IN PARTNERSHIP WITH PLYMOUTH UNIVERSITY

Name: Jayawardena Kavinda	
Student Reference Number: 10952523	

Module Code: PUSL2022	Module Name: Introduction to IoT
Coursework Title: Final Report of the Plant Optimizing System	
Deadline Date: 20/04/2024	Member of staff responsible for coursework: Mr. Isuru Bandara
Programme: BSc. (Hons) in Software Engineering	

Please note that University Academic Regulations are available under Rules and Regulations on the University website www.plymouth.ac.uk/studenthandbook.

Group work: please list all names of all participants formally associated with this work and state whether the work was undertaken alone or as part of a team. Please note you may be required to identify individual responsibility for component parts.

10952523 - Jayawardena Kavinda – Solution / Workload Matrix
10952463 - Dulaj Hewage - Background / Appendix / Finalizing
10952470 - Unagollage Wijesinghe - Background / Appendix
10952545 - Witharamalage Sirimewan - Testing and Evaluation
10952565 - Pathiranage Didula Theekshana - Testing and Evaluation
10953075 - Duwage Perera - Testing and Evaluation
10952645 - Chamathka Abeykoon - Background
10953214 - Suwanda Akarsha - Conclusion
10952629 - Rathnayaka Rathnayaka - Methodology and approach
10952566 - Yaddehige Himasha - Methodology and approach

We confirm that we have read and understood the Plymouth University regulations relating to Assessment Offences and that we are aware of the possible penalties for any breach of these regulations. We confirm that this is the independent work of the group.

Signed on behalf of the group: Dinath

Individual assignment: ***I confirm that I have read and understood the Plymouth University regulations relating to Assessment Offences and that I am aware of the possible penalties for any breach of these regulations. I confirm that this is my own independent work.***
Signed :

Use of translation software: failure to declare that translation software or a similar writing aid has been used will be treated as an assessment offence.

I *have used/not used translation software.

If used, please state name of software……………………………………………………………

Overall mark _____% **Assessors Initials ______** **Date_________**

*Please delete as appropriate Sci/ps/d:/students/cwkfrontcover/2013/14

PUSL2020 – Introduction to IoT

Plant Optimizing System using IoT
Final Report

Group – H

Plymouth ID	Plymouth Name	NSBM ID	NSBM Name	Degree Program
10952523	Jayawardena Kavinda	30399	JWD Kavinda	SE
10952463	Dulaj Hewage	29981	DD Hewage	SE
10952470	Unagollage Wijesinghe	29676	DAD Wijesinghe	SE
10952545	Witharamalage Sirimewan	29590	WVW Sirimewan	SE
10952565	Pathiranage Didula Theekshana	29905	PDT Heenatigala	SE
10953075	Duwage Perera	31863	DTT Perera	SE
10952645	Chamathka Abeykoon	30202	CD Abeykoon	SE
10952629	Rathnayaka Rathnayaka	30224	MSI Rathnayaka	SE
10952566	Yaddehige Himasha	29780	YGH Sandeepani	SE
10953214	Suwanda Akarsha	24549	SHR Akarsha	SE

Table of Contents

Table of Figures

1. Introduction

This project refers to the development of a smart automatic plant watering and optimizing system based on Arduino technology. It marries soil-based irrigation with hydroponics to create the ideal plant growth environment. Sensors are used by the system to check soil moisture, pH level, humidity and NPK (Nitrogen, Phosphorus, Potassium) values. Wi-Fi module is used to send all the sensor data to a web-based dashboard for real-time monitoring and control. And, with Grow lights project achieves the goal of fast growing of the plant than default plant growth. This automatic irrigation system switches on a water pump if the soil is dry and stop when the soil is wet. This helps with water conservation and it's great for busy or away-from-home people.

2. Background

2.1. Project background study discussion

Automated and sensor-based systems using integrated technology offered in agriculture have made it easier to manage the availability of resources in a more efficient way while increasing plant productivity. An automatic plant watering system, however, works with a soil moisture sensor that can detect dryness and trigger irrigation when necessary. The advent of IoT technology has made these systems even better by enabling remote monitoring. This is a combination of hydroponics and automation is more accurate in controlling nutrients as well as environmental factors. The Plant Optimizing System works on the fundamentals of both soil-based and hydroponic, where a multitude of Arduino-based sensors used to check parameters for growth can automatically activate a water pump if soil moisture is reduced, optimal for users that are busy with work or have great work travelling.

2.2. Similar Solutions

Current available automatic irrigation systems from commercial providers cost a lot while delivering excessive features beyond basic garden requirements. Automatic watering systems are in use yet fail to take moisture levels in the ground into account so users waste water through overwatering. Ex :-

- Xioami MI plant flower care monitor
  - Having soil moisture temperature light and fertility levels monitored is what this smart device does. It interacts with a mobile app via Bluetooth to update on the real time plant health. It also does not have any automatic watering capabilities.
- Arduino – based Smart irrigation system
  - In many cases, Arduino is combined with soil moisture sensors and water pumps to automate the irrigation of soil moisture sensors. They may be effective, but not typically with nutrient monitoring and online data access.
- Hydroponic Systems (AeroGarden)
  - The kits that commercial hydroponic like AeroGarden offers are soil-less plant growth, along with automated lighting and start with the nutrient delivery. However, on hydroponics, they may not check real time soil parameter monitoring or web based interfaces.
- PlantLink
  - A protocol of soil moisture sensor and a mobile app which suggests when to water based on soil moisture sensor's result. Some versions can be integrated with smart watering systems but not nutrients or pH.
- Farmbot
  - An advanced, open source, CNC farming system that does seeding, watering & data collection. The garden apps use a web based platform in managing garden tasks. It is highly functional but expensive and complex for small scale users.

2.3. Problem Identification

Caring for plants traditionally involves lots of timely, manual work - you need to get up regularly to check soil conditions, to water, to ensure the plant receives adequate amounts of light. Write about your day, people who have busy lifestyles or travel a lot find it difficult to keep healthy plants. Unhealthy plants can be caused by problems with too much or not enough water, poor nutrient soil (pH, NPK imbalance) and low light. Current solutions are typically confined to simple moisture sensing or miss automation and remote availability.

Additionally, traditional systems require stable power sources and cannot be deployed too far from power lines. The solution is a fully automated and self-sustained plant optimization system that:

- We check soil moisture, pH, humidity and NPK
- Uses a water pump and solenoid valve to control watering
- Mixing Nutrients (Albert Solution) automatically via a Mixture when the pump is ON
- Provides grow lights for artificial lighting
- Data is transmitted to a web-based front end over Wi-Fi
- Runs on solar panel and battery with solar controller

This smart system also supports healthy plant growth with minimal human input, emphasizes energy efficiency, and encourages sustainable agriculture.

2.4. Problem Justification

With busy lives, most people are unaware and unable to cater to all the needs of plants such as watering, light conditions and soil health. These are the main factors that directly influence our plants' growth and productivity. For limited features, moisture sensing or timed irrigation, for example, there are some automated systems in the market, but these systems are usually used with a 24/7 electrical power supply and run on electricity rendering them unfit for off-grid or farmlands where electric supply is poor.

- Integrating :- Our project justifies the need of a more complete self-sustainable solution.
- Accurate monitoring with Soil moisture, pH, Humidity, and NPK sensor
- Pump and solenoid valve based automatic watering based on soil condition
- Grow light to maintain proper lighting where needed, top of the line indoors or in poor light conditions
- Solar power with battery backup for sustainable off-grid operation
- A web-based interface for live viewing and remote access over Wi-Fi

Not only does this system automate plant care, but it also promotes environmental sustainability, making it perfect for home users, urban gardens, and even remote agricultural locations.

3. Methodology and Approach

3.1. Methodology

The Agile Development Methodology was followed by the project, allowing flexible planning, incremental advancements, and collaboration effectively. Development split along hardware integration and software development.

Sensor and actuator setup by the hardware integration team by handling soil moisture sensor, pH sensor, NPK sensor, DHT22 - humidity + temperature, water pump, solenoid valve, grow lights and Powering all of this was a solar panel system with battery and solar controller for off-grid, energy-efficient operation.

The backend logic and a web-based dashboard were developed by the software development team. They employed a Wi-Fi module (ESP8266) allowing for real-time data transfer and management of devices such as the water pump and grow lights. Users were able to observe live readings, get alerts, and control the system manually as needed through this interface.

This agile methodology not only guaranteed that the system was intelligent, user-friendly, and sustainable, but it also achieved the primary objective of establishing a Plant Optimization solution with the help of IOT and Renewable Energy.

3.2. Project Approach

This project is with hardware-software integration built on ESP8266 / Arduino Mega board powered by a solar panel system with battery and solar controller for sustainable operation. The Agile methodology was employed hence the development was iterative and cooperative composed of the following main steps:

- Research and Requirements Analysis Preliminary research established the relevant environmental factors required for optimal plant growth including temperature, humidity, soil moisture, pH, and nutrient concentrations (NPK).
- Choosing Components Selected appropriate parts to use, such as DHT22, soil moisture sensor, pH sensor, NPK Sensor, ultrasonic water level sensor, water pump, solenoid valve, grow lights, and ESP8266 Wi-Fi module used to communicate.
- All components were constructed with breadboards, jumper wires, and regulated power connections to ensure reliable and safe operation assemble circuit design and wiring.
- Programming the Microcontroller Utilized the Arduino IDE to implement automation logic, manage sensor readings, and facilitate real-time data transmission to the dashboard over Wi-Fi.
- Prototype Testing and Iteration Sensor accuracy, automation triggers, and connectivity were tested. Problems of sensor calibration, power fluctuations, and wireless interference have been dealt with and made the necessary improvements.

3.3. Technologies and Tools used

- Sensors :- DHT22, pH Sensor, Soil Moisture Sensor, NPK Sensor (RS485), Soil Moisture and Temperature Sensor (MTD50), Waterproof Ultra Sonic Sensor (JSN-SR04T), Light Sensor.
- Modules :- Arduino Mega 2560 board, ESP8266 Wi-Fi Module, 2 channel Relay Module, Bug Converter, pH Sensor Module, NPK Sensor Module, Soil Moisture and Temperature Module, Buzzer Module.
- Hardware Components :- Water Pump Motor, Solenoid Valve, Solar Panel, Solar Controller, Inverter (to convert DC to AC), Battery, Circuit Breakers, LCD display, Grow Lights, Nutrient Mixture and Controller, Breadboard, Jumper Wires, Electrical Wires.
- Web based Dashboard :- HTML, CSS and JS used for developed Dashboard.
- Backend :- Using PHP all the real-time data stored in a database. Database implemented using phpMyAdmin with XAMPP server.
- Communication Protocols :- HTTP for reliable, real-time data and command exchange.
- Component Actions :- ESP8266 (HTTP request listens on PORT 80 and send back JSON data) / Dashboard (HTTP Get requests to ESP's IP)

4. Solution

IoT and renewable energy could be used to automate plant care, and we proposed developing a Smart Plant Optimization system that performs this function. The system is based on ESP8266 microcontroller and includes a number of sensors to monitor soil moisture, pH, NPK, temperature and humidity. If the soil moisture drops below optimum levels, the system automatically turns on a water pump via the solenoid valve and then Nutrient Mixture to irrigate the plant.

To provide constant temperatures for the plants to grow, grow lights are also incorporated and can be turned on automatically. The data gained from the sensors are sent as it is received, using Wi-Fi capabilities, allowing users to see the health of the plants and receive alerts.

A solar panel with a battery and charge controller powers this entire system and makes it eco-friendly for use indoors. This approach provides smart plant care with low human effort in a sustainable and effective manner.

Also, this Smart Plant Optimizing System effective for:

- Busy home gardeners who cannot nurturing to their plants daily.
- For large scale operations, handling farms where manual monitoring each plant is impractical.
- Agriculture students can use this as a learning tool, and schools and Universities also can use this as a learning tool.
- Those who aim to reduce water usage and optimize resource management can use this sustainable project as a solution.

5. Testing and Evaluation

5.1. Testing of Prototype

1. Test Dry Soil :- Water Pump successfully ON when the Soil is dry.
2. Test Wet Soil :- Water Pump successfully and remains OFF when the Soil is wet.
3. Monitoring Continuously :- System getting readings in every few seconds to a loop.
4. Accuracy of Sensors :- Soil moisture, pH, NPK, temperature and humidity sensors were all accurate sensors and had good sensor accuracy.
5. Response Automation :- Grow Lights responding correctly by the given commands when the low light.
6. Reliable Transmission of Real-time data :- Data can be relied on to be transmitted real-time to the dashboard with the ESP8266 Wi-Fi module, and users can view sensor readings.
7. System Functioning :- Battery and solar panel kept the system running smoothly in the range of equipments without break.

5.2. Result Discussion

This testing showed that the system is capable of being reliable:

- Monitoring essential plant health parameters
- It brought automation of watering and lighting based on sensor feedback
- Supporting remote monitoring and manual control
- Operating continuously using solar energy

All the sensors and the actuators worked as they should, and all data transmissions are proper. There was an issue with the solar current transmission via the Arduino Mega board but however the group managed to fix it. The testing period had no major hardware and software failures.

5.3. Evaluation

Condition	Expected Result	Result	Status
Dry Soil	Water Pump ON	On	Success
Wet Soil	Water Pump OFF / Remains OFF	Remains OFF / OFF	Success
Data Uploading	Readings to the given commands	Reads to the loop correctly	Success
Grow Lights	Automatically on to the given commands with low light	Automatically working with commands	Success
Continuous Operation	Watering Efficiently and System running without any breaks	All the operations running continuously	Success

Overall system that met all the intended objectives. It is intelligent because it uses real time environmental data to make automated decision. It is a user-friendly remote access and control process, allowing remote access and control via a simple web-based dashboard. Sustainable as it is run fully on solar power and is excellent for the off grid or eco conscious environment.

6. Conclusion

6.1. Future Enhancements

1. Integration of Mobile Application
2. Cloud Logging and Data Analysis
3. SMS or Email Alerting System
4. Setting up a Camera module for Visual Monitoring
5. Automatic Nutrient Distribution
6. AI - Powered Plant health predictions

6.2. Final Conclusion

The Smart Plant Optimization System as an automation project, it is so much more than that, it's a step forward to smarter, more sustainable living. The system integrates real-time monitoring of parameters to automate irrigation and lighting while providing automatic care of plants even from afar, by means of renewing solar power. It is eco-friendly, has intelligent automation and user interface, bridging the gap between technology and nature in a way which saves time, saves water and healthy growth. Besides meeting current needs, the project also further fuels the potential developments of smart agriculture in the future.

7. Appendix

7.1. Codes, Videos and etc

https://liveplymouthac-my.sharepoint.com/:f:/g/personal/10952463_students_plymouth_ac_uk/ErscMvg3NiVOvlwYCPO7O1MBzgLbGwRcZWBOr6uvSsUBKg?e=gI3NFh

7.2. Sample Screenshots

ESP8266_WIFI_DataSending.ino

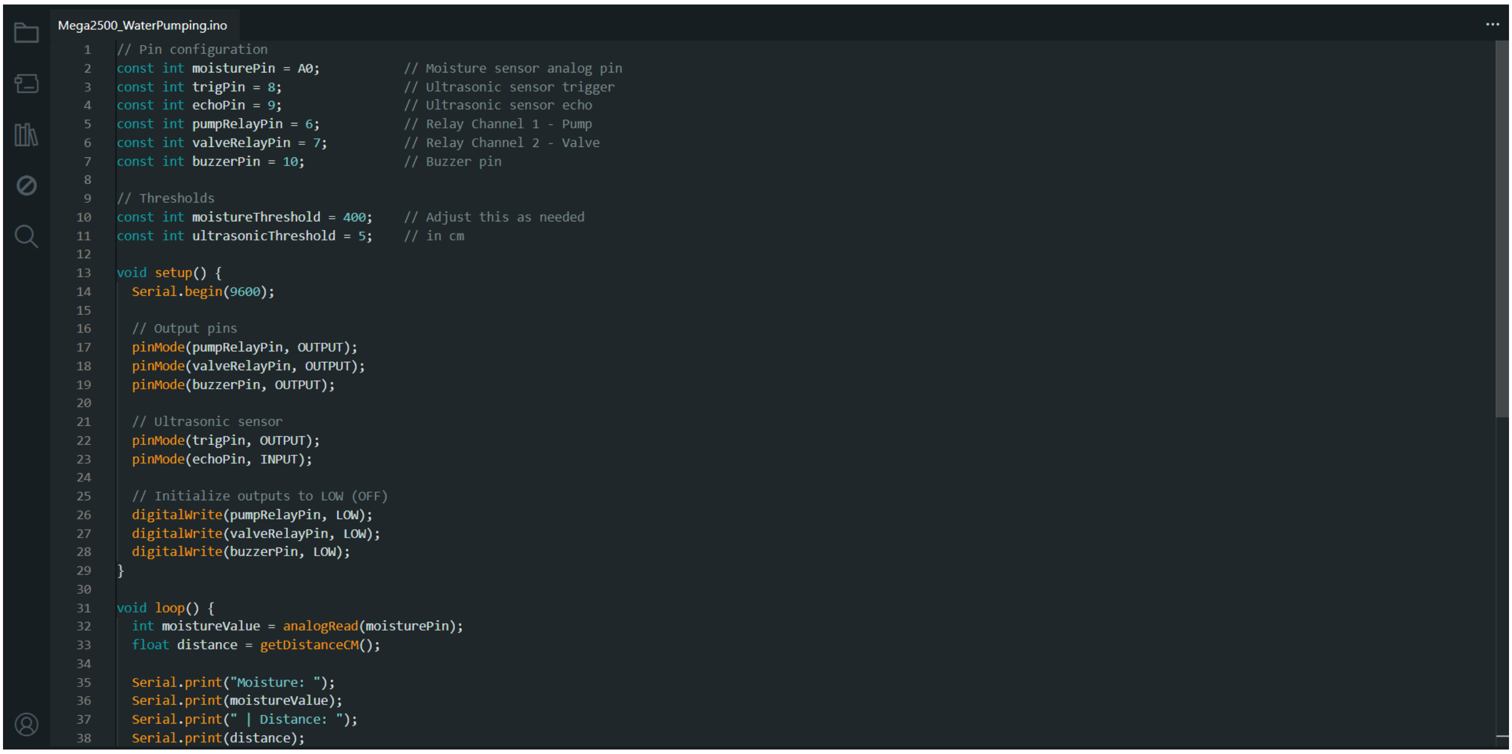
```
#include <ESP8266WiFi.h>
#include <ESP8266WebServer.h>
#include <DHT.h>
#include <SoftwareSerial.h>

// Pins
#define DHTPIN 2            // D4
#define DHTTYPE DHT22
#define PH_SENSOR A0        // Shared analog pin
#define SOIL_SENSOR A0      // Shared analog pin
#define NPK_RX D5           // GPIO14
#define NPK_TX D6           // GPIO12

DHT dht(DHTPIN, DHTTYPE);
ESP8266WebServer server(80);
SoftwareSerial npkSerial(NPK_RX, NPK_TX);  // RX, TX

// WiFi Credentials
const char* ssid = "Dulaj's i13";
const char* password = "dulaj2913";

// WiFi Connection
void connectToWiFi() {
    Serial.print("Connecting to WiFi...");
    WiFi.begin(ssid, password);
    int attempts = 0;
    while (WiFi.status() != WL_CONNECTED) {
        delay(1000);
        Serial.print(".");
        if (++attempts > 20) {
            Serial.println("\n⚠ Failed to connect! Restarting ESP...");
            ESP.restart();
        }
    }
    Serial.println("\n✅ Connected to WiFi!");
    Serial.print("🌍 IP Address: ");
    Serial.println(WiFi.localIP());
```

Figure 1 - ESP8266 Wi-Fi Module

This Arduino code connects ESP8266 microcontroller and creates the live environment data for smart agriculture software. This code used to follow below components :-

- DHT22 Sensor: Measures temperature and humidity.
- Soil Moisture Sensor: Measures soil moisture as a percentage.
- The pH Sensor takes analog signals before determining their pH values.
- The NPK Sensor produces readings of Nitrogen (N), Phosphorus (P) and Potassium (K) content through randomly generated values before implementing actual UART-based communication.
- The web server endpoint located at sensor, provides JSON-formatted sensor readings that serve as excellent input for website monitoring applications or mobile applications as well as real-time monitoring.

ESP8266 links up to Wi-Fi networks through stored authentication while perpetually waiting for client requests before delivering current sensor information. The system configuration works well for implementing IoT agriculture monitoring solutions.



Mega2500_WaterPumping.ino

```
// Pin configuration
const int moisturePin = A0;          // Moisture sensor analog pin
const int trigPin = 8;               // Ultrasonic sensor trigger
const int echoPin = 9;               // Ultrasonic sensor echo
const int pumpRelayPin = 6;          // Relay Channel 1 - Pump
const int valveRelayPin = 7;         // Relay Channel 2 - Valve
const int buzzerPin = 10;            // Buzzer pin

// Thresholds
const int moistureThreshold = 400;    // Adjust this as needed
const int ultrasonicThreshold = 5;    // in cm

void setup() {
  Serial.begin(9600);

  // Output pins
  pinMode(pumpRelayPin, OUTPUT);
  pinMode(valveRelayPin, OUTPUT);
  pinMode(buzzerPin, OUTPUT);

  // Ultrasonic sensor
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);

  // Initialize outputs to LOW (OFF)
  digitalWrite(pumpRelayPin, LOW);
  digitalWrite(valveRelayPin, LOW);
  digitalWrite(buzzerPin, LOW);
}

void loop() {
  int moistureValue = analogRead(moisturePin);
  float distance = getDistanceCM();

  Serial.print("Moisture: ");
  Serial.print(moistureValue);
  Serial.print(" | Distance: ");
  Serial.print(distance);
```

Figure 2 - Arduino Mega 2500

The Arduino code operates a systematic irrigation system through :-

- Soil Moisture Sensor: Reads analog moisture level.
- An Ultrasonic Sensor (HC-SR04) detects distances to verify object existence such as humans or obstacles.
- Relay Module: Controls a water pump and a valve.
- The buzzer sound indicates that the soil contains dry conditions.

How it operates :-

- If soil is dry (moistureValue > 400) :
  - The pump along with the valve together with buzzer begin their operation.
- The soil becomes wet and there exists a near object (distance <= 5CM) :
  - The system disables all outputs in order to conserve power.
- The getDistanceCM() function of the ultrasonic sensor determines distance in centimeters. Real-time debugging and monitoring occur through Serial Monitor that displays sensor values.

8. Workload Matrix

Plymouth Name	Plymouth ID	Contribution
Jayawardena Kavinda	10952523	As the group leader, I coordinated all tasks and support every group member to complete the IOT project.
Dulaj Hewage	10952463	Responsible for assembling components and make sure all sensors are connected before testing.
Unagollage Wijesinghe	10952470	Contribute to coding of sensors data collection.
Witharamalage Sirimewan	10952545	Act as a bridge between software and hardware.
Pathiranage Didula Theekshana	10952565	Quickly identified faults in the system and give some solutions.
Duwage Perera	10953075	Giving best suggestion to all group members to improve the system.
Chamathka Abeykoon	10952645	Do some research about farming and select valuable components.
Suwanda Akarsha	10953214	Gave the support to finding resources.
Rathnayaka Rathnayaka	10952629	Monitoring all the plants and watering system in the project.
Yaddehige Himasha	10952566	With great communication skills and shows great effort.

9. References

bere_an4, 2024. *Irrigation system project.* [Online] Available at: https://forum.arduino.cc/t/irrigation-system-project/1288487

Dirksen, K., 2017. *FarmBot: open-source backyard robot.* [Online] Available at: https://www.youtube.com/watch?pp=0gcJCfcAhR29_xXO&v=BqYrAWssrrY

EveryNeighborhooddog, 2025. *Smart Irrigation - Beginner's project.* [Online] Available at: https://www.reddit.com/r/arduino/comments/1iow9wc/smart_irrigation_beginners_project/?rdt=39915

Ferro, S., 2017. *5 Smart Gardening Devices to Turn Your Thumb Green.* [Online] Available at: https://www.mentalfloss.com/article/503431/5-smart-gardening-devices-turn-your-thumb-green

Gebhart, A., 2014. *Oso Technologies PlantLink review.* [Online] Available at: https://www.cnet.com/reviews/oso-technologies-plantlink-review/

KLM, P. T., 2023. *Smart irrigation system with Arduino Uno.* [Online] Available at: https://www.youtube.com/watch?v=7KdhL_gAqHQ

lyfeline, 2024. *Soil moisture sensor with relay and pump.* [Online] Available at: https://forum.arduino.cc/t/soil-moisture-sensor-with-relay-and-pump/1236761

Maind, M., 2023. *How to make a smart irrigation system.* [Online] Available at: https://srituhobby.com/how-to-make-a-smart-irrigation-system-with-arduino-uno-r4-wifi-board/?srsltid=AfmBOoonM_Tq2XlbuHu9A1cw54LSkd00cnAgvapDGDs9dLBtX9gFSc_-

Michael, n.d. *https://nosoilsolutions.com/aerogarden-system-guide/.* [Online] Available at: https://nosoilsolutions.com/aerogarden-system-guide/

Mjoen, P., 2019. *Plantlink Support Discontinued.* [Online] Available at: https://community.smartthings.com/t/plantlink-support-discontinued/153055

TechPunt, 2024. *The Xiaomi Mi Flower Care Plant Sensor.* [Online] Available at: https://www.youtube.com/watch?v=T7GsC6ACZ98