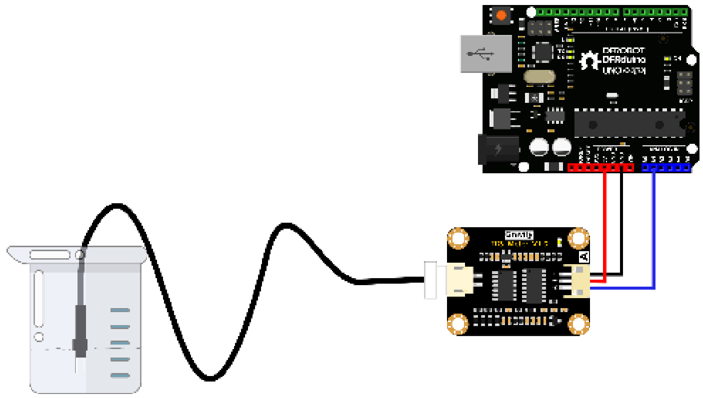


Figure 3.3.3.4 Calibration of TDS Sensor

DFRobot Analog TDS Sensor is a TDS meter kit compatible with Arduino that measures the TDS value of the water to reflect the water's cleanliness. The probe's excitation source was an AC signal, which helped boost the output signal's stability while efficiently preventing the probe from being polarized and extending its life.

3.3.3.5 Water Level Sensor Calibration

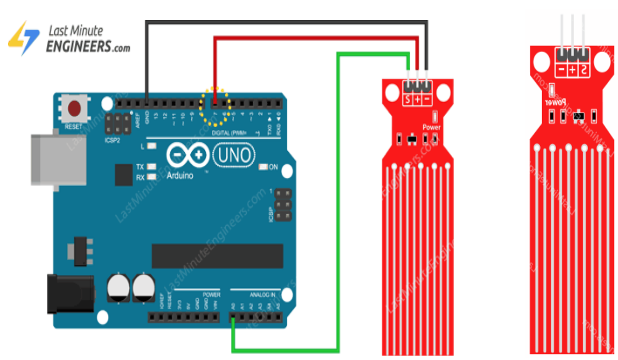


Figure 3.3.3.5 Calibration of Water Level Sensor

Water level Sensor is a device that measures the liquid level in a fixed container that is too high or too low. This sensor may be used as an indicator, low and high alarms, and automatically adjusts water levels.

3.3.4 Positioning of the Sensors and USB Camera

Figure 3.3.4 USB Camera

The proponents need to strategically place the sensors and USB camera in the hydroponics system for them to serve their functions properly. The water level sensor, pH sensor, and TDS will be placed directly in contact with the nutritional solution. The USB camera with 1080 pixel and a frame rate of 30 fps should be placed in the best position where it can monitor the plants in its maximized field of view.

3.3.5 Automation of the System

The proponents must devise a circuit with sensors and actuators to monitor and neutralize the air temperature, light intensity, TDS, and pH level for plants using LED grow lights, chamber air conditioning, and ventilation systems.

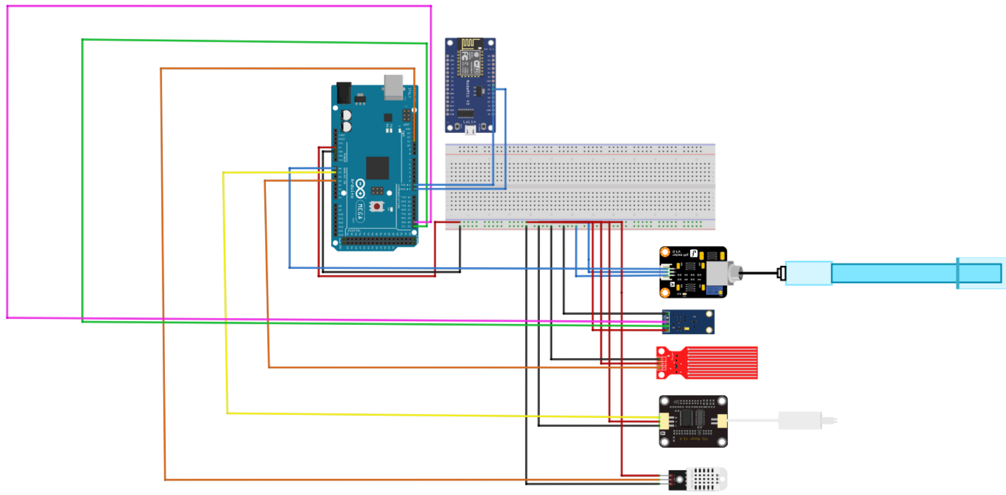


Figure 3.3.5 Circuit Diagram of Sensors of Arduino Mega 2560 R3 and NodeMCU ESP8266

Figure 3.3.5 shows the Circuit Diagram of the sensors where all the sensors were connected to the Arduino Mega 2560, reading the data simultaneously. Then it was connected to NodeMCU ESP8266 to send data from the sensors to the Firebase and transmit it to the Mobile Application wirelessly.

3.3.5.1 Actuators and Relay

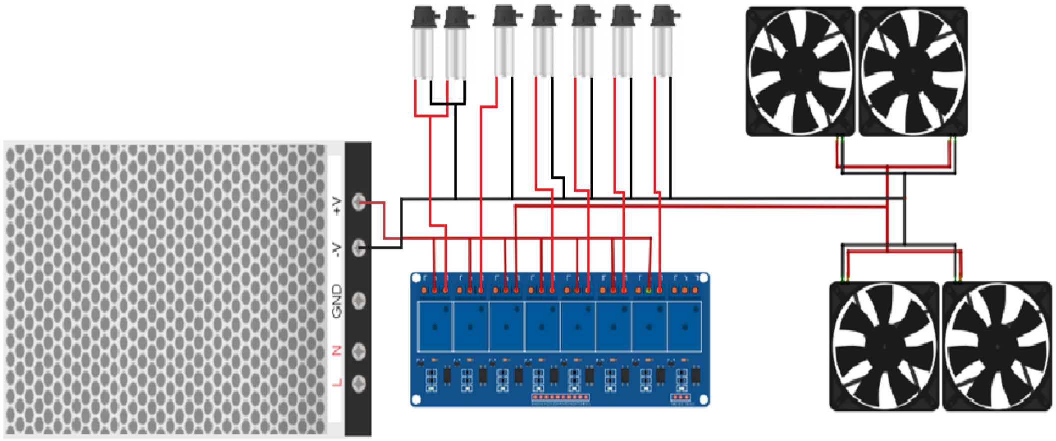


Figure 3.3.5.1 Connections of Actuators to Relay

Figure 3.3.5.1 shows the Circuit Diagram of Actuators to Relay that triggers to turn on depending if the readings from the sensors were out of the set threshold from the plant. A 12V, 10A Power Supply, powered the actuators.

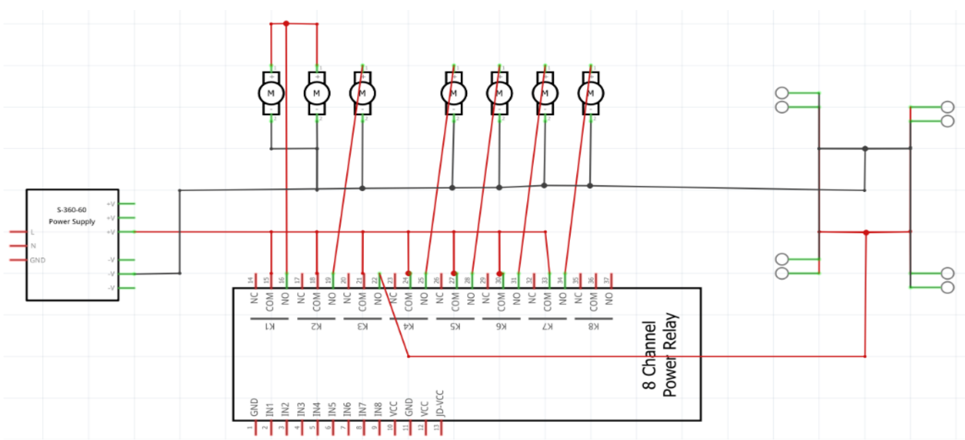


Figure 3.3.5.1.2 Schematic Diagram of Actuator and Relay

The 8-channel relay driver circuit, as shown in Figure 3.3.5.1.2, controlled the exhaust fan for the cooling system and the ventilation of the greenhouse. This also included the peristaltic buffer pump responsible for correcting pH, Total Dissolved Solids, and water level from the reservoir.

3.4 Software Development

In this section, the proponents will develop a model for disease detection of plants, as well as the mobile application needed for the monitoring and notification system.

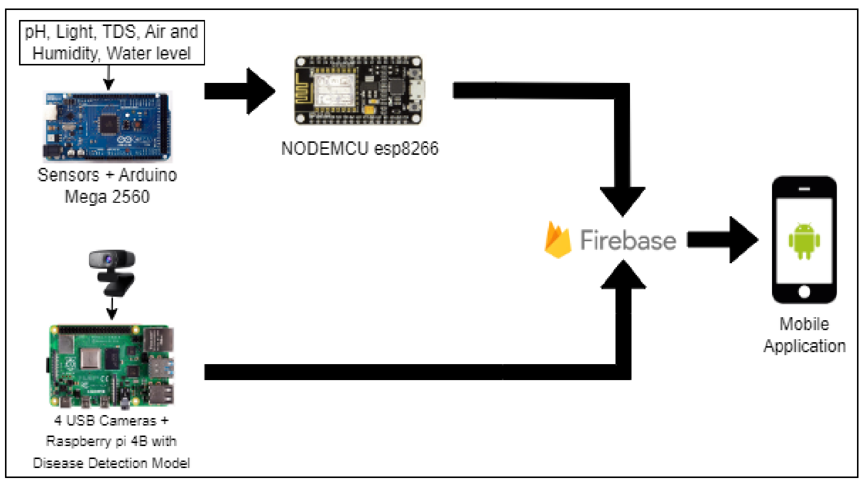


Figure 3.4 Network Architecture

The Firebase network architecture utilized two microcontrollers: the Arduino Mega 2560 R3 and the Raspberry Pi IV Model B. The Arduino Mega 2560 R3 established communication with the NodeMCU ESP8266 through a physical connection, collecting sensor data and transmitting it to the microcontroller. On the other hand, the Raspberry Pi 4 Model B was responsible for disease detection as it has an installed Disease Detection Model. This model incorporated an integrated Convolutional Neural Network (CNN) algorithm. Finally, the mobile application received data from the Google Firebase and displayed it on the user's phone screen.

3.4.1 Development of the Android Mobile Application

The proponents used the MIT App Inventor to create an Android mobile application. This application functioned as a monitoring and notification system, sending notifications anytime the sensor value fell outside the required range, a disease was found among the plants, or the plants were ready for harvest. To enable the notification system, the mobile application was directly linked to a database that stored the sensor and disease detection data.

Figure 3.4.1 MIT App Inverter

The MIT App Inventor is used to create the HyPi IV mobile application. Since this design is visual and relies on drag and drop, this tool is a programming learning tool for beginners, making it simple to understand the components required for an app. It developed due to collaboration between Google and MIT; proponents used coding to establish the project's Android mobile app.

3.4.1.1 Notification System Development

This section explains the steps required to create a notification system that sends information from the user whenever the system's parameters are outside of specification, as well as for plants that have detected disease; they will be notified as unhealthy plants. The user will also be notified when a plant is ready for harvest based on the estimated number of days that is shown on the mobile application's calendar.

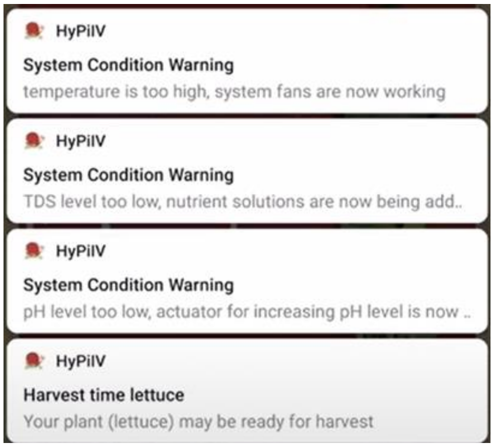


Figure 3.4.1.1 Notification System

Figure 3.4.1.1 Another feature of the mobile application. The mobile application will notify the user if the plants are healthy and ready to harvest; if not, the plants have a disease. It will also show what kind of disease is detected by the camera.

3.4.2 Collection of the Data Sets

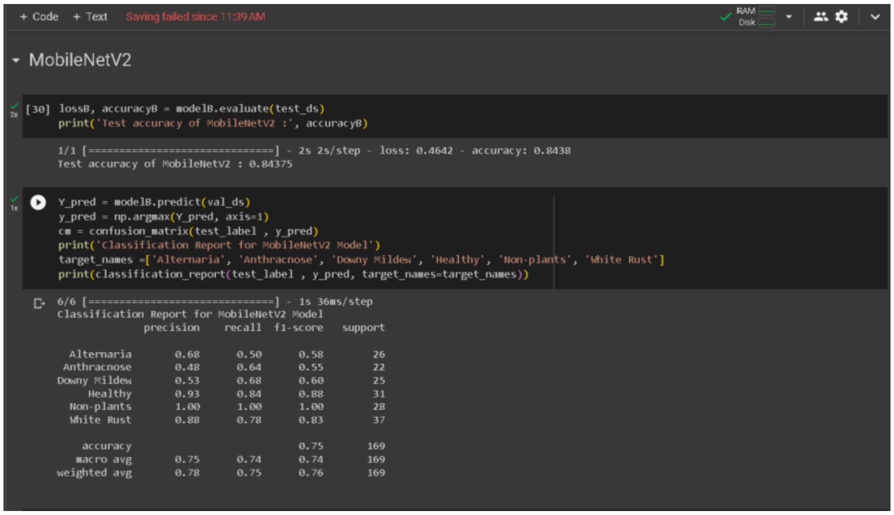
The datasets for various leafy vegetable diseases have been gathered by searching for images on the websites of Git Hub, Kaggle, and Reddit. These datasets were then used to train the disease detection model.

3.4.3 Development of the Model for Plant Disease Detection

By training the gathered data sets, the proponents created a model for detecting plant diseases. The disease detection model can determine whether the plant is healthy, has downy mildew disease, anthracnose disease, white rust disease, alternaria leaf spot disease, or is not yet fully grown.

Figure 3.4.3.1 Google Colab

Google Colaboratory, or Google Colab, is a cloud-based Jupyter notebook environment that is a product of Google Research. This runs in your web browser, which authorizes everybody to implement arbitrary Python code through the browser. Using Google Colab lets everyone experiment with machine learning and coding for artificial intelligence by having internet access. The Pi-Droponics Mobile Software Application used Google Colab as its programming tool to execute the model for plant disease detection.



(a)

(b)

Figure 3.4.3.2 Testing of Pre-trained Model for Disease Detection

Figure 3.4.3.2 presents a visual representation of employing a dual ensemble of convolutional neural network (CNN) pre-trained models to facilitate a comprehensive comparative analysis. The objective of this analysis was to identify the model that exhibits the highest level of accuracy. This evaluation aimed to determine the most appropriate candidate for incorporation into the disease detection framework. The dataset utilized in this study comprised 170 images per disease, which had been employed for training, testing, and validating the model.

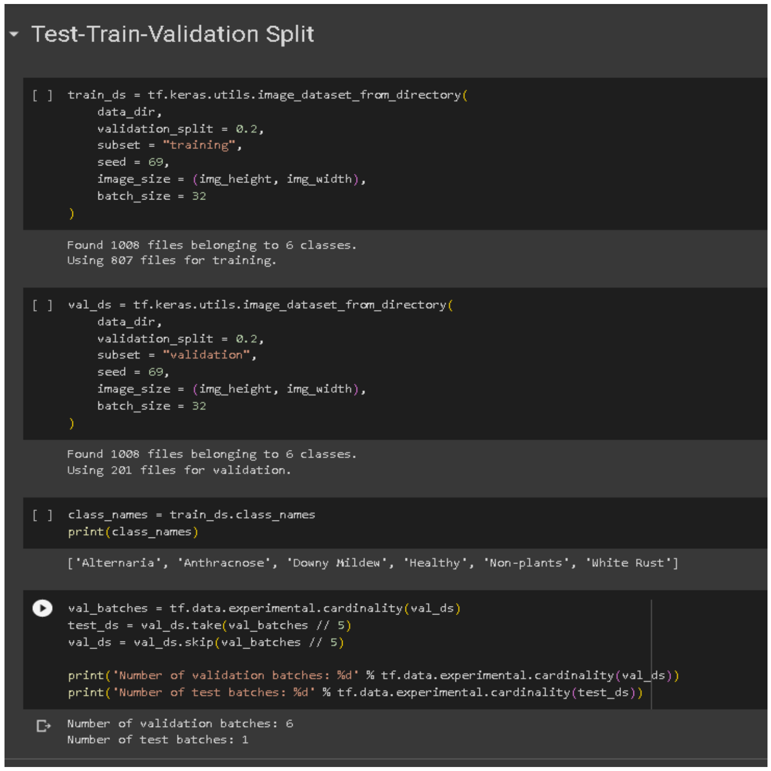


Figure 3.4.3.3 Test, Train, and Validation of Datasets for Disease Detection

The testing, training, and validation of a CNN pre-trained model with 170 images per disease involves preparing a representative dataset; partitioning it into training, validation, and test sets, training the model on the training set to learn disease-specific features, evaluating, and fine-tuning the model using the validation set, and finally, assessing the model's generalization ability on the test set. This iterative process ensures the development of an accurate and reliable CNN model for disease detection.

```
[ ] class_names = train_ds.class_names
    print(class_names)

    ['Alternaria', 'Anthracnose', 'Downy Mildew', 'Healthy', 'Non-plants', 'White Rust']
```

```
val_batches = tf.data.experimental.cardinality(val_ds)
test_ds = val_ds.take(val_batches // 5)
val_ds = val_ds.skip(val_batches // 5)

print('Number of validation batches: %d' % tf.data.experimental.cardinality(val_ds))
print('Number of test batches: %d' % tf.data.experimental.cardinality(test_ds))

Number of validation batches: 6
Number of test batches: 1
```

Figure 3.4.3.4 Naming 6 Classes for Disease Detection

A total of 1008 files were considered for the training and validation of the CNN model. Among these files, 807 files were allocated for the purpose of training the model, while the remaining 201 files were reserved for validation purposes. The dataset consisted of six distinct classes, namely 'Alternaria', 'Anthracnose', 'Downy Mildew', 'Healthy', 'Non-plants', and 'White Rust'. The primary objective of the model was to accurately identify and classify samples belonging to these specific classes.

3.6 System Assessment

3.6.1 Comparison between Conventional Method and Hydroponics System

The proponents compared the growth of the leafy vegetables based on the parameters of length, height, number of leaves, and the color of the plants. These parameters assessed which farming method was best for producing a higher yield of good-quality of plants.

3.6.2 Testing of the Functionality

The proponents tested if the system can distinguish healthy plants and unhealthy plants, as well as if the application's calendar can determine whether the plant was ready to harvest or not.

3.6.3 Testing of Reliability and Efficiency

The proponents tested the reliability and the efficiency of the system by comparing the response time of the notification system through the mobile application to the actual time of detection of disease.

3.7 Statistical Analysis

The gathered data were evaluated using a two-tailed t-test statistical analysis. This analysis aimed to determine and identify whether there is a significant difference between the conditions of plants in a vertical hydroponics system and conventional farming.

Table 3.7 Two-tailed t-test Formula

Mean	$\text{Mean } (\mu) = \frac{\sum x}{N}$	**Σx** = sum of all values **N** = total number of values
Standard deviation	$\text{Standard deviation } (\sigma) = \sqrt{\frac{\sum(x_i - \mu)^2}{N}}$	$\mathbf{x_i}$ = each value from the population **μ** = population mean **N** = total number of values
Variance	$\text{Variance } (\sigma^2) = \frac{\sum(x_i - \mu)^2}{N}$	$\mathbf{x_i}$ = each value from the population **μ** = population mean **N** = total number of values
Degrees of freedom	$df = n_1 + n_2 - 2$	$\mathbf{n_1}$ = the total number of values for population 1 $\mathbf{n_2}$ = the total number of values for population 2
T-value	$t - \text{value} = \frac{\lvert\mu_1 - \mu_2\rvert}{\sqrt{\frac{{\sigma_1}^2}{n_1} + \frac{{\sigma_2}^2}{n_2}}}$	μ_1 = mean of population 1 μ_2 = mean of population 2 $\boldsymbol{\sigma_1}^2$ = variance of population 1 $\boldsymbol{\sigma_2}^2$= variance of population 2 $\mathbf{n_1}$ = the total number of values for population 1 $\mathbf{n_2}$ = the total number of values for population 2
Critical Value	To obtain the critical value from the t-table, you can refer to the corresponding value based on the degrees of freedom and a significance level of p=0.05.	

Table 3.7 shows the formulas of the two-tailed t-test to get the statistical computation for the comparisons of the two (2) farming methods for the equipment and sensors that will be used.

3.8 Research Instrument

In the Pi-Droponics: An Automated Hydroponics System with Notification System using Convolutional Neural Network, the proponents gathered user feedback on the system's performance to assess the functionality and sustainability of the prototype and mobile application—residents of Barangay 786 in Zone 86 of Sta. Ana served as the respondent where the prototype was deployed. The responses to the questions provided directed evaluation and future prototype improvement. The evaluation used a 4-point Likert scale based on different standards according to ISO 25010 to evaluate the software's quality.

The responses from the ISO 25010 survey were evaluated through statistical analysis by getting the mean for each given standard. The proponents used the Range formula to know the ranges of the interpretation where it fell on. Four (4) interpretations for the 4-Likert Scale were Excellent Performance, Good Performance, Average Performance, and Poor Performance.

Table 3.8 Statistical Analysis for User's Evaluation

Range	Interpretation
3.26 – 4	Excellent Performance
2.51 – 3.25	Good Performance
1.76 – 2.50	Average Performance
1 – 1.75	Poor Performance

$$\text{Mean} = \frac{Sum\ of\ All\ Data\ Points}{Number\ of\ Data\ Points}$$

$$\text{Range} = \frac{\text{Maximum Value} - \text{Minimum Value}}{\text{Maximum Value}} + \text{Minimum Value}$$

INTRODUCTION: The students involved in the completion of this study needs to conduct an evaluation questionnaire regarding to the different factors and parameters, entitled "Pi-Droponics: An Automated Hydroponics System with Notification System using Convolutional Neural Network." The questionnaires are based on different standard based on ISO 25010 to determine the quality of the software and technical using a 4-point Likert scale.
INSTRUCTION: Please rate each of the parameters and check (✓) one response if your review suffice to this rating. Check (✓) 4 – **Very Functional**, 3 – **Functional**, 2 – **Slightly Functional** and 1 – **Not Functional**

FUNCTIONAL SUITABILITY	**RANGE**	**INTERPRETATION**	**CHECK**
Efficiently meets all the stated and implied need; and no weakness are found.	4	Very Functional	
Satisfactorily meets all the stated and implied needs but acceptable /tolerable weakness are found which will however not affect its function.	3	Functional	
Meet only some of the stated and implied needs and minor weaknesses are found that will slightly affect its function.	2	Slightly Functional	
Meets very few stated and implied needs; and major weaknesses are found that will greatly affect its function.	1	Not Functional	

PERFORMANCE EFFICIENCY	**RANGE**	**INTERPRETATION**	**CHECK**
Efficiently performs all the stated and implied needs relative to the amount of resources without any flaws.	4	Very Functional	
Satisfactorily performs all the stated and implied needs relative to the amount resources used but tolerable flaws are found which will however not affect the MTA's over-all performance.	3	Functional	
Poorly performs the stated and implied needs relative to the amount of resources used and major flaws are found that will greatly affect performance.	2	Slightly Functional	
Does not perform the stated and implied needs relative to the amount of resources used. Thus, restricting is needed.	1	Not Functional	

COMPATIBILITY	RANGE	INTERPRETATION	CHECK
Efficiently meets the required function while sharing the same hardware and software environment. It can exchange information with other systems with no inadequacies at all.	4	Very Functional	
Satisfactorily meet the required function while sharing the same hardware and software environment. It can exchange information with other systems but tolerable inadequacies are found which will not in any way affect the MTA's over-all compatibility.	3	Functional	
Poorly performs the required function while sharing the same hardware and software environments. It can exchange information with other systems but major inadequacies are found that will greatly affect its compatibility.	2	Slightly Functional	
Does not meet the required function while sharing the same hardware and software environment. It cannot exchange information with other systems thus restructuring is needed.	1	Not Functional	

USABILITY	RANGE	INTERPRETATION	CHECK
Provides a user interface that is easy to operate and control such that the users can effortlessly perform their appropriate needs even without guide or supervision.	4	Very Functional	
Provides a user interface that is easy to operate and control such that the users can easily perform their appropriate needs with minimal guide or supervision.	3	Functional	
Provides a user interface that is slightly difficult to operate and control such that the users can easily perform their appropriate needs.	2	Slightly Functional	
Provides a user interface that is difficult to operate and control such that it is difficult for the users to perform their appropriate needs at all. Thus, restructuring is required for the MTA.	1	Not Functional	

RELIABILITY	RANGE	INTERPRETATION	CHECK
Efficiently performs its functions without failure under specified conditions and period of time with no flaws at all.	4	Very Functional	
Satisfactory performs its functions without failure under specified conditions and period of time with minimal flaws which will not in any way affect the MTA's overall reliability.	3	Functional	
Performs only some of its functions without failure under specified conditions and period of time with major flaws which will affect the MTA's overall reliability.	2	Slightly Functional	
Cannot perform any of its functions without failure under specified conditions and period of time, thus restructuring is required.	1	Not Functional	

SECURITY	RANGE	INTERPRETATION	CHECK
Efficiently protects information and data with no weaknesses at all.	4	Very Functional	
Satisfactory protects information and data with minimal weaknesses which will not in any way affect the MTA's overall security.	3	Functional	
Protects only some of its information and data with major weaknesses which will affect the MTA's overall security.	2	Slightly Functional	
Cannot protect any information and data, thus restructuring is required.	1	Not Functional	

MAINTAINABILITY	RANGE	INTERPRETATION	CHECK
Efficiently retains its original form and can be restored to that form in case of failure, with no weaknesses at all.	4	Very Functional	
Satisfactory retains its original form and be restored to that form in case of failure; with minimal weaknesses, which however, will not affect its overall maintainability.	3	Functional	
Retains and restores only some of its original features in case of failure; with moderate weaknesses which may affect its overall maintainability.	2	Slightly Functional	
Cannot retain and restore its original form in case of failure, thus restructuring is required.	1	Not Functional	

PORTABILITY	RANGE	INTERPRETATION	CHECK
Efficiently adapts to changes in environment and can be installed / uninstalled in specified environment, with no weaknesses at all.	4	Very Functional	
Satisfactory adapts to changes in environment and can be installed / uninstalled in specified environment with minimal weaknesses, which however, will not affect its overall portability.	3	Functional	
Barely adapts to changes in environment and can be installed / uninstalled in specified environment with moderate weaknesses which may affect its overall portability.	2	Slightly Functional	
Cannot adapt to changes in environment and cannot be installed / uninstalled in specified environment.	1	Not Functional	

3.9 Project Work Plan (Gantt Chart)

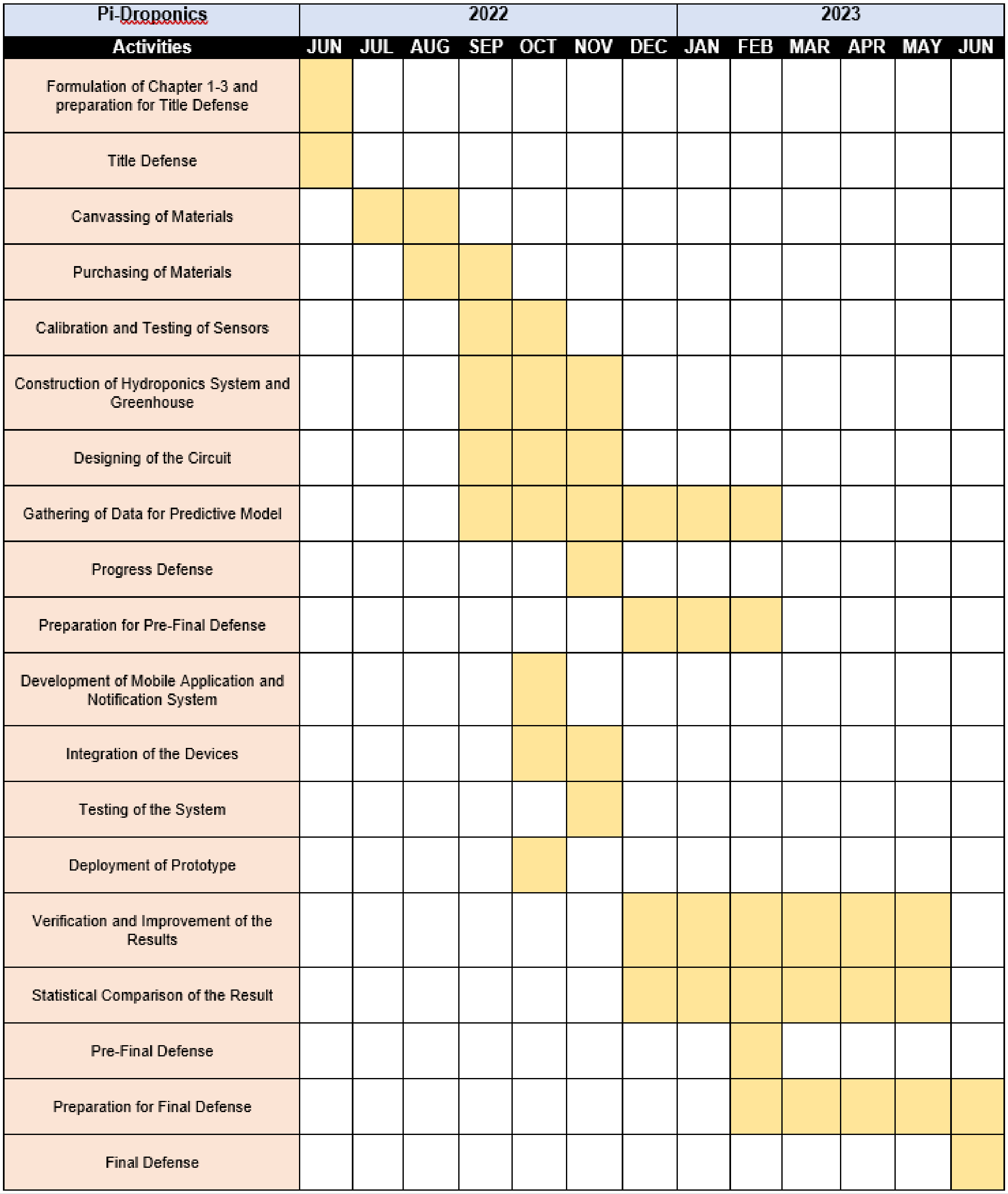


Figure 3.9 Project Gantt Chart

Chapter 4

RESULTS AND DISCUSSION

Chapter 4 provides an overview of the results obtained from the research and initiates a comprehensive discussion around those findings. This chapter presents the data analysis outcomes, highlighting key trends, patterns, and relationships identified. The discussion section critically examines the implications of the findings, explores their significance, and addresses any limitations or inconsistencies observed. This chapter contributes to the overall understanding of the research topic and helps to draw meaningful conclusions.

4.1 Project Technical Description

Since there had been numerous studies on hydroponics—an agricultural method that involved growing plants without soil and instead in water systems that contain nutrients and other growth-supporting media—researchers added innovation to these studies. The research study, entitled "An Automated Hydroponics System with Notification System Using Convolutional Neural Network," utilized machine learning to identify plant diseases and included a mobile application-based notification system, making the project technologically advanced. Due to the automated hydroponics technology, this project required minimal human interaction.

The sensors detected various system parameters such as pH, TDS (total dissolved solids), water level, light intensity, air temperature, and humidity. A light intensity sensor, DHT22, pH sensor, TDS sensor, and water level sensor were all programmed and controlled by an Arduino Mega

2560 R3 used in the study. The serial communication data from the Arduino were then transmitted to the ESP8266 NodeMCU, to Google Firebase, and eventually to the mobile application.

The system can detect diseases using a CNN algorithm and also has a notification system through a mobile application. The proponents used four cameras connected to a Raspberry Pi 4 Model B that served as a microprocessor that collected data for plant monitoring to determine the plant's condition. The collected datasets (170 images) were used to classify the different types of plants for disease detection; each plant has four different types of disease, and the plants used are lettuce, spinach, and Chinese cabbage. The readings of the sensors were transmitted to the Arduino Mega 2560 R3 for the notification system, and the serial communication data from the Arduino were transmitted to the ESP8266 NodeMCU, Google Firebase, and finally to the mobile application that displayed the status of the plants. The proponents used ensemble learning and image classification for the machine learning portion and MIT App Inventor for the mobile application.

4.2 Project Structural Organization

4.2.1 The Greenhouse and Hydroponics Set-up

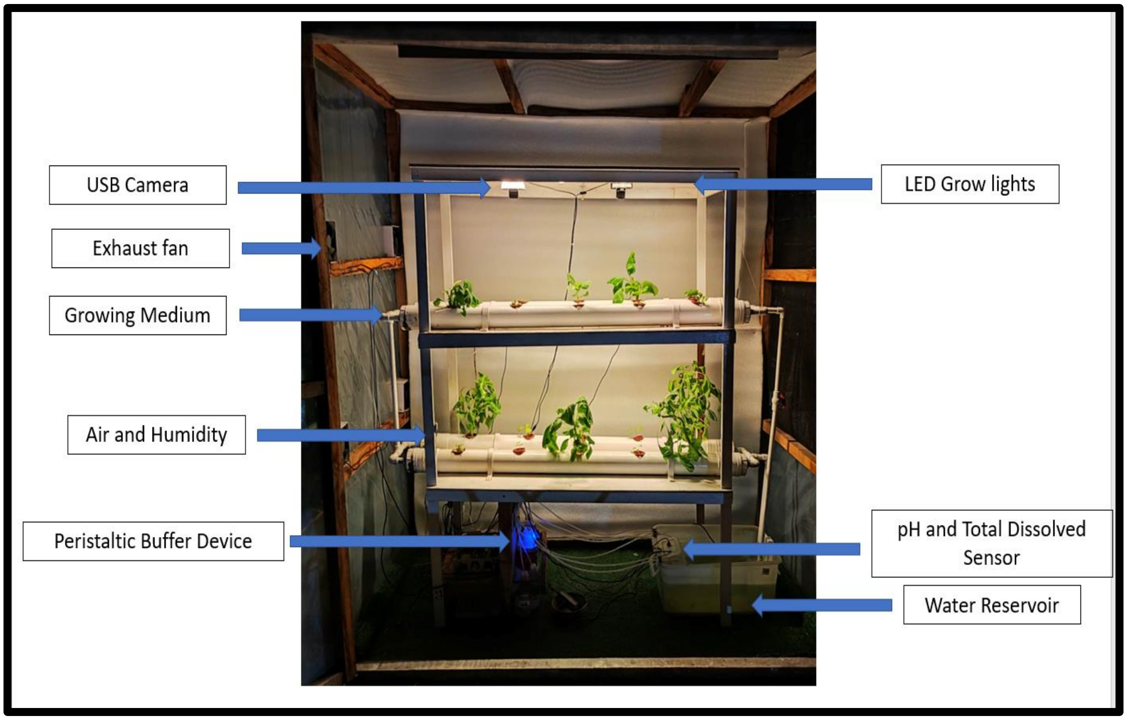


Figure 4.2.1 The Greenhouse and Hydroponics System

The greenhouse measured 6.3 feet by 3.4 feet by 6 feet and was made of low-cost wood covered with UV plastic. It has four 12V DC fans mounted on UV plastic, shown in Figure 4.1, to remove humid air and supply fresh air, allowing the plants to grow quickly and abundantly. A Nutrient Film Technique (NFT) was used in the automated vertical hydroponics system to ensure a constant flow of nutrient-rich water, which aided in maintaining health and stress-free conditions. PVC pipes and wood were used to construct the hydroponics system. The system included a single water pump for continuous water circulation, a valve connected to the pipes to control water flow through the hydroponics, and an automated neutralizer. The container tank contained a pH sensor, a water level sensor, and a temperature sensor. The peristaltic buffer device was attached to the plywood, where the silicone hose was connected to neutralizers, as shown in Figure 4.2.1.

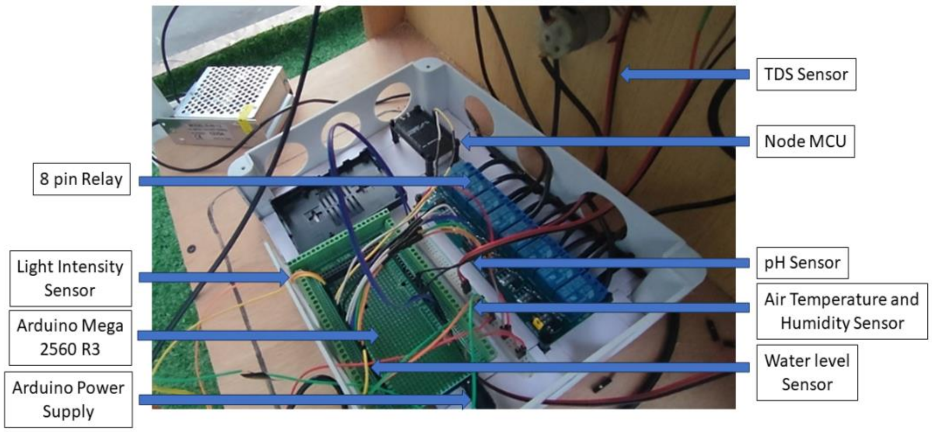


Figure 4.2.2 The connection of Arduino Mega 2560 R3 to ESP8266 NodeMCU

All the sensors, such as the pH sensor, TDS sensors, DHT22, GY-30, and water level sensor shown in Figure 4.2 were connected to an Arduino Mega 2560 R3. Serial communication was required when transferring sensor data and using a serial library to communicate with the RX and TX. As a result, sensor data was transmitted to the EPS8266 NodeMCU.

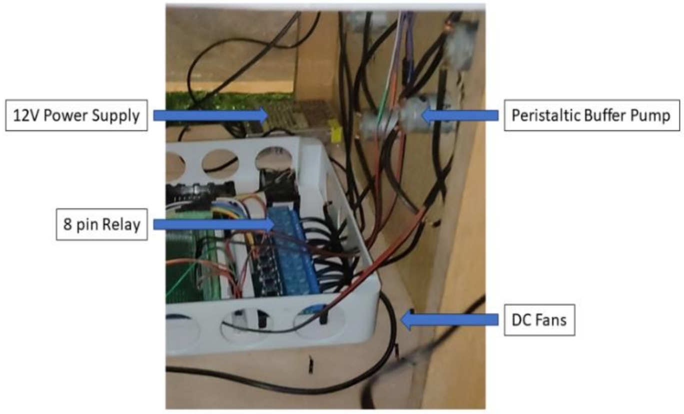


Figure 4.2.3 The connection of Actuators to Relay

The actuators were the 12V DC fan and the peristaltic pump connected to the relay. The delay was based on the gathered data programmed into the Arduino Mega 2560 R3; if the actuators did not reach the threshold set by the programmed sensors, it would turn on.

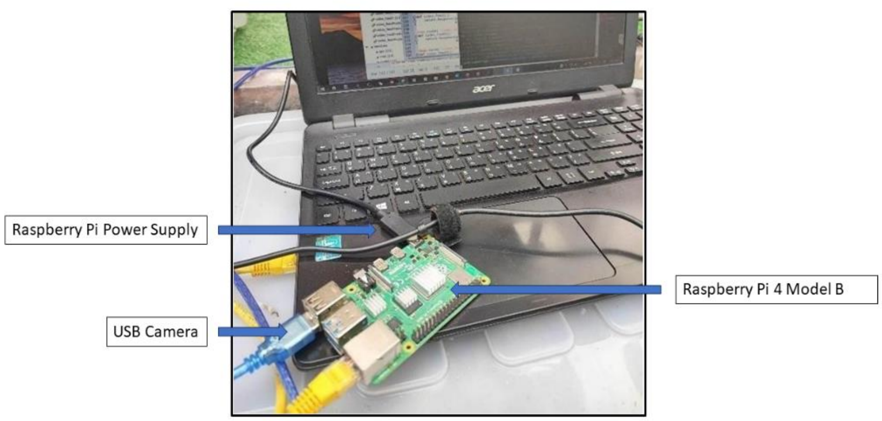


Figure 4.2.4 Connection of Cameras to Raspberry Pi 4 Model B

The 1080P full HD USB cameras required standard USB video class drivers supported by Linux and Windows operating systems to connect to the Raspberry Pi 4 Model B for disease detection through a mobile application and used an OpenCV for real-time monitoring through the web.

4.2.2 Mobile Application

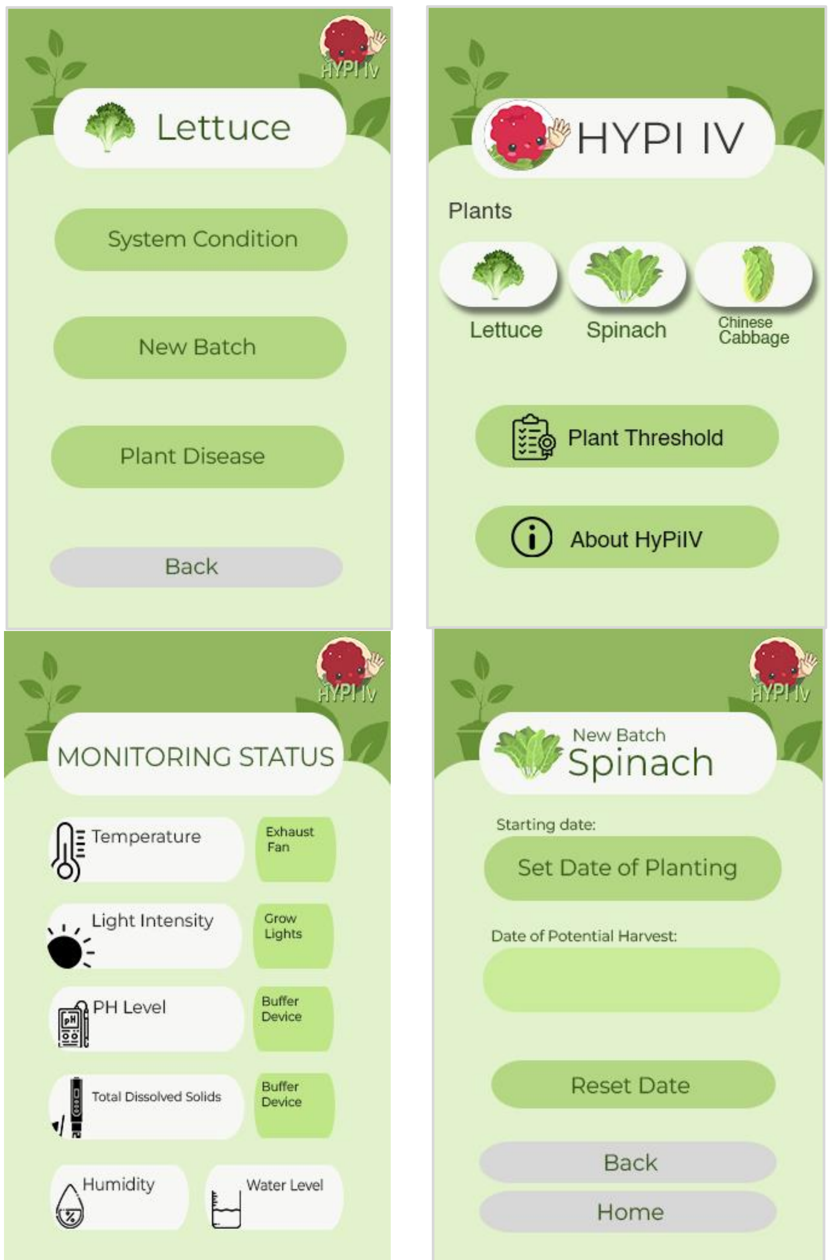


Figure 4.2.2 Pi-Droponics Mobile Application

Figure 4.2.2 shows the features of the Pi-Droponics Mobile Application. The mobile app featured a monitoring status of the sensors, plant disease detection, and a date potential harvest. Also, it indicated the plant threshold so the user can be aware of that.

4.3 Project Limitations and Capabilities

The project can detect and classify diseases in each plant and push notifications to users through a mobile app. Furthermore, the project's control system can alter critical plant growth parameters. The system included four cameras for plant monitoring to determine the health of the plants. The project limited the comparison of smart hydroponic plant growth to conventional soil-based farming of three leafy vegetables.

The following were the project's constraints:

1. The system can correct the critical pH and TDS level parameters ranging from 5.5 to 7.0 for pH and PPM values ranging from 560 - 2100.
2. The system can only grow a leafy vegetable of Spinach.
3. The mobile application is compatible with Android version 13.

4.4 Project Evaluation

All the data was collected at the deployment site in Barangay 786, Zone 86, District 5, Sta. Ana, Manila. The data has shown a comparison between Conventional Soil-based Farming and the Hydronics System, assessing the growing conditions of each plant, including temperature, light intensity, pH level, and TDS. Furthermore, the figures illustrated compared the two farming methods using parameters such as height, length, number of leaves, and color.

4.4.1 Accuracy of the trained model for Disease Detection

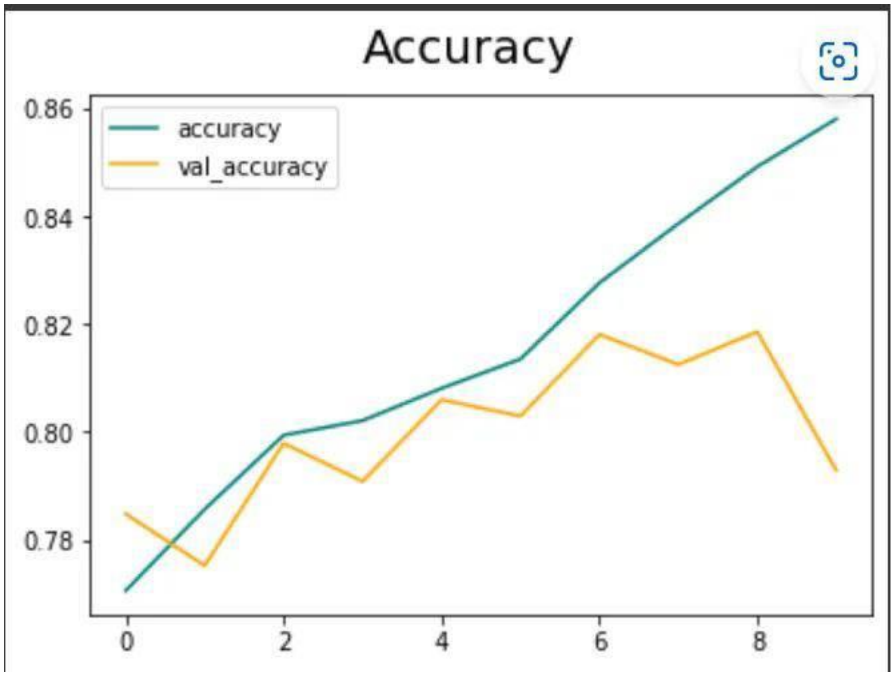


Figure 4.4.1 Accuracy of the trained model for Disease Detection

In Figure 4.4.1, the trained model's accuracy for Disease Detection was displayed. The validation accuracy was represented by the yellow line, while the blue line represented the training accuracy of the disease detection model. The accuracy of the disease detection model was utilized to evaluate and test the functionality using the sampled images and plants from the conventional and hydroponics system.

4.4.1 Disease Detection of Mobile Application

Figure 4.4.2 Disease Detection of Mobile Application

Figure 4.4.2 shows the result of the disease detection in the Mobile Application. Camera 1 displayed the plant detected a disease, and its identified disease, which is Downy Mildew; while cameras 2, 3, and 4 didn't detect any disease because the plant was not fully grown. It proved that the disease detection of the Pi-Droponics worked.

4.4.3 Air Temperature and Humidity

Table 4.4.3.1 Collected Data from Experiments using Digital Thermometer and DHT22 Sensor

Air Temperature and Humidity		
Trial	Digital Thermometer (°C)	DHT22 Sensor (°C)
1	23	29.2
2	28	29.2
3	32	29.3
4	34	32.6
5	30	29
6	27	32.6
7	33	29.7
8	26	29.3
9	26	33
10	34	29.5

Table 4.4.3.1 displays the recorded data comparing the two different devices: The Digital Thermometer for Conventional Soil-based farming and the DHT22 Sensor for Hydroponics Systems.

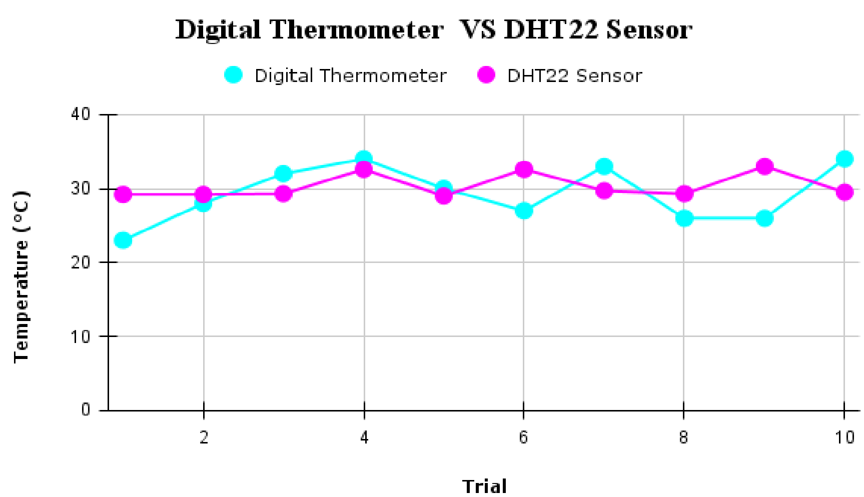


Figure 4.4.3.1 Comparison between Digital Thermometer vs. DHT22 Sensor

Figure **4.4.3.1** displays the collected data obtained from the Digital Thermometer and DHT22 Sensor. The data was taken with

10 trials. The comparison of the data revealed no significant differences between the two devices, as seen in the statistical comparison shown in Table 4.4.3.1.

Table 4.4.3.2 Statistical Comparison between Digital Thermometer vs. DHT22 Sensor

	Mean	SD	Variance	n	df	t-value	critical value
Digital Thermometer	29.3	3.8601	14.9	10	18	0.78	2.1
DHT22 Sensor	30.34	1.6655	2.7738	10			

Table 4.4.3.2 shows the statistical comparison of the Digital Thermometer and DHT22 Sensor, which included ten (10) trials with an hour interval. Based on the given data, the DHT22 sensor provided measurements with a higher mean temperature of 29.3, a lower standard deviation of 1.6655, and a variance of 2.77, lower than the digital thermometer. Both devices had the same data values of 10, 18 degrees of freedom, t-values of 0.78, and critical values of 2.1. The null hypothesis states that there is no significant difference in between using the Digital Thermometer and DHT22 Sensor.

4.4.4 Light Intensity

Table 4.4.4.1 Collected Data from Experiments using Light Intensity Meter and GY-30 BH1750FVI Module Digital Light Intensity Illumination Sensor

Light Intensity		
Trial	Light Intensity Meter (lux)	GY-30 BH1750FVI Module Digital Light Intensity Illumination Sensor (lux)
1	2152	2591.67
2	2153	2442.5
3	2654	2505
4	2153	2433.33
5	1365	1915.83
6	1323	2166.69
7	2284	2400
8	2465	2272.5
9	1983	2359
10	2152	2562.5

Table 4.4.4,.1 displays the recorded data comparing the two farming methods using different devices: The Light Intensity Meter for Conventional Soil-based farming and the GY-30 BH1750FVI Module Digital Light Intensity Illumination Sensor for Hydroponics Systems.

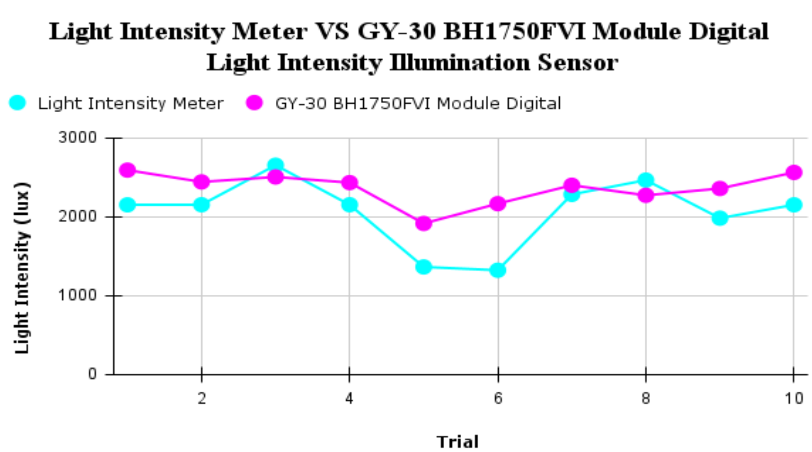


Figure 4.4.4.1 Comparison between Light Intensity Meter vs. GY-30 BH1750FVI Module Digital Light Intensity Illumination Sensor

Figure 4.4.4.1 displays the experimental data for Conventional Light Intensity Meter and BH1750FVI Light Intensity Sensor. The figure proved that there were no significant differences between the two devices' readings.

Table 4.4.4.2 Statistical Comparison between Light Intensity Meter vs. GY-30 BH1750FVI Module Digital Light Intensity Illumination Sensor

	Mean	SD	Variance	n	df	t-value	critical value
Light Intensity Meter	2068.4	425.9213	181408.9333	10	18	1.98	2.1
BH1750FVI Sensor	2364.902	203.1853	41284.2752	10			

Table 4.4.4.2 shows the statistical comparison between Light Intensity Meter vs. GY-30 BH1750FVI Module Digital Light Intensity Illumination Sensor, which included ten (10) trials with an hour interval. Based on the given data, it has shown that the GY-30 BH1750FVI Module Digital Light Intensity Illumination sensor provided measurements with a higher mean temperature of 2364.9 rather than the Light Intensity Meter, a lower standard deviation of 203.18 and a variance of 41284.2752 that is, lower compared to the Light Intensity Meter. Both devices had the same number of data values 10, 18 degrees of freedom, t-values of 1.98, and critical values of 2.1. The null hypothesis states that there is no significant difference in between using the Light Intensity Meter and GY-30 BH1750FVI Module Digital Light Intensity Illumination Sensor

4.4.5 pH Level

Table 4.4.5.1 Collected Data from Experiments using pH Meter and DFRobot pH Sensor v1.1

pH Level		
Trial	**pH Meter**	**DFROBOT pH Sensor v1.1**
1	7.82	6.5
2	6.83	6.4
3	7.23	7.24
4	7.55	6.59
5	7.57	7
6	7.6	6.4
7	7.87	6.61
8	6.98	7.02
9	6.87	6.58
10	7.45	6.59

Table 4.4.5.1 displays the recorded data comparing the two farming methods using different devices: the pH Meter for Conventional Soil-based farming and the DFROBOT pH Sensor for Hydroponics Systems.

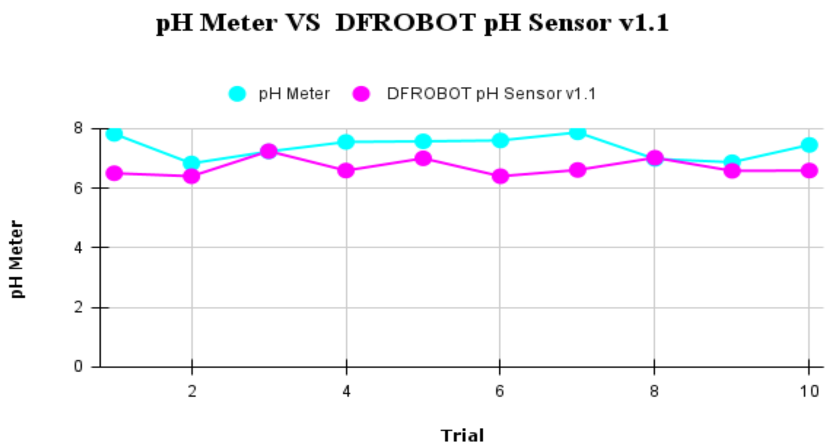


Figure 4.4.5.1 Comparison between pH Meter vs. DFRobot pH Sensor v1.1

Figure 4.4.5.1 displays the collected data obtained from the pH Meter and DFRobot pH Sensor v1.1. The data was collected using 10 trials. The difference in readings between the two devices indicated that there were significant differences between them, as shown in Table 4.6.

Table 4.4.5.2 Statistical Comparison between pH Meter vs. DFRobot pH Sensor v1.1

	Mean	SD	Variance	n	df	t-value	critical value
pH meter	7.377	0.3796	0.1441	10	18	4.53	2.1
pH sensor	6.693	0.2887	0.0834	10			

Table 4.4.5.2 shows the statistical comparison between pH Meter vs. DFRobot pH Sensor v1.1, which includes ten (10) trials with an hour interval. Based on the given data, the pH sensor provided measurements with a lower mean temperature of 7.377 compared to the pH meter, a lower standard deviation of 0.2887, and a variance of 0.0834, lower than the pH meter. Both devices had the same data values of 10, 18 degrees of freedom, t-values of 4.53, and critical values of 2.1. The alternative hypothesis states that there is significant difference in

between using the pH Meter and DFRobot pH Sensor.

4.4.6 Total Dissolved Solid Sensor

Table 4.4.6.1 Collected Data from Experiments using TDS Meter and DFRobot Analog Total Dissolved Solid Sensor

Total Dissolve Solution (TDS)		
Trial	TDS Meter	DFRobot Analog Total Dissolved Solid Sensor
1	134	239
2	109	239
3	111	356
4	117	1362
5	116	1225
6	116	1362
7	176	1157
8	143	1043
9	112	1061
10	126	982

Table 4.4.6.1 presents the gathered data, which compared the performance of two farming methods using distinct devices: the TDS Meter utilized for Conventional Soil-based farming and the DFROBOT Analog Total Dissolved Solid Sensor employed in Hydroponics System

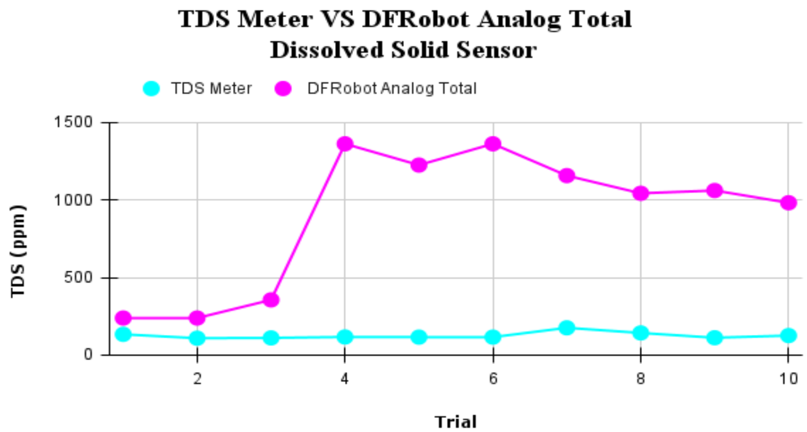


Figure 4.4.6.1 Comparison between TDS Meter vs. DFRobot Analog Total Dissolved Solid Sensor

Figure 4.4.6.1 exhibits the data acquired from the TDS Meter and DFRobot Analog Total Dissolved Solid Sensor. The data collection spanned 10 trials, with varying days sampled from each trial. The DFRobot Analog Total Dissolved Solid Sensor demonstrated higher parts per million (ppm) readings, attributed to its calibrated threshold tailored for the particular plant requirements. Nevertheless, Table 4.8 illustrates a substantial disparity in readings between the two devices.

Table 4.4.6.2 Statistical Comparison between TDS Meter VS DFRobot Analog Total Dissolved Solid Sensor

	Mean	SD	Variance	n	df	t-value	critical value
TDS meter	127.125	22.2995	497.2679	10	18	9.52	2.1
TDS sensor	1068.5	320.8146	102922	10			

Table 4.4.6.2 shows the statistical comparison between TDS Meter vs. DFRobot Analog Total Dissolved Solid Sensor, which included ten (10) trials with an hour

interval. Based on the given data, it has shown that the sensor provided measurements with a lower mean temperature of 1068.5 compared to the TDS Meter, a higher standard deviation of 320.81.46, and a variance of 102922 that is higher than the TDS Meter. Both devices had the same data values of 10, 18 degrees of freedom, t-values of 9.52, and critical values of 2.1. The alternative hypothesis states that there is significant difference in between using the TDS Meter and DFRobot Analog Total Dissolved Solid Sensor

4.4.7 Comparison of Spinach Growth between Hydroponics Systems and Conventional Soil-based Farming

4.4.7.1 Length of Spinach Plant

Table 4.4.7.1 Comparison of Length of the Spinach in Conventional Method and Hydroponics System

Comparison of Length of the Spinach in Conventional Method and Hydroponics System		
Weeks	**Conventional Method (cm)**	**Hydroponics System (cm)**
1	0.9	0.9
2	1.86	2.1
3	3.93	3.9
4	4.26	7.23
5	4.43	7.4
6	6.67	10.85

Table 4.4.7.1 presents the gathered data for comparing the length of the Spinach in the Conventional Method and the Hydroponics System for six (6)

weeks. As shown above, within six (6) weeks, the length of Spinach grew bigger by 10.85 cm using the Hydroponics system rather than a Conventional Method that grows only 6.67cm.

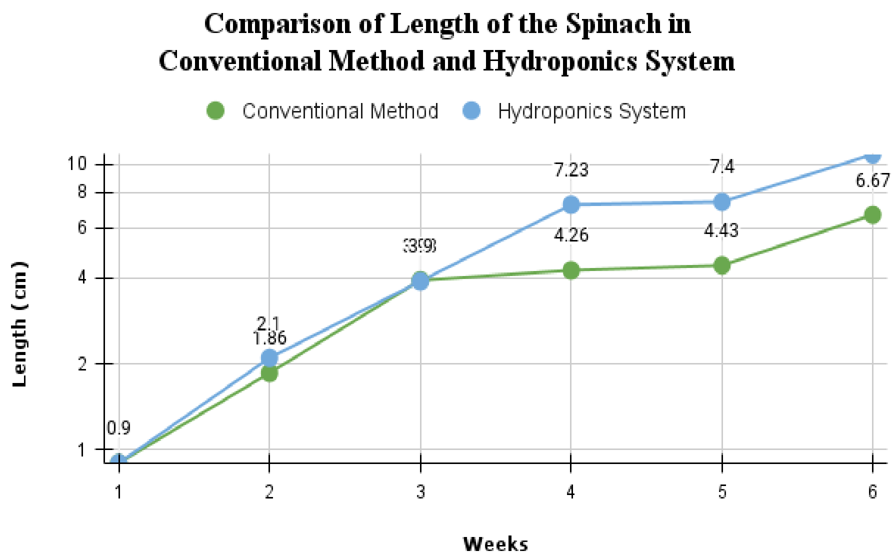


Figure 4.4.7.1 Length of the Spinach in Conventional Method vs. Hydroponics System

The graph displays the comparative analysis of the Spinach length in the Conventional Method and Hydroponics System in six (6) weeks. Notably, significant differences in plant length between the two methods became apparent starting from week four (4) up to week six (6), or the harvest time for the Spinach. It has shown the effectiveness of the Hydroponics System as a method in farming due to the faster growth of plants.

Table 4.4.7.2 Statistical Comparison of Length of the Spinach in Conventional Method and Hydroponics System

	Mean	SD	Variance	n	df	t-value	critical value
Conventional	3.675	2.0466	4.1884	6	10	0.9865	2.23
Hydroponics	5.3967	3.7534	14.0882	6			

The collected data obtained from measuring the length of spinach plants underwent statistical analysis. A t-test was conducted to compare the mean lengths of spinach plants in the conventional and hydroponic method. The null hypothesis states that there is no significant difference between the length of the spinach plant in conventional and hydroponic method. The calculated t-value was determined to be below the critical value at a significance level of 0.05, indicating that there is no statistically significant difference in the length of spinach plants between the two cultivation methods.

4.4.7.2 Height of Spinach Plant

Table 4.4.7.3 Comparison of Height of the Spinach in Conventional Method and Hydroponics System

Comparison of Height of the Spinach in Conventional Method and Hydroponics System		
Weeks	**Conventional Method (cm)**	**Hydroponics System (cm)**
1	2.33	2.3
2	3.86	4.96
3	6.16	7.9
4	7.88	10.67
5	8.76	15.46
6	12.33	19.5

Table 4.4.7.3 presents the gathered data for comparing the height of the Spinach in the Conventional Method and the Hydroponics System for six (6) weeks. As shown above, within six (6) weeks, the height of Spinach had grown taller by 19.5 cm using the Hydroponics system rather than a Conventional Method that only grew 12.33cm in height.

Table 4.4.7.4 Statistical Comparison of Height Spinach in Conventional Method and

	Mean	SD	Variance	n	df	t-value	critical value
Conventional	6.8867	3.5938	12.9156	6	10	1.0736	2.23
Hydroponics	10.1317	6.4731	41.9016	6			

Hydroponics System

The collected data obtained from measuring the height of spinach plants underwent statistical analysis. A t-test was conducted to compare the mean lengths of spinach plants in the conventional and hydroponic method. The null hypothesis states that there is no significant difference between the length of the spinach plant in conventional and hydroponic method. The calculated t-value was determined to be below the critical value at a significance level of 0.05, indicating that there is no statistically significant difference in the length of spinach plants between the two cultivation methods.

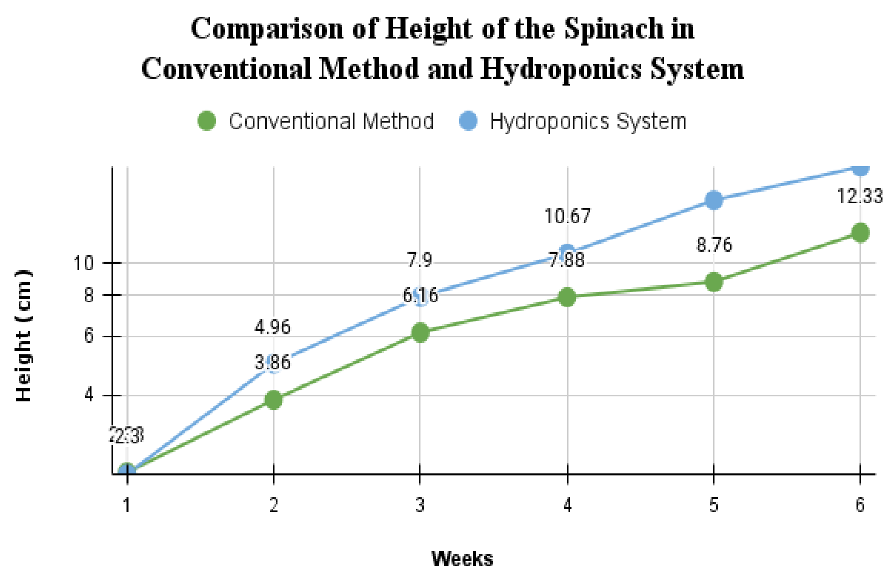


Figure 4.4.7.4 Height of the Spinach in Conventional Method vs. Hydroponics System

Figure 4.4.7.4 presents a comparative assessment of the Height of Spinach in the Conventional Method and Hydroponics System for six (6) weeks. The data reveals consistent differences observed in the heights of plants between both methods. It shows the faster growth of the Spinach using the Hydroponics system, which implies the effectiveness of the system.

4.4.7.3 Number of Leaves of Spinach Plant

Table 4.4.7.5 Number of Leaves of Spinach in Conventional Method and Hydroponics System

Comparison of Number of Leaves of the Spinach in Conventional Method and Hydroponics System		
Weeks	**Conventional Method**	**Hydroponics System**
1	4.33	4.33
2	9.67	11.67
3	14	22.67
4	18	43.33
5	24.67	59
6	33.33	81.67

Table 4.4.7.5 presents the gathered data for comparing the number of leaves of Spinach in the Conventional Method and the Hydroponics System for six (6) weeks. As shown above, within six (6) weeks, the number of Spinach grows more leaves, having an average of 81.67 in total, using the Hydroponics system rather than a Conventional Method that grows fewer leaves, having an average of 33.33 in total.

Table 4.4.7.6 Statistical Comparison of Number of Leaves of Spinach in Conventional Method and Hydroponics System

	Mean	SD	Variance	n	df	t-value	critical value
Conventional	17.3333	10.4794	109.8178	6	10	1.5347	2.23
Hydroponics	37.1117	29.7782	886.7379	6			

The collected data obtained from measuring the number of leaves of spinach plants underwent statistical analysis. A t-test was conducted to compare the mean lengths of spinach plants in the conventional and hydroponic method. The null hypothesis states that there is no significant difference between the length of the spinach plant in conventional and hydroponic method. The calculated t-value was determined to be below the critical value at a significance level of 0.05, indicating that there is no statistically significant difference in the length of spinach plants between the two cultivation methods.

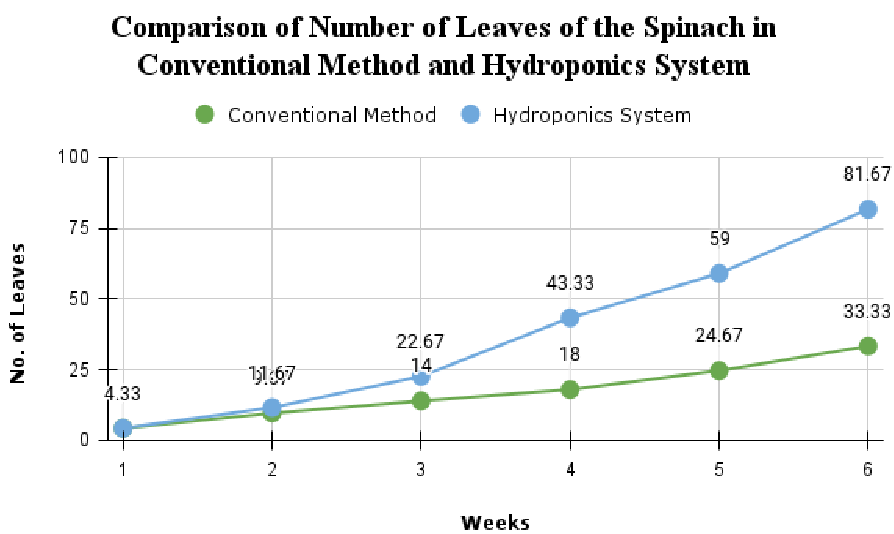


Figure 4.4.7.6 Number of Leaves of Spinach in Conventional Method and Hydroponics System

Figure 4.4.7.6, The graphical presentation for the comparison of the number of leaves for the Spinach plant, shows the Hydroponics System's higher yield than the Conventional Method. In six (6) weeks, which corresponds to the harvest period, it can be seen the differences in the numbers of leaves immediately starting from week two (2). It proves the

Hydroponics system can produce a higher yield faster than the Conventional Method.

4.4.7.4 Color of the Leaves of Spinach Plant

Table 4.4.7.7 Color of the Leaves of Spinach in Conventional Method and Hydroponics System

Comparison of Color of the Spinach in Conventional Method and Hydroponics System		
COLOR	CONVENTIONAL	HYDROPONICS
Strong Green	4	4
Green	2	3
Yellowish	2	1

Table 4.4.7.7 shows the gathered data for comparing the Color of the Leaves of the Spinach in the Conventional Method and the Hydroponics system. There were no changes in colors of the Spinach plants after six (6) weeks for both methods.

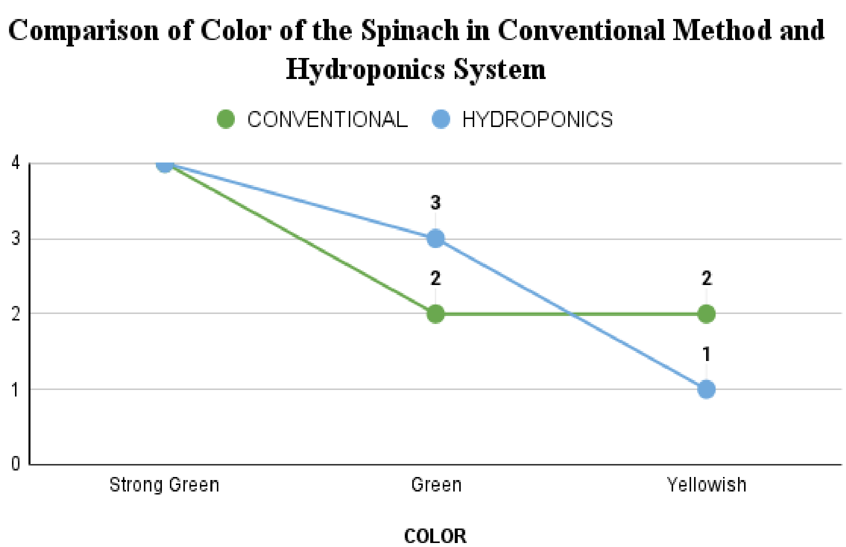


Figure 4.4.7.7 Color of the Leaves of Spinach in Conventional Method and Hydroponics System

Figure 4.4.7.7 presents a comparative evaluation of the color of the Spinach in the Conventional Method and Hydroponics System. It was shown that the color of the Spinach is not significantly different in both Conventional Method and Hydroponics System. There were no changes in color; strong green when it reached the harvest period within six (6) weeks.

4.4.8 User's Evaluation

Table 4.4.8 User's Evaluation from Barangay 786 Zone 86, Sta. Ana, Manila Residents

STANDARDS	INTEPRETATION	AVERAGE
Functional Suitability	Excellent Performance	3.50
Performance Efficiency	Excellent Performance	3.48
Compatibility	Excellent Performance	3.52
Usability	Excellent Performance	3.64
Reliability	Excellent Performance	3.66
Security	Excellent Performance	3.68
Maintainability	Excellent Performance	3.55
Portability	Excellent Performance	3.50
Average	**Excellent Performance**	**3.57**

According to Table 4.4.8, with 44 respondents, the mean for each given standard from the Information Technologists' Exploration and Technical Evaluation quality of Developed Hydroponic System Based on ISO 25010 Software, for the Functional Suitability is three-point fifty (3.50), Performance

Efficiency is three-point forty-eight (3.48), Compatibility is three-point fifty-two (3.52), Usability is three-point sixty-four (3.64), Reliability is three-point sixty-six (3.66), Security is three-point sixty-eight (3.68), Maintainability is three-point fifty-five (3.55), and lastly Portability is three-point fifty (3.50). The mean for each standard fall under the range of four (4) or Excellent Performance with the range of 106 three-point twenty-six to four (3.26-4). In conclusion, the average answer from the respondents is three-point fifty-seven (3.57), which means that the Pi-Droponics falls under the interpretation of Excellent Performance based on ISO 25010

Chapter 5

SUMMARY OF FINDINGS, CONCLUSION AND RECOMMENDATIONS

5.1 Summary of Findings

The utilization of sensors for monitoring and actuators for adjusting conditions in a hydroponics system has had a significant positive impact on plant production, as evidenced by the comparison of yield in hydroponics versus conventional farming. When considering the environment within the greenhouse where the hydroponics system is located, accurate readings of pH level and TDS level had proven to be effective in monitoring the environmental parameters. Additionally, the use of air humidity sensors, temperature sensors, and light intensity sensors has shown that the greenhouse environment was not significantly different from the outside environment.

Traditional methods of periodically inspecting plant growth and directly measuring their requirements were labor-intensive and time-consuming. In contrast, the implementation of a remote monitoring and notification system through a mobile application has greatly enhanced the effectiveness of the hydroponics system, surpassing other similar hydroponics projects.

Overall, the integration of sensors, actuators, and remote monitoring technology has revolutionized the efficiency and productivity of hydroponics systems, providing accurate and real-time data for optimal plant growth and reducing the need for manual labor and time-consuming processes.

5.2 Conclusion

The goal of the project study was to develop an automated vertical hydroponics system with monitoring and control, along with a notification system in the form of a mobile application. This application sends information whenever the system deviates from specified parameters, detects plant diseases, or estimates that the plants may be ready for harvest.

A circuit comprising various actuators and sensors was designed and implemented to monitor and control the vertical hydroponics system. The circuit includes components such as a light intensity sensor, total dissolved solids (TDS) sensor, pH sensor, temperature sensor, humidity sensor, water level sensor, grow lights, peristaltic buffer device, and water pump. The Arduino Mega served as the sensor node, effectively regulating and monitoring the necessary factors for plant growth. The peristaltic buffer device and water pump were used for controlled nutrient and water release.

A notification system integrated with a mobile application was developed to alert the user when the system parameters exceeded defined thresholds. It also provides notifications for plants exhibiting signs of diseases and plants that may be ready for harvesting based on estimated maturity days. This system ensured timely responses, helped prevent disease-related damage, and enabled optimal harvesting, thereby enhancing the efficiency and effectiveness of the hydroponics system.

A disease detection model using convolutional neural networks (CNN) was successfully developed. Although the model was tested using leaves from plants outside the system, as the plants within the system remained disease-free, it demonstrated the functionality and effectiveness of the model. While the disease detection model was not directly tested with the system's plants, their healthy state supported the overall effectiveness of the system.

5.3 Recommendations

To further improve this study, the proponents suggest the following actions. Firstly, conduct future research involving a wider range of plants, including Chinese Cabbage and Lettuce, in addition to the Spinach plant studied in this paper. This will provide a more comprehensive understanding of the system's performance. Allocating sufficient time and resources to ensure effective planting of all target plants is essential for obtaining reliable results.

To enhance the disease detection capabilities of the model, it is recommended to expand the range of diseases considered beyond the four studied such as Alternaria Leaf Spot, Anthracnose, Downy Mildew, and White Rust Disease. This expansion will increase the model's practical value for agricultural applications. Additionally, incorporating darkness and whiteness shades from the dataset images can significantly improve the model's accuracy in analyzing and distinguishing between healthy and diseased plant, leading to more precise disease detection.

Furthermore, if the project expands, it would be beneficial to develop and integrate the software component on a PC. This allows for scalability, easier system management, and enhanced user interface capabilities. Lastly, regular upgrading and calibration of the sensors and actuators used in the study are recommended to minimize potential issues, ensuring accurate and effective monitoring of plant growth.

References:

Admin. (2020, May 21). Send Sensor Data to Android Using Google Firebase & ESP8266. How to Electronics. https://how2electronics.com/send-sensor-data-android-google-firebase-esp8266

Aiswarya, B., Sharma, A., Chakraborty, R., & Kumar, R. (2019). Image Processing Based Edibility Analysis of Spinach Leaves using Machine Learning Approach-BY-NC-ND license http://creativecommons.org/licenses/by-nc-nd/4.0. International Journal of Recent Technology and Engineering (IJRTE), 7, 2277–3878. https://www.ijrte.org/wp-content/uploads/papers/v7i6/F2249037619.pdf .

Albawi, S., Mohammed, T. A., & Al-Zawi, S. (2017). Understanding of a Convolutional Neural Network. 2017 International Conference on Engineering and Technology (ICET), 1–6. https://doi.org/10.1109/icengtechnol.2017.8308186

Alternaria Leaf Spot of Brassicas | Cornell Vegetables. (n.d.). https://www.vegetables.cornell.edu/crops/brassicas-broccoli-cabbage-cauliflower-and-brussels-sprouts/alternaria-leaf-spot-of-brassicas/#:~:text=Introduction,lead%20to%20an%20unmarketable%20crop

Al-Rawahy, M. S., Al-Rawahy, S. A., Al-Mulla, Y. A., & Nadaf, S. K. (2018). Effect of Cooling Root-Zone Temperature on Growth, Yield and Nutrient Uptake in Cucumber Grown in Hydroponic System During Summer Season in Cooled Greenhouse. Journal of Agricultural Science, 11(1), 47. https://doi.org/10.5539/jas.v11n1p47

Anthracnose Management Guidelines--UC IPM. (n.d.). https://ipm.ucanr.edu/PMG/PESTNOTES/pn7420.html#:~:text=Anthracnose%20is%20a%20term%20used,cankers%20on%20twigs%20and%20stems

Arshad, J., Aziz, M., Al-Huqail, A. A., Zaman, M. H., Husnain, M., Rehman, A. U., & Shafiq, M. (2022b). Implementation of a LoRaWAN Based Smart Agriculture Decision Support System for Optimum Crop Yield. *Sustainability*, *14*(2), 827. https://doi.org/10.3390/su14020827

Canlas, J. (2020, July 14). *Urban Agriculture: Coping with crisis towards Food Security*. Special Area for Agricultural Development. https://saad.da.gov.ph/2020/07/urban-agriculture-coping-with-crisis-towards-food-security-5

Carlo, J., Haban, I., Alejandrino, J., Bandala, A. A., & Dadios, E. P. (2020). Design of A Nutrient Film Technique Hydroponics System with Fuzzy Logic Control. https://doi.org/10.1109/tencon50793.2020.9293749

Chhetri, S., Dulal, S., Subba, S., & Gurung, K. (2022). Effect of different growing media on growth and yield of leafy vegetables in nutrient film technique hydroponics system. Archives of Agriculture and Environmental Science, 7(1), 12–19. https://doi.org/10.26832/24566632.2022.070103

Color Space Analysis Using KNN for Lettuce Crop Stages Identification in Smart Farm Setup. (2018, October 1). IEEE Conference Publication | IEEE Xplore. https://ieeexplore.ieee.org/document/8650209

Comparison between growing plants in hydroponic system and ... - avestia. (n.d.).http://avestia.com/MCM2018_Proceedings/files/paper/ICMIE/ICMIE_131.pdf

Dunn, B., & Singh, H. (2016, October). Electrical Conductivity and pH Guide for Hydroponics - researchgate. https://www.researchgate.net/publication/309293679_Electrical_Conductivity_and_pH_Guide_for_Hydroponics

Ghael, H., Dipak Ghael, H., Solanki, L., Sahu, G., & Professor, A. (2008). A Review Paper on Raspberry Pi and its Applications. International Journal of Advances in Engineering and Management (IJAEM, 2(February), 225. https://doi.org/10.35629/5252-0212225227.

Kuncoro, C. B. D., Asyikin, M. B. Z., & Amaris, A. (2021). Development of an Automation System for Nutrient Film Technique Hydroponic Environment. In *Advances in engineering research*. Atlantis Press. https://doi.org/10.2991/aer.k.211106.069

Lean, K. S. T., Rachelle, B. D. S., Nikka, M. V. E., Wendel, G. M., Rafael, Q. S. a. K., Raymond, M. T. R., Gilfred, A. M. M., Romeo, L. J., Fel, Q. J., Cherry, G. P., & Jessica, S. V. (2021). HyLo: Implementation of LoRaWAN in an Automated Hydroponics System. In *2021 IEEE International Conference on Mobile Networks and Wireless Communications (ICMNWC)*. https://doi.org/10.1109/icmnwc52512.2021.9688540

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep Learning. Nature, 521(7553), 436–444. https://doi.org/10.1038/nature14539

Maboko, M. M., & Du Plooy, C. P. (2018). Yield response of hydroponically grown mustard spinach and non-heading Chinese cabbage to frequency of leaf harvest and flower removal. International Journal of Vegetable Science, 25(2), 185–195. https://doi.org/10.1080/19315260.2018.1496511

Michael, G. W., Tay, F. S., & Then, Y. L. (2021). Development of Automated Monitoring System for Hydroponics Vertical Farming. *Journal of Physics: Conference Series*, *1844*(1), 012024. https://doi.org/10.1088/1742-6596/1844/1/012024

Murugan, M. B., Rajagopal, M. K., & Roy, D. (2021, November 1). *IOPscience*. Journal of Physics: Conference Series. https://iopscience.iop.org/article/10.1088/1742-6596/2115/1/012017

Musa, A., Hassan, M., Hamada, M., & Aliyu, F. (2022). Low-Power Deep Learning Model for Plant Disease Detection for Smart-Hydroponics Using Knowledge Distillation Techniques. Journal of Low Power Electronics and Applications, 12(2), 24. https://doi.org/10.3390/jlpea12020024

NodeMCU ESP8266 Specifications, Overview and Setting Up. (2021, September 15). https://www.make-it.ca/nodemcu-details-specifications/#:~:text=The%20ESP8266%2C%20designed%20and

Nutrient film technique. (n.d.). DBpedia. https://dbpedia.org/page/Nutrient_film_technique

Patil, N., Patil, S., Uttekar, A., & Suryawanshi, A. R. (2020). Monitoring of Hydroponics System using IoT Technology. International Research Journal of Engineering and Technology (IRJET), 07(06, June), 1455– 1458.

Python Software Foundation. (2019). What is Python? Executive Summary. Python.org; Python.org. https://www.python.org/doc/essays/blurb/

Pramono, S., Nuruddin, A., & Hamka, M. (2020, April). Design of a Hydroponic Monitoring System with Deep Flow Technique (DFT). https://www.researchgate.net/publication/340633543_Design_of_a_Hydroponic_Monitoring_System_with_Deep_Flow_Technique_DFT

Prasetia, Y., Putrada, A. G., & Rakhmatsyah, A. (2021). Evaluation of IoT-Based Grow Light Automation on Hydroponic Plant Growth. Jurnal Ilmiah Teknik Elektro Komputer Dan Informatika, 7(2), 314. https://doi.org/10.26555/jiteki.v7i2.21424 .

Pratama, I. Y., Wahab, A., & Alaydrus, M. "Deep Learning for Assessing Unhealthy Lettuce Hydroponic Using Convolutional Neural Network based on Faster R-CNN with Inception V2," pp. 4–9.

Prathibha, S. R., Hongal, A., & Jyothi, M. C. (2017b). *IOT Based Monitoring System in Smart Agriculture*. https://doi.org/10.1109/icraect.2017.52

Sardare, D. M. (2013). A REVIEW ON PLANT WITHOUT SOIL - HYDROPONICS. *International Journal of Research in Engineering and Technology*, *02*(03), 299–304. https://doi.org/10.15623/ijret.2013.0203013

Siregar, B., Efendi, S., Pranoto, H., Ginting, R. B., Andayani, U., & Fahmi, F. (2017). *Remote monitoring system for hydroponic planting media*. https://doi.org/10.1109/ictss.2017.8288884

SITNFlash. (2019, October 4). *Hydroponics: The power of water to grow food - Science in the News*. Science in the News. https://sitn.hms.harvard.edu/flash/2019/hydroponics-the-power-of-water-to-grow-food/

The Trustees of Princeton University. (n.d.). *The future of farming: Hydroponics - PSCI*. Princeton University. https://psci.princeton.edu/tips/2020/11/9/the-future-of-farming-hydroponics

Tolentino, L. K. S., Fernandez, E. O., Amora, S. N. D., Bartolata, D. K. T., Sarucam, J. R. V., Sobrepeña, J. C. L., & Sombol, K. Y. P. (n.d.). *Yield evaluation of brassica rapa, Lactuca sativa, and brassica integrifolia using image processing in an IOT-based aquaponics with temperature-controlled greenhouse*. AGRIVITA, Journal of Agricultural Science. https://agrivita.ub.ac.id/index.php/agrivita/article/view/2600/1276

Touliatos, D., Dodd, I. C., & McAinsh, M. (2016). Vertical farming increases lettuce yield per unit area compared to conventional horizontal hydroponics. Food and Energy Security, 5(3), 184–191. https://doi.org/10.1002/fes3.83

UC IPM. (2020, May). *How to manage pests*. UC IPM Online. https://ipm.ucanr.edu/PMG/PESTNOTES/pn7420.html

What is Camera Module? - Supertek. (2020, October 8). https://www.supertekmodule.com/what-is-camera-module

White Rust - Bugwoodwiki. (n.d.). https://wiki.bugwood.org/HPIPM:White_Rust.

Wifi Module - ERC Handbook. (n.d.). Erc-Bpgc.github.io. https://erc-bpgc.github.io/handbook/electronics/Modules/wifi_module/#:~:text=Wifi%20modules%20or%20wifi%20microcontrollers Yang, R., Wu, Z., Wentai, F., Zhang, H. Y., Wang, W., Fu, L., Majeed, Y., Li, R., & Cui, Y. (2023). Detection of abnormal hydroponic lettuce leaves based on image processing and machine learning. *Information Processing in Agriculture*, *10*(1), 1–10. https://doi.org/10.1016/j.inpa.2021.11.001

Yekti Puji Lestari, Helilusiatiningsih, N., & Eva Pebriana. (2022). Response of Liquid Organic Fertilizer and Type of Media on Pakcoy (Brassica Rapal.) Production by Wick Hydroponics. Journal of Soilscape and Agriculture, 1(1), 32–39. https://doi.org/10.19184/jsa.v1i1.129

Yudha Pratama, I., Wahab, A., & Alaydrus, M. (2020). Deep learning for assessing unhealthy lettuce hydroponic using convolutional neural networks based on faster R-CNN with Inception V2. 2020 5th International Conference on Informatics and Computing, ICIC 2020, 4–9. https://doi.org/10.1109/ICIC50835.2020.9288554.

ANNEX I

Bill of Materials

Quantity	Specification	Description	Amount (Php)
1	4GB, 5V	Raspberry Pi 4B	₱7,200.00
1	5V	Arduino Mega 2560 Rev3	₱850.00
1	5V	Air and Humidity DHT22 Sensor	₱200.00
1	5V	Light Intensity Sensor GY-30 BH1750FVI	₱90.00
1	5V	DFRobot Analog TDS Sensor/Meter	₱875.00
1	5V	DFRobot Gravity: Analog pH Sensor	₱1800.00
1	5V	Water Level Sensor	₱40.00
2	5V	LED Grow Lights	₱874.00
1	5V, 10A	8-Channel 5V Relay Module	₱200.00
6	12V	Peristaltic Buffer Device	₱2100.00
4	12V	DC Fan	₱480.00
1	5V	NodeMCU V3 ESP8266 ESP-12E	₱160.00
1	36W	Water Pump DC Motor	P850.00
4		USB Camera	₱1700.00
1	12V, 5A	AC/DC Switching Power Supply	₱490.00
1	32GB	SD Card	₱450.00
1		Junction Box	₱350.00
50	30cm	Jumping Wires (Male to Male)	₱140.00
50	30cm	Jumping Wires (Male to Female)	₱140.00
1	5m	Copper Wire	₱125.00
1	7m	Silicone Tube	₱700.00
1		Hydroponic System	₱4895.00
1		Greenhouse	₱6020.00
1		Water Reservoir	₱400.00
20		Net Cup	₱400.00
1		Nutrient Solution	₱500.00
1		Ph Up and Down	₱410.00
4		Sponge	₱50.00
1		Seeds	₱65.00
1		Conventional Farming	₱240.00
TOTAL			**₱31,944.00**

ANNEX II

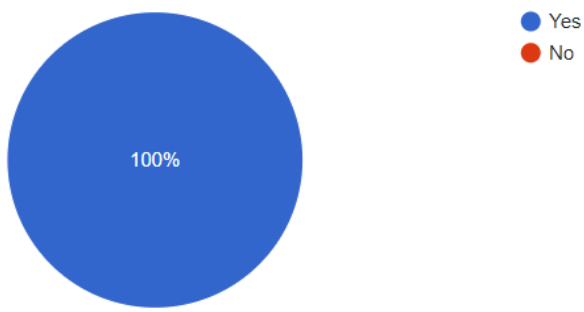
Survey

DATA PRIVACY

Copy

In compliance with the Data Privacy Act (DPA) of 2012, and it's implementing Rules and Regulations (IRR) effective since September 9, 2016, I allow the the researchers to store and use my data for the purpose and execution of operations of the Survey form entitled "HyPi-IV Survey Form".

44 responses



Name (Last Name, First Name, Middle Initial)

44 responses

Chavez, Ofelia I.

Escalderon, Ciara J.

Nelson E. Esmeña

Esmele, Jessa M.

Pansacola, Jose

Joper Pansacola

Medina, Rafael S.

Christoper Muan

Santiago, Helen T.

Name (Last Name, First Name, Middle Initial)

44 responses

Sunshine Delos Reyes

Magnaye, Joseph K.

Charlene Aquino

Ramos, Alfonso V.

Elias Curtis Dagala Celiz

Aguinaldo, Rolando O.

Kelvin Solomon Quimson Rubio

Ruby Kelli Akbar Magsakay

Maningas, Gabrielle J.

Name (Last Name, First Name, Middle Initial)

44 responses

Sofia Mangubat

Cruz, Patricia M.

Pia Philips

Castro, Jennilyn L.

Milagrosa Dalia Butil Silvestre

Latrell Montes Reotutar Dimalanta

Kim Set

Cariaso, Andrei J.

Johniel Singson

Name (Last Name, First Name, Middle Initial)

44 responses

Ian Suaviso

Aquino, Connie V.

Johann Medina

Magnaye, Emerald K.

Felipe Jose

Hanz Kobe

Lance Craig

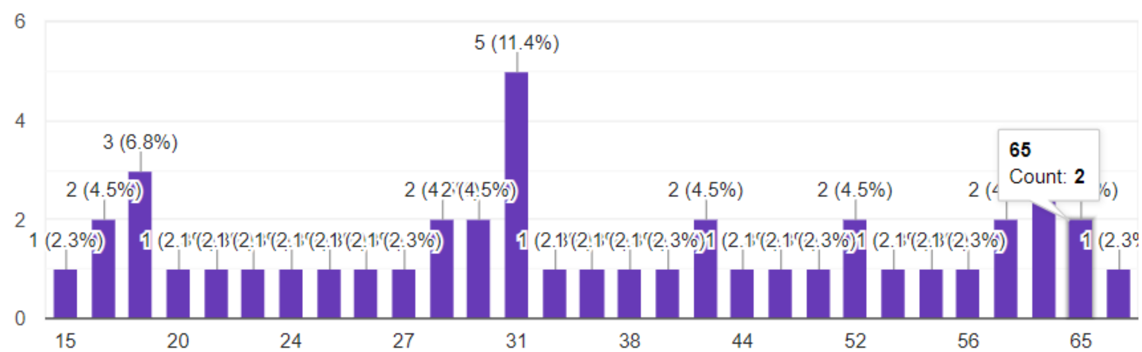
Rizal Santos

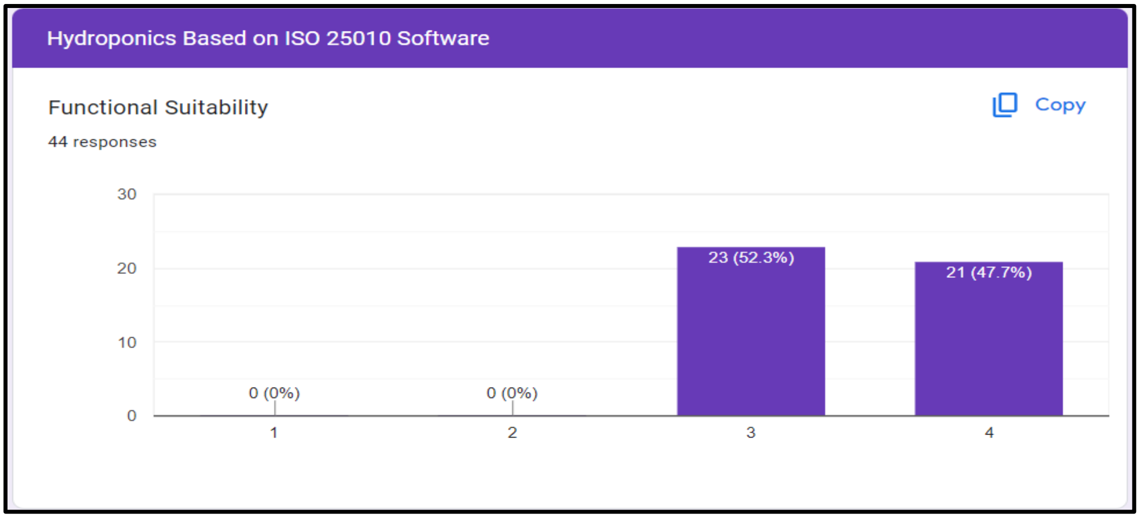
Dela Cruz, John

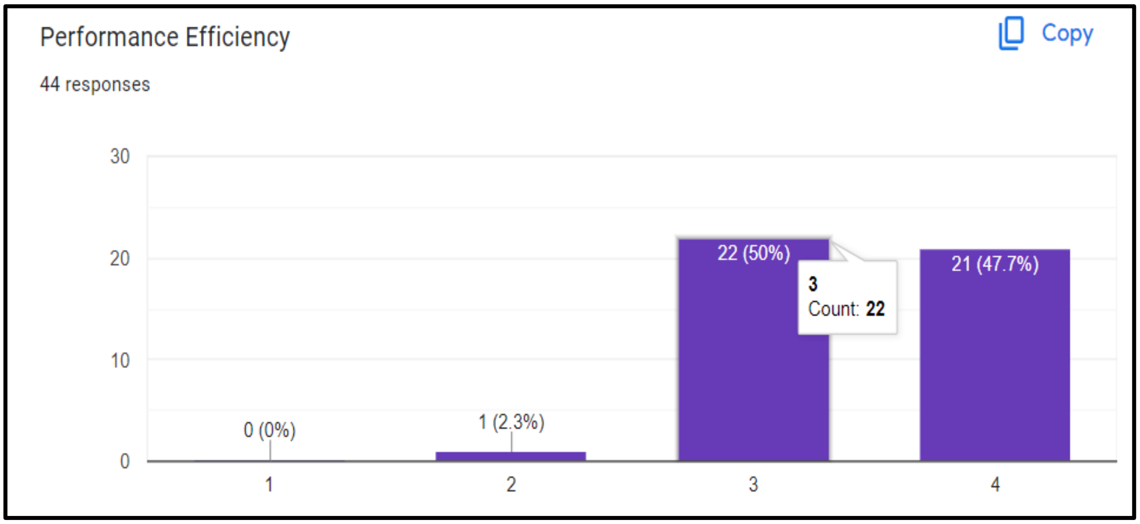
Age

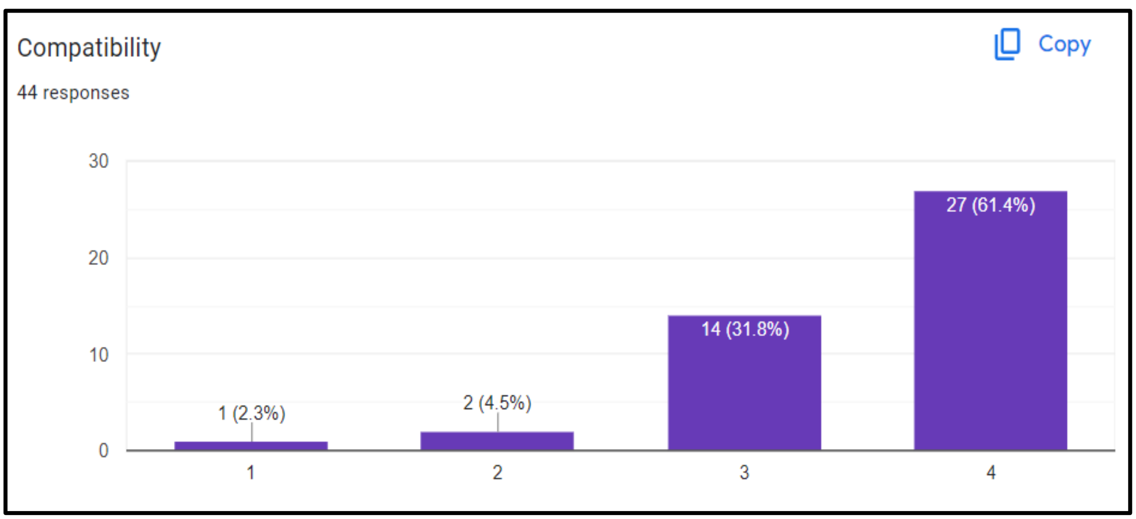
Copy

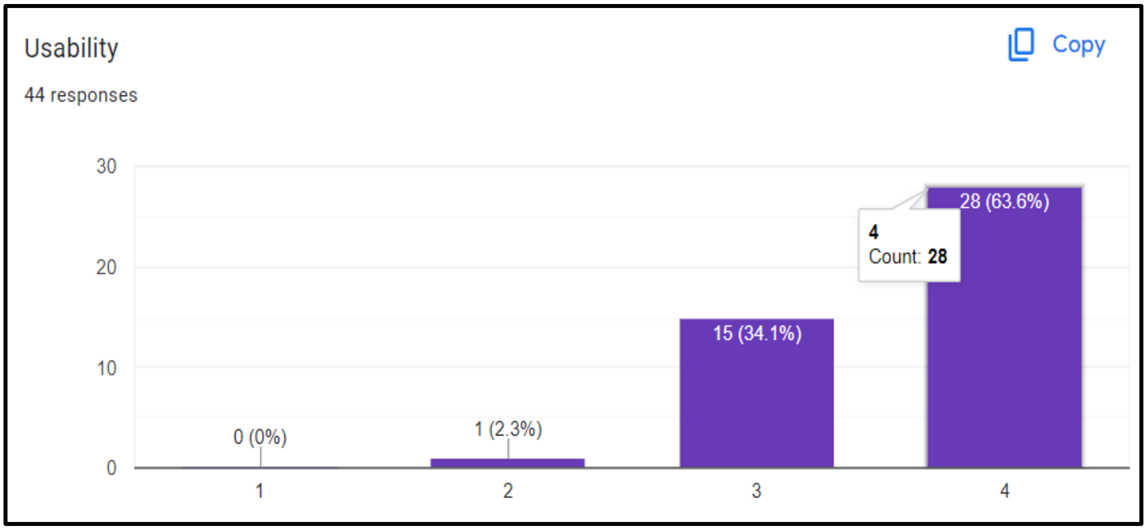
44 responses

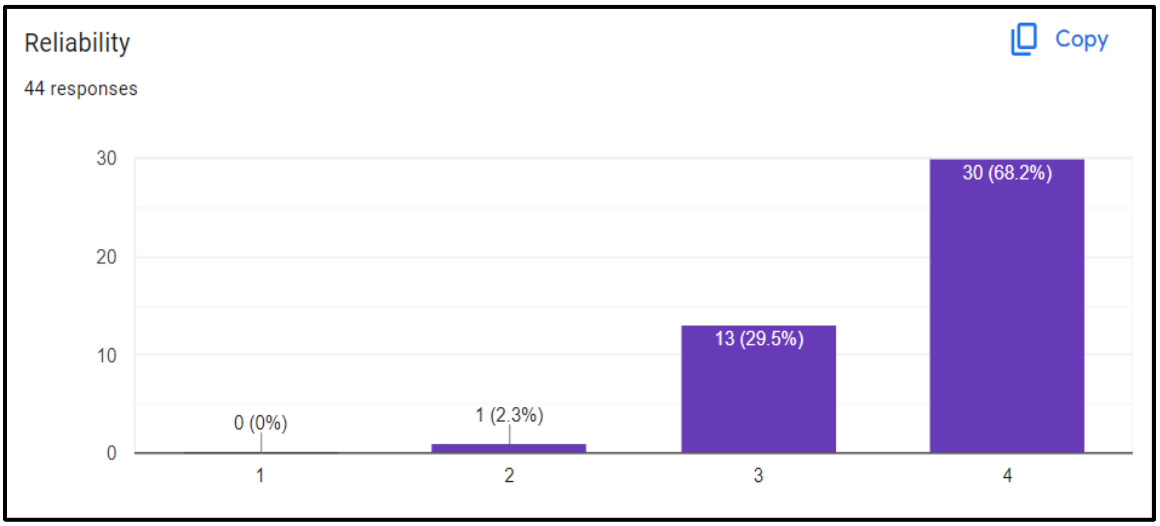


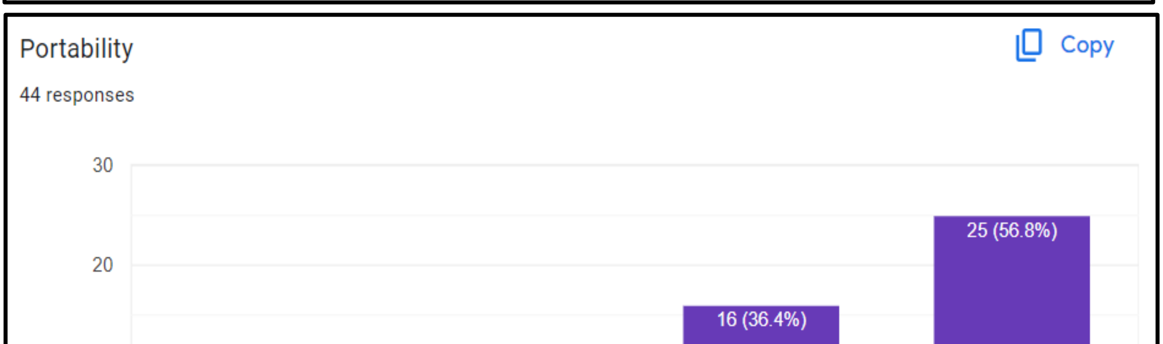

Hydroponics Based on ISO 25010 Software
Functional Suitability
Copy
44 responses
30
20
10
0
0 (0%)
0 (0%)
23 (52.3%)
21 (47.7%)
1
2
3
4

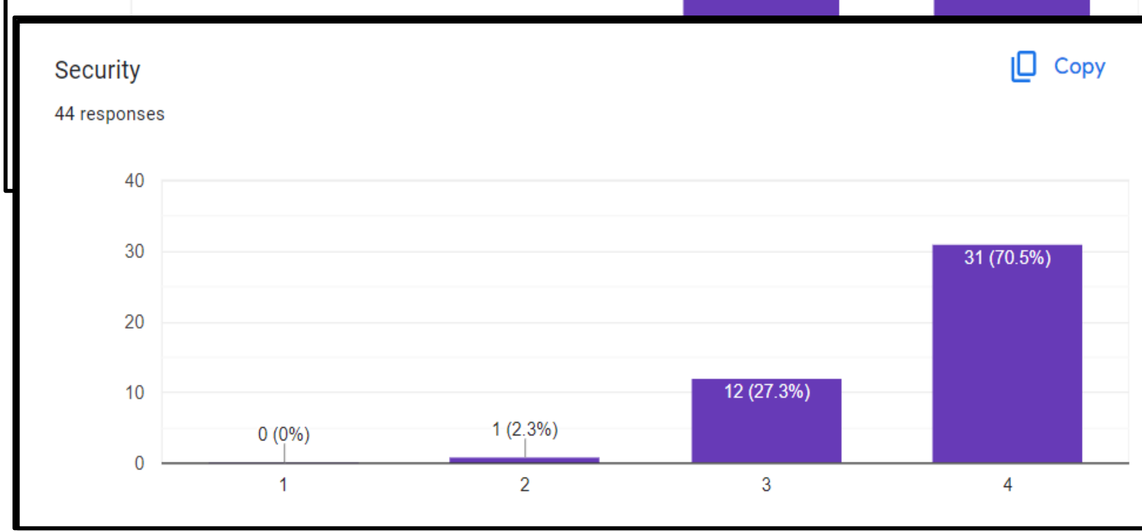

Performance Efficiency
Copy
44 responses
30
20
10
0
0 (0%)
1 (2.3%)
22 (50%)
3
Count: 22
21 (47.7%)
1
2
3
4


Compatibility
Copy
44 responses
30
20
10
0
1 (2.3%)
2 (4.5%)
14 (31.8%)
27 (61.4%)
1
2
3
4


Usability
Copy
44 responses
30
20
10
0
4
Count: 28
28 (63.6%)
15 (34.1%)
1 (2.3%)
0 (0%)
1
2
3
4


Reliability
Copy
44 responses
30
20
10
0
30 (68.2%)
13 (29.5%)
1 (2.3%)
0 (0%)
1
2
3
4


Portability
Copy
44 responses
30
20
25 (56.8%)
16 (36.4%)


Security
Copy
44 responses
40
30
20
10
0
31 (70.5%)
12 (27.3%)
1 (2.3%)
0 (0%)
1
2
3
4

INTRODUCTION: The students involved in the completion of this study needs to conduct an evaluation questionnaire regarding to the different factors and parameters, entitled "Pi-Droponics: An Automated Hydroponics System with Notification System using Convolutional Neural Network." The questionnaires are based on different standard based on ISO 25010 to determine the quality of the software and technical using a 4-point Likert scale.
INSTRUCTION: Please rate each of the parameters and check (✓) one response if your review suffice to this rating. Check (✓) 4 – **Very Functional**, 3 – **Functional**, 2 – **Slightly Functional** and 1 – **Not Functional**

FUNCTIONAL SUITABILITY	RANGE	INTERPRETATION	CHECK
Efficiently meets all the stated and implied need; and no weakness are found.	4	Very Functional	
Satisfactorily meets all the stated and implied needs but acceptable /tolerable weakness are found which will however not affect its function.	3	Functional	✓
Meet only some of the stated and implied needs and minor weaknesses are found that will slightly affect its function.	2	Slightly Functional	
Meets very few stated and implied needs; and major weaknesses are found that will greatly affect its function.	1	Not Functional	

PERFORMANCE EFFICIENCY	RANGE	INTERPRETATION	CHECK
Efficiently performs all the stated and implied needs relative to the amount of resources without any flaws.	4	Very Functional	
Satisfactorily performs all the stated and implied needs relative to the amount resources used but tolerable flaws are found which will however not affect the MTA's over-all performance.	3	Functional	
Poorly performs the stated and implied needs relative to the amount of resources used and major flaws are found that will greatly affect performance.	2	Slightly Functional	✓
Does not perform the stated and implied needs relative to the amount of resources used. Thus, restricting is needed.	1	Not Functional	

COMPATIBILITY	RANGE	INTERPRETATION	CHECK
Efficiently meets the required function while sharing the same hardware and software environment. It can exchange information with other systems with no inadequacies at all.	4	Very Functional	
Satisfactorily meet the required function while sharing the same hardware and software environment. It can exchange information with other systems but tolerable inadequacies are found which will not in any way affect the MTA's over-all compatibility.	3	Functional	✓
Poorly performs the required function while sharing the same hardware and software environments. It can exchange information with other systems but major inadequacies are found that will greatly affect its compatibility.	2	Slightly Functional	
Does not meet the required function while sharing the same hardware and software environment. It cannot exchange information with other systems thus restructuring is needed.	1	Not Functional	

USABILITY	RANGE	INTERPRETATION	CHECK
Provides a user interface that is easy to operate and control such that the users can effortlessly perform their appropriate needs even without guide or supervision.	4	Very Functional	
Provides a user interface that is easy to operate and control such that the users can easily perform their appropriate needs with minimal guide or supervision.	3	Functional	✓
Provides a user interface that is slightly difficult to operate and control such that the users can easily perform their appropriate needs.	2	Slightly Functional	
Provides a user interface that is difficult to operate and control such that it is difficult for the users to perform their appropriate needs at all. Thus, restructuring is required for the MTA.	1	Not Functional	

RELIABILITY	RANGE	INTERPRETATION	CHECK
Efficiently performs its functions without failure under specified conditions and period of time with no flaws at all.	4	Very Functional	
Satisfactory performs its functions without failure under specified conditions and period of time with minimal flaws which will not in any way affect the MTA's overall reliability.	3	Functional	✓
Performs only some of its functions without failure under specified conditions and period of time with major flaws which will affect the MTA's overall reliability.	2	Slightly Functional	
Cannot perform any of its functions without failure under specified conditions and period of time, thus restructuring is required.	1	Not Functional	

SECURITY	RANGE	INTERPRETATION	CHECK
Efficiently protects information and data with no weaknesses at all.	4	Very Functional	
Satisfactory protects information and data with minimal weaknesses which will not in any way affect the MTA's overall security.	3	Functional	✓
Protects only some of its information and data with major weaknesses which will affect the MTA's overall security.	2	Slightly Functional	
Cannot protect any information and data, thus restructuring is required.	1	Not Functional	

MAINTAINABILITY	RANGE	INTERPRETATION	CHECK
Efficiently retains its original form and can be restored to that form in case of failure, with no weaknesses at all.	4	Very Functional	
Satisfactory retains its original form and be restored to that form in case of failure; with minimal weaknesses, which however, will not affect its overall maintainability.	3	Functional	
Retains and restores only some of its original features in case of failure; with moderate weaknesses which may affect its overall maintainability.	2	Slightly Functional	✓
Cannot retain and restore its original form in case of failure, thus restructuring is required.	1	Not Functional	

PORTABILITY	RANGE	INTERPRETATION	CHECK
Efficiently adapts to changes in environment and can be installed / uninstalled in specified environment, with no weaknesses at all.	4	Very Functional	✓
Satisfactory adapts to changes in environment and can be installed / uninstalled in specified environment with minimal weaknesses, which however, will not affect its overall portability.	3	Functional	
Barely adapts to changes in environment and can be installed / uninstalled in specified environment with moderate weaknesses which may affect its overall portability.	2	Slightly Functional	
Cannot adapt to changes in environment and cannot be installed / uninstalled in specified environment.	1	Not Functional	

ANNEX III

Pi-Droponics - Codes

a. Arduino Codes

```
//Define global static variable here
 //DEFINE RELAY PINS HERE
 #define RELAY_PIN_2 2
 #define RELAY_PIN_3 3
 #define RELAY_PIN_4 4
 #define RELAY_PIN_5 5
 #define RELAY_PIN_6 6
 #define RELAY_PIN_7 7
 #define RELAY_PIN_8 8

 //DEFINE ANALOG AND DIGITAL PINS HERE
 #define TDS_PIN A1
 #define PH_PIN A0
 #define WATERLEVEL_PIN A3
 #define DHTPIN 11

 //DEFINE DATA VARIABLES HERE
 #define Offset 6.19
 #define samplingInterval 20
 #define printInterval 800
 #define ArrayLength  40

 //DEFINE OTHER DATA HERE
```

```
  #define DHTTYPE   DHT22

// Include Libraries here
  #include "DFRobot_EC.h"
  #include <EEPROM.h>
  #include <DHT.h>
  #include <DHT_U.h>
  #include <Wire.h>
  #include <LiquidCrystal_I2C.h>
  #include <BH1750.h>
  #include <ArduinoJson.h>
  #include <SoftwareSerial.h>

// initialize object creation here
  DFRobot_EC ec;
  DHT dht(DHTPIN, DHTTYPE);
  BH1750 GY30;
  SoftwareSerial megaToNode(14,15); //PIN 17->D5, PIN 18 -> D6
  DynamicJsonDocument data(192);

// Add Global Variables per sensor here
  // pH Sensor
  float voltageForPh;
```

```
  int pHArray[ArrayLength];
  int pHArrayIndex=0;
  unsigned long int avgValue;
  float b;
  int buf[10],temporary;

  // TDS Sensor
  float tdsValue = 0;

  // Water Sensor
  int WaterLevelSensor;

  // Light Sensor
  float lux;

  // Temperature and humidity
  float temp, humid;

//------------------------------------------------------------
// Function Declaration here
  // For pH Sensor
  double avergearray(int* arr, int number){
    int i;
```

```
int max,min;
double avg;
long amount=0;
if(number<=0){
  Serial.println("Error number for the array to avraging!/n");
  return 0;
}
if(number<5){   //less than 5, calculated directly statistics
  for(i=0;i<number;i++){
    amount+=arr[i];
  }
  avg = amount/number;
  return avg;
}else{
  if(arr[0]<arr[1]){
    min = arr[0];max=arr[1];
  }
  else{
    min=arr[1];max=arr[0];
  }
  for(i=2;i<number;i++){
    if(arr[i]<min){
      amount+=min;        //arr<min
```

```
        min=arr[i];
      }else {
        if(arr[i]>max){
          amount+=max;    //arr>max
          max=arr[i];
        }else{
          amount+=arr[i]; //min<=arr<=max
          }
        }//if
      }//for
      avg = (double)amount/(number-2);
    }//if
  return avg;
 }


//---------------------------------------------------------
void setup() {
 // put your setup code here, to run once:
 Serial.begin(115200); // Debugging Consoles
 megaToNode.begin(115200); //Sending data to nodemcu


 ec.begin();     // Init TDS Sensor
 dht.begin();    // Init dht Sensor
```

```
  Wire.begin();   // Initialize the I2C bus for use by the BH1750 library
  GY30.begin();    // Initialize the sensor object

  sensor_t sensor;

  // Init Relay pinMode
  pinMode(RELAY_PIN_2, OUTPUT);
  pinMode(RELAY_PIN_3, OUTPUT);
  pinMode(RELAY_PIN_4, OUTPUT);
  pinMode(RELAY_PIN_5, OUTPUT);
  pinMode(RELAY_PIN_6, OUTPUT);
  pinMode(RELAY_PIN_7, OUTPUT);
  pinMode(RELAY_PIN_8, OUTPUT);

  // Set Digital Write to each relay pin
  digitalWrite (RELAY_PIN_2, HIGH);
  digitalWrite (RELAY_PIN_3, HIGH);
  digitalWrite (RELAY_PIN_4, HIGH);
  digitalWrite (RELAY_PIN_5, HIGH);
  digitalWrite (RELAY_PIN_6, HIGH);
  digitalWrite (RELAY_PIN_7, HIGH);
  digitalWrite (RELAY_PIN_8, HIGH);
}
```

```
void loop() {

  // Code for pH sensor reading
  for(int i=0;i<10;i++)       //Get 10 sample value from the sensor for smooth the value
    {
      buf[i]=analogRead(PH_PIN);
      delay(10);
    }
  for(int i=0;i<9;i++)        //sort the analog from small to large
    {
      for(int j=i+1;j<10;j++)
      {
        if(buf[i]>buf[j])
        {
          temporary=buf[i];
          buf[i]=buf[j];
          buf[j]=temporary;
        }
      }
    }
  avgValue=0;
  for(int i=2;i<8;i++)                              //take the average value of 6 center sample
```

```
  avgValue+=buf[i];
float phValue=(float)avgValue*5.0/1024/6;         //convert the analog into millivolt
phValue=3.5 * phValue + Offset;                    //convert the millivolt into pH value
Serial.print("    pH:");
Serial.print(phValue,2);
Serial.println(" ");
if (phValue < 6) {
     digitalWrite(RELAY_PIN_2, LOW);
     delay(4000);
     digitalWrite(RELAY_PIN_2, HIGH);
}
if (phValue >= 7){
     digitalWrite(RELAY_PIN_4, LOW);
     delay(4000);
     digitalWrite(RELAY_PIN_4, HIGH);
}
delay(1000);

// Code for tds sensor
tdsValue = (analogRead(TDS_PIN)/1024.0*5000) + 1950/2;   // read the voltage
Serial.print(" TDS:");
Serial.print(tdsValue);
Serial.println("ppm");
```

```
if (tdsValue < 1260) {
  digitalWrite(RELAY_PIN_6, LOW);
  delay(32000);   //<---- comment out for debugging
  digitalWrite(RELAY_PIN_6, HIGH);
}
else if (tdsValue >= 1610){
      digitalWrite(RELAY_PIN_3, LOW);
      delay(32000); //<---- comment out for debugging
      digitalWrite(RELAY_PIN_3, HIGH);
}
delay(1000);


// Code for water level sensor
WaterLevelSensor = analogRead(WATERLEVEL_PIN);
Serial.print("water:");
Serial.println(WaterLevelSensor);
 while(WaterLevelSensor<10){
   digitalWrite(RELAY_PIN_5, LOW);
   delay(30000);
 }
 digitalWrite(RELAY_PIN_5, HIGH);
delay(1000);
```

```
// Code for light sensor
lux = GY30.readLightLevel();
Serial.print("Light:");
Serial.print(lux);
Serial.println("lx");
delay(1000);


// Code for temperature sensor
//dht22 temperature
temp = dht.readTemperature();
humid = dht.readHumidity();
Serial.print(F("Temperature: "));
Serial.print(temp);


Serial.println(F("°C"));
Serial.print(F("Humidity: "));
Serial.print(humid);


Serial.println(F("%"));
if ( temp >= 21){
  digitalWrite(RELAY_PIN_7, LOW);
  delay(180000);  //<---- comment out for debugging
  digitalWrite(RELAY_PIN_7, HIGH);
```

```
  }
  delay(1000);

  // Stringify values into JSON format then send to the NodeMcu using Serial
Communication
  data["tds"] = tdsValue;
  data["waterlevel"] = WaterLevelSensor;
  data["ph"] = phValue;
  data["light"] = lux;
  data["temp"] = temp;
  data["humid"] = humid;

  megaToNode.print(serializeJson(data, megaToNode));
  Serial.print(serializeJson(data, Serial));



  delay(3600000);  //Adjust the value of this delay
}
```

b. NodeMCU Codes

```
#include <SoftwareSerial.h>
#include <Firebase_ESP_Client.h>
#include <Arduino.h>
```

```
#include <ArduinoJson.h>
#include <ESP8266WiFi.h>
#include "addons/TokenHelper.h"
#include "addons/RTDBHelper.h"

#define API_KEY "AIzaSyACeuQeQQf98WW8XjwTOyIYNUE8qPF8oTA"
//Change API key
#define DATABASE_URL "https://hypiv-database-default-rtdb.firebaseio.com/"
//Change firebase url
SoftwareSerial megaToNode(D6,D5); // RX TX
//Define Firebase Data object
FirebaseData fbdo;
FirebaseAuth auth;
FirebaseConfig config;

const char* ssid     = "HUAWEI-8755";         // The SSID (name) of the Wi-Fi
network you want to connect to
const char* password = "41265499";     // The password of the Wi-Fi network
int tds; // 0
int waterlevel; // 0
float ph; // 8.405518
float light ; // 219.1667
float temp; // 25.8
```

```
float humid; // 58.1
unsigned long sendDataPrevMillis = 0;
int count = 0;
bool signupOK = false;

void setup() {
  // put your setup code here, to run once:
  megaToNode.begin(115200); //Serial Communication Receiver
  Serial.begin(115200); //Debug Console
  delay(10);
  Serial.println('\n');
  //Connect to internet
  WiFi.begin(ssid, password);             // Connect to the network
  Serial.print("Connecting to ");
  Serial.print(ssid); Serial.println(" ...");

  int i = 0;
  while (WiFi.status() != WL_CONNECTED) { // Wait for the Wi-Fi to connect
    delay(1000);
    Serial.print(++i); Serial.print(' ');
  }

  Serial.println('\n');
```

```
  Serial.println("Connection established!");
  Serial.print("IP address:\t");
  Serial.println(WiFi.localIP());         // Send the IP address of the ESP8266 to the
computer
  //Setup firebase
  config.api_key = API_KEY;
  config.database_url = DATABASE_URL;
  if (Firebase.signUp(&config, &auth, "", "")){
    Serial.println("ok");
    signupOK = true;
  }
  else{
    Serial.printf("%s\n", config.signer.signupError.message.c_str());
  }

  /* Assign the callback function for the long running token generation task */
  config.token_status_callback = tokenStatusCallback; //see
addons/TokenHelper.h

  Firebase.begin(&config, &auth);
  Firebase.reconnectWiFi(true);
}
```

```
void loop() {
  //Receive data from arduino using serial communication
  if (megaToNode.available()>0) {
    StaticJsonDocument<192> doc;
    Serial.setTimeout(5000);
    DeserializationError error = deserializeJson(doc, megaToNode.readString());
    if (error) {
      Serial.print(F("deserializeJson() failed: "));
      Serial.println(error.f_str());
      return;
    }

    tds = doc["tds"]; // 0
    waterlevel = doc["waterlevel"]; // 0
    ph = doc["ph"]; // 8.405518
    light = doc["light"]; // 219.1667
    temp = doc["temp"]; // 25.8
    humid = doc["humid"]; // 58.1

    //add firebase code below
  if (Firebase.ready() && signupOK && (millis() - sendDataPrevMillis > 15000 ||
sendDataPrevMillis == 0)){
    sendDataPrevMillis = millis();
```

```
 // Send TDS value to the database
 if (Firebase.RTDB.setFloat(&fbdo, "data/tds", tds)){
   Serial.println("PASSED");
   Serial.println("PATH: " + fbdo.dataPath());
   Serial.println("TYPE: " + fbdo.dataType());
 }
 else {
   Serial.println("FAILED");
   Serial.println("REASON: " + fbdo.errorReason());
 }
// Send water level value to the database
 if (Firebase.RTDB.setFloat(&fbdo, "data/waterLevel", waterlevel)){
   Serial.println("PASSED");
   Serial.println("PATH: " + fbdo.dataPath());
   Serial.println("TYPE: " + fbdo.dataType());
 }
 else {
   Serial.println("FAILED");
   Serial.println("REASON: " + fbdo.errorReason());
 }

 // Send ph Level value to the database
 if (Firebase.RTDB.setFloat(&fbdo, "data/pHLevel", ph)){
```

```
  Serial.println("PASSED");
  Serial.println("PATH: " + fbdo.dataPath());
  Serial.println("TYPE: " + fbdo.dataType());
}
else {
  Serial.println("FAILED");
  Serial.println("REASON: " + fbdo.errorReason());
}

// Send light Level value to the database
if (Firebase.RTDB.setFloat(&fbdo, "data/light", light)){
  Serial.println("PASSED");
  Serial.println("PATH: " + fbdo.dataPath());
  Serial.println("TYPE: " + fbdo.dataType());
}
else {
  Serial.println("FAILED");
  Serial.println("REASON: " + fbdo.errorReason());
}

// Send temperature Level value to the database
if (Firebase.RTDB.setFloat(&fbdo, "data/temperature", temp)){
  Serial.println("PASSED");
```

```
  Serial.println("PATH: " + fbdo.dataPath());
  Serial.println("TYPE: " + fbdo.dataType());
}
else {
  Serial.println("FAILED");
  Serial.println("REASON: " + fbdo.errorReason());
}

// Send humidity value to the database
if (Firebase.RTDB.setFloat(&fbdo, "data/humidity", humid)){
  Serial.println("PASSED");
  Serial.println("PATH: " + fbdo.dataPath());
  Serial.println("TYPE: " + fbdo.dataType());
}
else {
  Serial.println("FAILED");
  Serial.println("REASON: " + fbdo.errorReason());
}
//Serial Output
Serial.print("TDS Value: ");
Serial.println(tds);
Serial.print("Waterlevel Value: ");
Serial.println(waterlevel);
```

```
        Serial.print("pH Value: ");
        Serial.println(ph);
        Serial.print("Light Value: ");
        Serial.println(light);
        Serial.print("Temp Value: ");
        Serial.println(temp);
        Serial.print("Humidity Value: ");
        Serial.println(humid);
        Serial.print("\n\n\n");
      }
      }
      }
```

c. Raspberry Pi Codes

```
#! /usr/bin/python3

from flask import Flask, flash, request, redirect, url_for, Response, render_template
import cv2
import numpy as np
from keras.models import load_model
import firebase_admin
from firebase_admin import credentials
from firebase_admin import db

app = Flask(__name__)
model = load_model("/home/pi/rpi_codes/static/DiseaseDetector2.hdf5")
```

```
classes = ['Alternaria', 'Anthracnose', 'Downy Mildew', 'Healthy', 'Non-plants', 'White
Rust']

cred = credentials.Certificate("Credentials.json")
firebase_admin.initialize_app(cred,{
    'databaseURL': 'https://database-9eefd-default-rtdb.firebaseio.com/'
})

ref = db.reference('Predictions/')

def predictLive(imageSource,cameraPort):
    img_rgb = cv2.cvtColor(imageSource, cv2.COLOR_BGR2RGB)
    img_res = cv2.resize(img_rgb, (224,224), interpolation = cv2.INTER_AREA)
    x = np.expand_dims(img_res , 0)
    pred = model.predict_on_batch(x).flatten()
    if(cameraPort == 0):
        pred_ref = ref.child('camera1')
        pred_ref.update({ 'Class' : classes[int(np.argmax(pred))],
        'PredMax' : int(np.argmax(pred))
        })
    elif(cameraPort == 2):
        pred_ref = ref.child('camera2')
        pred_ref.update({
            'Class' : classes[int(np.argmax(pred))],
            'PredMax' : int(np.argmax(pred))
        })
    elif(cameraPort == 4):
        pred_ref = ref.child('camera3')
        pred_ref.update({
            'Class' : classes[int(np.argmax(pred))],
```

```
            'PredMax' : int(np.argmax(pred))
        })
    elif(cameraPort == 6):
        pred_ref = ref.child('camera4')
        pred_ref.update({
            'Class' : classes[int(np.argmax(pred))],
            'PredMax' : int(np.argmax(pred))
        })
    return np.argmax(pred)

def genCam1():
    cap = cv2.VideoCapture(0)
    cap.set(cv2.CAP_PROP_FOURCC, cv2.VideoWriter_fourcc('M', 'J', 'P', 'G'))
    while True:
        success, frame = cap.read()  # read the camera frame
        if not success:
            break
        else:
            ret, buffer = cv2.imencode('.jpg', frame)
            print("Camera 1 Prediciton: " + classes[int(predictLive(frame,0))])
            frame = buffer.tobytes()
            yield (b'--frame\r\n'
                   b'Content-Type: image/jpeg\r\n\r\n' + frame + b'\r\n')  # concat frame one by
one and show result

def genCam2():
    capB = cv2.VideoCapture(2)
    capB.set(cv2.CAP_PROP_FOURCC, cv2.VideoWriter_fourcc('M', 'J', 'P', 'G'))
    while True:
        _, frameB = capB.read()
        frameB = cv2.flip(frameB, 1)
```

```
        retB, jpegB = cv2.imencode('.jpg', frameB)
        print("Camera 2 Prediciton: " + classes[int(predictLive(frameB,2))])
        if retB:
            yield (b'--frame\r\n'
                   b'Content-Type: image/jpeg\r\n\r\n' + jpegB.tobytes() + b'\r\n\r\n')
def genCam3():
    capC = cv2.VideoCapture(4)
    capC.set(cv2.CAP_PROP_FOURCC, cv2.VideoWriter_fourcc('M', 'J', 'P', 'G'))
    while True:
        _, frameC = capC.read()
        frameC = cv2.flip(frameC, 1)
        retC, jpegC = cv2.imencode('.jpg', frameC)
        print("Camera 3 Prediciton: " + classes[int(predictLive(frameC,4))])
        if retC:
            yield (b'--frame\r\n'
                   b'Content-Type: image/jpeg\r\n\r\n' + jpegC.tobytes() + b'\r\n\r\n')

def genCam4():
    capD = cv2.VideoCapture(6)
    capD.set(cv2.CAP_PROP_FOURCC, cv2.VideoWriter_fourcc('M', 'J', 'P', 'G'))
    while True:
        _, frameD = capD.read()
        frameD = cv2.flip(frameD, 1)
        retD, jpegD = cv2.imencode('.jpg', frameD)
        print("Camera 4 Prediciton: " + classes[int(predictLive(frameD,6))])
        if retD:
            yield (b'--frame\r\n'
                   b'Content-Type: image/jpeg\r\n\r\n' + jpegD.tobytes() + b'\r\n\r\n')

@app.route('/')
def index():
```

```
    return render_template('index.html')

#Routes for multi camera rendering


@app.route('/video_feed0')
def video_feed0():
    return Response(genCam1(),
                    mimetype='multipart/x-mixed-replace; boundary=frame')

@app.route('/video_feed1')
def video_feed1():
    return Response(genCam2(),
                    mimetype='multipart/x-mixed-replace; boundary=frame')

@app.route('/video_feed2')
def video_feed2():
    return Response(genCam3(),
                    mimetype='multipart/x-mixed-replace; boundary=frame')

@app.route('/video_feed3')
def video_feed3():
    return Response(genCam4(),
                    mimetype='multipart/x-mixed-replace; boundary=frame')

if __name__ == '__main__':
    app.run(host='192.168.8.104', port='5000')
```

ANNEX IV

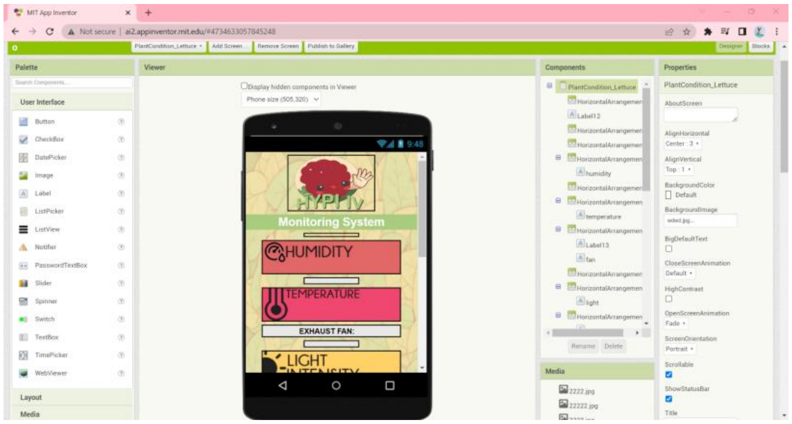
Project Documentation

Progress Documentation

Barangay Visit

Construction of Hydroponic System and Greenhouse

Panel's Visit

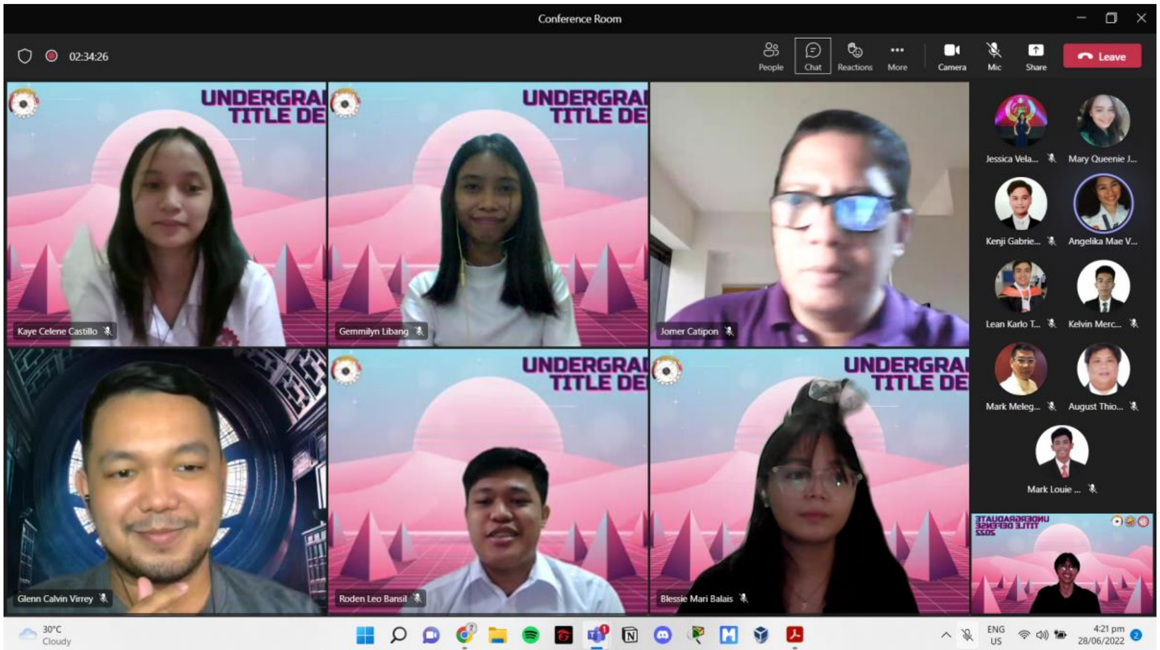


Creation of Mobile Application

Testing of the Model

The Hydroponics Model with Spinach Plant

Photos from Previous Defense



Title Defense

Topic Defense

Progress Defense

Pre-final Defense

Appreciate 2023

Final Defense and Pitching

ANNEX VI

User's Manual

HYPI IV

USERS MANUAL

Pi-Droponics: An Automated Hydroponics System with Notification System using Convolutional Neural Network

ABOUT US

We are HyPi IV, and we proposed the Pi-Droponics: An Automated Hydroponics System with Notification System using Convolutional Neural Network, develop an automated vertical hydroponics system with monitoring and control that will include a real-time disease detection system for leafy plants and vegetables and a notification system that will send information whenever the system is out of specification parameters, as well as the plants that have detected disease and if it is ready to harvest.

HyPi IV

Barangay 786 Zone 86,

Sta. Ana Manila

pdroponics@gmail.com

0915-503-6742

SETTING UP THE SYSTEM

1. Set up the right amount of water for the reservoir needed for the Pi-Droponics.
2. Put the nutrients and water under the designated Peristaltic Buffer Pump.
3. Put the silicon tube into the Peristaltic Buffer Pump and place it in the reservoir. Make sure to know where the *Suction* and *Discharge* sides of the Peristaltic Buffer Pump are.
4. Put the garden pots with seedlings in the holes of each 3 inches diameter PVC pipe.
5. Plug the power supply of Raspberry Pi IV Model B, Arduino Mega 2560 R3, NodeMCU, Actuators, water motor, and LED grow lights into the extension wire.
6. Turn on the pocket Wi-Fi to send the data from the sensors to the firebase. Make sure that the pocket Wi-Fi is charged and turned on.
7. Plug in the main power supply for it to operate the system simultaneously.
8. Set up the right amount of luminosity for the Spinach and time duration for the LED grow lights to turn on within 24 hours.

HOW TO USE THE HYPI IV APPLICATION

1. Scan the QR code to download the HyPi IV Application.

Note: the mobile application can only be downloaded through all version of android devices

2. Install the application to your mobile phone
3. Select the plant to check the system condition, plant condition, and plant live monitoring.
4. The user can set a date for the new batch of the plant to know the potential harvest date.

REMINDERS

1. Always check the water from the reservoir to see if some developed mosses are around it.
2. Check the system at least three (3) to four (4) a week if the components are complete because the greenhouse is made of wood and UV plastic and is very vulnerable to thieves.
3. Monitor the characteristic of plant growth once a week.
4. Charge the pocket Wi-Fi once a day.

ANNEX VII

Manual for duplication of Prototype

HYPI IV

MANUAL FOR DUPLICATION OF PROTOTYPE

Pi-Droponics: An Automated Hydroponics System with Notification System using Convolutional Neural Network

HARDWARE

Hydroponics System:

a. Materials

Raspberry Pi IV Model B	Screw and nails
Arduino Mega 2560	Solvent Cement
DFRobot Gravity: Analog pH Sensor	Stick well
DFRobot Analog TDS Sensor/Meter	Vulcaseal
DHT22 Temperature and Humidity Sensor	Metal clamp
GY-30 BH1750FVI Light Intensity Sensor	LED Grow Lights
Water Level Sensor	NodeMCU V3 ESP8266 ESP-12E
8-Channel 5V Relay Module	Water pump DC motor
Peristaltic Buffer Pump	AC/DC Switching Power Supply
12V DC fans	SD card
USB camera	Silicon tubes
AC/DC Power Supply	Junction Box
PVC [1/2 elbow, 1/2x1/2 Tee, 3x3 Clean Out, Coupling (1/2 Inch diameter), Pipe (1/2 Inch diameter), and Pipe (3 inches diameter)]	Jumping Wires
Wood (1x2 wood and 4x8 plywood)	Copper Wire

b. Procedures

1.) Collect all the specified materials mentioned above that are necessary for building a hydroponic system.

2.) Cut the 1x2 wood in the following sizes:

- 4 pcs 381 cm
- 6 pcs 244 cm

- 6 pcs 122 cm

3.) Create a 3-piece wooden frame using the 244 cm 1x2 wood and 122 cm 1x2 wood, it will serve as the support of the plywood.

4.) The combined wooden frame and plywood will act as the shelves and top.

5.) Secure the four pieces of 381 cm 1x2 wood to the four corners of the wooden shelves and top, creating a structural support for the entire system.

6.) Construct a compact shelf underneath the initial layer of the system, designed to accommodate the junction box and serve as a convenient space for positioning the peristaltic pump and storing the bottled nutrient solutions.

7.) Drill 5 holes for each 3inch diameter horizontal pipe to accommodate net pots. Ensure that the holes are evenly spaced and properly aligned.

8.) Attach the 3x3 clean out to the ends of the horizontal pipes and use solvent cement and vulcaseal around the clean out to prevent water leakage.

9.) Attach the 3x3 pipes to the shelves of the hydroponics system, each must consist of 2 pipes.

10.) Build a water distribution system using a combination of a 1/2 PVC elbow, tee, and pipe, to achieve uniform water distribution across every 3-inch diameter PVC pipe.

11.) Cut an appropriate length of 1x2 PVC pipe that will serve as the connection between the water pump and the assembled water distribution system.

12.) Apply a coat of white paint to the entire system, not only to enhance its visual appeal but also to reduce the heat buildup within the system.

13.) Arrange the water reservoir by installing the water pump and a holder for three sensors, including a water level sensor, a TDS sensor, and a pH sensor.

14.) Mount the LED grow lights onto the system, ensuring that their placement allows for uniform distribution of light to each plant.

15.) Attach the USB cameras to the system, placing two cameras on each layer, and ensure that the field of view of each camera covers all the holes in the system.

16.) Position the Peristaltic Buffer Device in its designated location within the system, and insert the silicone tube into the device, ensuring it reaches the water reservoir area.

17.) Position the DHT22 Air and Humidity Sensor in close proximity to the plant placement area.

18.) The junction box will function as the chassis for the Arduino Mega 2560 Rev3, Raspberry Pi 4B, 8-Channel 5V Relay Module, NodeMCU V3 ESP8266 ESP-12E, and sensor connection platform.

19.) Establish the connections of the following:

- Sensors ---> Arduino Mega 2560 Rev3
- Arduino Mega 2560 Rev3 ---> 8-Channel 5V Relay Module
- 8-Channel 5V Relay Module ---> Actuators (Peristaltic Buffer Device, and DC Fan)
- NodeMCU V3 ESP8266 ESP-12E ---> Arduino Mega 2560 Rev3
- USB Camera ---> Raspberry Pi 4B

20.) Arrange the AC/DC power supply in both series and parallel connections to effectively power each actuator within the system.

21.) Position the bottles of nutrient solution on the compact shelf located beneath the initial layer of the system.

Greenhouse:

a. Materials

2x2 Wood	Screws
UV Plastic	Gun Tacker Staple
UV Net	PVC Net
Insulator Foam	Sintra Board
Nails	Glue Stick

b. Procedures:

1.) Collect all the necessary equipment required for the construction of a greenhouse.

2.) Utilize the 2x2 wood to construct a sizable cabinet frame with two doors. Ensure that the cabinet includes a designated area where the hydroponic system can be positioned.

3.) Construct a spacious cabinet frame with two doors using the 2x2 wood. Make sure that the cabinet provides ample room for accommodating the hydroponics system.

4.) Cover the wooden cabinet frame with UV plastic, ensuring that the material fully wraps around the frame. Pay special attention to ensuring that the doors of the cabinet can be securely closed even with the plastic covering in place.

5.) Create an opening on one side of the greenhouse and install a UV net in that space. This will facilitate proper ventilation within the greenhouse, allowing for improved airflow and temperature regulation.

6.) Apply insulator foam to cover the top and back part of the interior of the greenhouse. This will contribute to creating a cooler environment inside the greenhouse by providing insulation and reducing heat transfer.

7.) To facilitate optimal ventilation and air circulation in the greenhouse, install four DC fans on the side opposite to where the UV net is placed. These fans will greatly contribute to maintaining a healthy and well-regulated environment within the system. It is essential to ensure that each DC fan is securely and properly connected to the 8-Channel 5V Relay Module.

8.) Create individual covers for each DC fan using Sintra board and PVC net. The Sintra board will provide a sturdy base for the cover, while the PVC net will allow for proper airflow while preventing debris or pests from entering the fan.

SOFTWARE

Machine Learning:

a. Procedures:

1.) Acquire a basic understanding of how to use and program on the Raspberry Pi IV Model B.

2.) Download and install the following software to setup the Raspberry Pi:

- Raspberry Pi Imager - for installation of OS on your SD card.
- PuTTY - to SSH your Raspberry Pi remotely.
- VNC Viewer - for the GUI of your Raspberry Pi

- Pi-Tunnel - for remote access to your Raspberry Pi.

3.) After downloading and installing the aforementioned software, insert the SD card to your laptop to install the recommended OS using the Raspberry Pi Imager.

4.) Then after that, open your Raspberry Pi using the PuTTy. Enter the username and password of your Raspberry Pi. The default username is "pi" and the default password are "raspberry pi". You can also open the GUI of your Raspberry Pi with the using of VNC Viewer.

5.) After setting up your Raspberry Pi, start creating a folder inside the Raspberry Pi in order to store all your files that are needed for the web application.

6.) Insert your code in that folder. Then run it using the python IDE or other software that may be used inside the Raspberry Pi to test if the app is running. You can also create your code inside the Raspberry Pi.

7.) Establish a connection between the data for the disease detection model, from Firebase to MIT App Inventor, for it to display the data in the application.

8.) Once you have finished, run the code that is inside of the Raspberry Pi. Then go to your mobile application and click "Live Monitoring", in which it will show the 4 cameras that detect the diseases of the plants. Then it will also show which camera captures a healthy and/or a disease plant.

Mobile Application

a. Procedures:

1.) Acquire a basic understanding of how to use and program with blocks on the MIT App Inventor website.

2.) Sign into your Google account to begin creating the application.

3.) Create a new project and give it a name of your choice.

4.) Design the user interface for each screen using the Designer mode. Add necessary elements such as buttons, labels, text boxes, images, time pickers, etc. The required screens for this application include the home screen, estimated date of harvest screen, system condition screen, and plant disease screen.

5.) Make the elements functional by switching to Blocks mode and adding control, logic, and text blocks to them.

6.) Establish a connection between the data for the system condition and the disease detection model, from Firebase to MIT App Inventor, to display the data in the application.

7.) Utilize control blocks and the notification extension to set up notifications whenever the system falls outside the threshold range or when a plant is affected by a disease.

8.) Once you have finished, go to the "Build" section, and start generating the APK (Android Package).

9.) Install the APK on your Android mobile device.

ANNEX VIII

Proponents' Information

Blessie Mari S. Balais

I am keen to work in a highly organized and contribute my expertise and knowledge in the software application. I am willing to acquire new knowledge and experiences for professional growth through collaborations with other professionals.

0915 503 6742

blessiebalais@gmail.com

No.8 9D Area 7 UPS 5
Sucat Parañaque City

Blessie Balais

CERTIFICATES

Master IP Addressing and Subnetting for CCNA - MNET IT
(May 2022)

Certified Associate - Fortinet Network Security
Expert Level 3 (May 2023)

Certified Associate - Fortinet Network Security Expert
Level 2 (May 2023)

Certified Associate - Fortinet Network Security Expert
Level 1 (April 2023)

Certified in Cybersecurity (CC) Self Paced Training - (ISC)^2
(June 2023)

SKILLS

- Proficiency in Writing and speaking in both English and Tagalog
- Proficiency in using Microsoft Office Tools
- Proficiency in using Cisco Packet Tracer
- Proficiency in IP Addressing and Subnetting
- Basic Octave/MATLAB Language Program
- Basic NI Multisim Program

WORK EXPERIENCE

August 2022 - Setember 2022
Commsec Inc.

- Site Engineer Trainee
  -Visit the sites of the different projects
  -Monitoring the Project
  -Office works

EDUCATION

Tertiary | 2019 - Present
Bachelor of Science in Electronics Engineering
Technological University of the Philippines - Manila
Ayala Blvd, Ermita St. Manila

Secondary | 2017 - 2019
Science, Technology, Engineering and Mathematics (STEM)
Rizal Technological University - Boni Campus
Boni Ave, Mandaluyong City

Roden Leo Q. Bansil

I am an upcoming Electronics Engineering graduate with background knowledge in IP Addressing, Subnetting, and Network Simulation Tools. I am enthusiastic about expanding my knowledge and skills to further enhance my performance in a professional work environment.

0927 257 8513

roden.bansil@gmail.com

B15 L17 Yellow St. Casimiro Townhomes Phase 2, Pulang Lupa Uno, Las Piñas City.

Roden Leo Bansil

CERTIFICATES

Master IP Addressing and Subnetting for CCNA - MNET IT
(May 2022)

Certified Associate - Fortinet Network Security
Expert Level 3 (May 2023)

Certified Associate - Fortinet Network Security Expert
Level 2 (May 2023)

Certified Associate - Fortinet Network Security Expert
Level 1 (April 2023)

Certified in Cybersecurity (CC) Self Paced Training - (ISC)^2
(June 2023)

SKILLS

- Basic Python Language Program
- Basic Octave/MATLAB Language Program
- Basic NI Multisim Program
- Basic Microsoft Office Techniques
- Proficiency in IP Addressing and Subnetting
- Flexible Worker
- Hard Working Good Communication Skills
- Easily to adapt

WORK EXPERIENCE

August 2022 - Setember 2022
GoSolar Philippines

- Site Engineer Trainee
- Sales Engineer Trainee

EDUCATION

Tertiary | 2019 - Present
Bachelor of Science in Electronics Engineering
Technological University of the Philippines - Manila
Ayala Blvd, Ermita St. Manila

Secondary | 2017 - 2019
Science, Technology, Engineering and Mathematics (STEM)
STI College Las Piñas
STI Academic Center, Alabang-Zapote Road corner V. Guinto Street, Pamplona Tres, Las Piñas City, 1740Metro Manila

Kaye Celene S. Castillo

I aim to fully utilize and apply my skills, enhancing the knowledge I've gained through my academic pursuits to excel in practical applications. By actively seeking opportunities to expand my expertise, I strive to explore diverse career paths and lay a solid groundwork for my professional growth.

0915 390 1915

kaylenecstll@gmail.com

3 Mapalad St. Tonsuya, Malabon

Kaye Celene S. Castillo

CERTIFICATES

Master IP Addressing and Subnetting for CCNA - MNET IT
(May 2022)

Certified Associate - Fortinet Network Security
Expert Level 3 (May 2023)

Certified Associate - Fortinet Network Security Expert
Level 2 (May 2023)

Certified Associate - Fortinet Network Security Expert
Level 1 (April 2023)

Certified in Cybersecurity (CC) Self Paced Training - (ISC)^2
(June 2023)

SKILLS

- Critical Thinking
- Problem Solving
- Proficient in Octave, MATLAB, and Python Programming
- Proficient in NI Multisim
- Experienced with Cisco Packet Tracer
- Proficient in Microsoft Office
- Skilled in Canva

WORK EXPERIENCE

August 2022 - Setember 2022
Melham Construction Company

Research and Development

EDUCATION

Tertiary | 2019 - Present
Bachelor of Science in Electronics Engineering
Technological University of the Philippines - Manila
Ayala Blvd, Ermita St. Manila

- Dean's Lister
- DOST Undergraduate Scholar

Secondary | 2017 - 2019
Science, Technology, Engineering and Mathematics (STEM)
Caloocan CIty Science High School
10th Avenue, Caloocan City

Gemmilyn T. Libang

I am an Electronics Engineering graduate with background knowledge in IP Addressing, Subnetting, and Network Simulation Tools. I am enthusiastic about expanding my knowledge and skills to further enhance my performance in a professional work environment.

0995 024 5432

libanggemmilyn@gmail.com

#8 De Vera St., Brgy. Mairblo, Quezon City, 1104

Gemmilyn Libang

CERTIFICATES

Master IP Addressing and Subnetting for CCNA - MNET IT
(May 2022)

Certified Associate - Fortinet Network Security
Expert Level 3 (May 2023)

Certified Associate - Fortinet Network Security Expert
Level 2 (May 2023)

Certified Associate - Fortinet Network Security Expert
Level 1 (April 2023)

Certified in Cybersecurity (CC) Self Paced Training - (ISC)^2
(June 2023)

SKILLS

- Proficiency in utilizing Canva for video and powerpoint presentations.
- Proficient in Microsoft Office and Google tools
- Proficient in NI Multisim
- Basic skills in Python and R Programming Language
- Experienced with Cisco Packet Tracer
- Skilled in utilizing Raspberry Pi

WORK EXPERIENCE

August 2022 - Setember 2022
Melham Construction Company
Research and Development

EDUCATION

Tertiary | 2019 - Present
Bachelor of Science in Electronics Engineering
Technological University of the Philippines - Manila
Ayala Blvd, Ermita St. Manila

Secondary | 2017 - 2019
Science, Technology, Engineering and Mathematics (STEM)
Siena College of Quezon City
1105 Del Monte Ave, Quezon City

Wanting to acquire work experience in the field of electronics and communication, and to be able to utilize my current skills and knowledge in the said field, as well enhancing these skills and knowledge.

0967 609 3649

curtmaracha0401@gmail.com

2175 Road 5 Fabie Estate,
Sta. Ana, Paco, Manila

Curt Johann Maracha

CERTIFICATES

Master IP Addressing and Subnetting for CCNA - MNET IT
(May 2022)

Certified Associate - Fortinet Network Security
Expert Level 3 (May 2023)

Certified Associate - Fortinet Network Security Expert
Level 2 (May 2023)

Certified Associate - Fortinet Network Security Expert
Level 1 (April 2023)

Certified in Cybersecurity (CC) Self Paced Training - (ISC)^2
(June 2023)

Curt Johann M. Maracha

SKILLS

- Basic knowledge in IP addressing and subnetting and use of Cisco Packet Tracer
- Basic working knowledge with different circuit simulation tools like Proteus, TinkerCAD and NI Multisim
- Basic programming skills in Python, HTML and CSS
- Can work under pressure

WORK EXPERIENCE

August 2022 - Setember 2022
Commsec Inc.
Research and Development

EDUCATION

Tertiary | 2019 - Present
Bachelor of Science in Electronics Engineering
Technological University of the Philippines - Manila
Ayala Blvd, Ermita St. Manila

- Dean's Lister (2021 - 2022)

Secondary | 2017 - 2019
Science, Technology, Engineering and Mathematics (STEM)
Benedictine Institute of Learning
Subdivision, Abad Homes, Imus, Cavite