Project Report

"IOT BASED HYDROPONIC SYSTEM FOR SUSTAINABLE GROWTH OF PLANTS"

PHY1901

Introduction to Innovative Projects

Fall Semester (2021-2022)

TG1 Slot

Submitted to:

Prof. Dr. KALAINATHAN S

Index

Page No.

- **Team Members** 3
- **Problem Statement** 4
    - **Aim** 4
- **Apparatus** 4
    - **Hardware** 4
    - **Software** 6
- **Methodology** 6
    - **Circuit Diagram** 6
    - **Working Principle** 7
    - **Arduino Code** 12
- **Outcome of the Project** 15
    - **Graphs Obtained** 16
- **Innovative Applications** 18
- **References** 19

Team Members

- VISWESHWAR S - 19BEE0123
- ARJUN DAMARAJU - 19BEE0322
- ESWAR S S - 19BEE0133
- ANISH KUMAR P - 19BEE0135
- HARENDHAR K - 19BEE0140
- SWARUP TRIPATHY - 19BEE0167
- VIJYANT THAKUR - 19BEE0281
- DEEPAK KUMAR - 19BEE0332
- SNEHIL PATEL - 19BEE0164
- ASHIR - 19BEE0125

Problem Statement

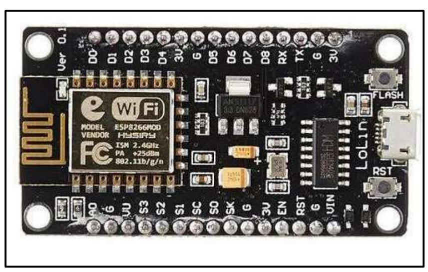
Agriculture in India is considered to be the roots of economy for the country and yet it is one of the most underdeveloped sector. The world is reaching to a stage where technologies have been advanced like never before and even after that we haven’t cultured the habit of giving importance to the agricultural sector of India.

According to a source, Agriculture is the primary source of livelihood for about 58% of India’s population.

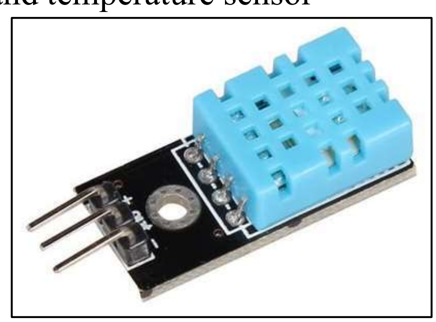
- **Aim**
  - Implement an IOT based hydroponic system to ensure the sustainable growth of plants with the help of water and nutrients without soil

Apparatus

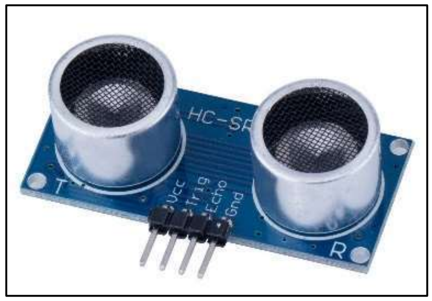
- **Hardware**
  - ESP8266 NodeMCU module with WiFi



  - DHT11 humidity and temperature sensor

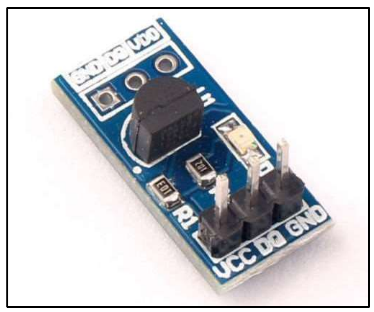


- Ultrasonic Sensor

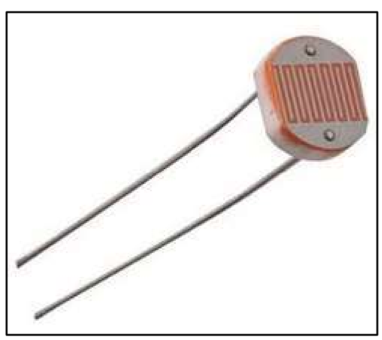


- pH Sensor Board

- DS18B20 waterproof temperature sensor



- LDR Sensor



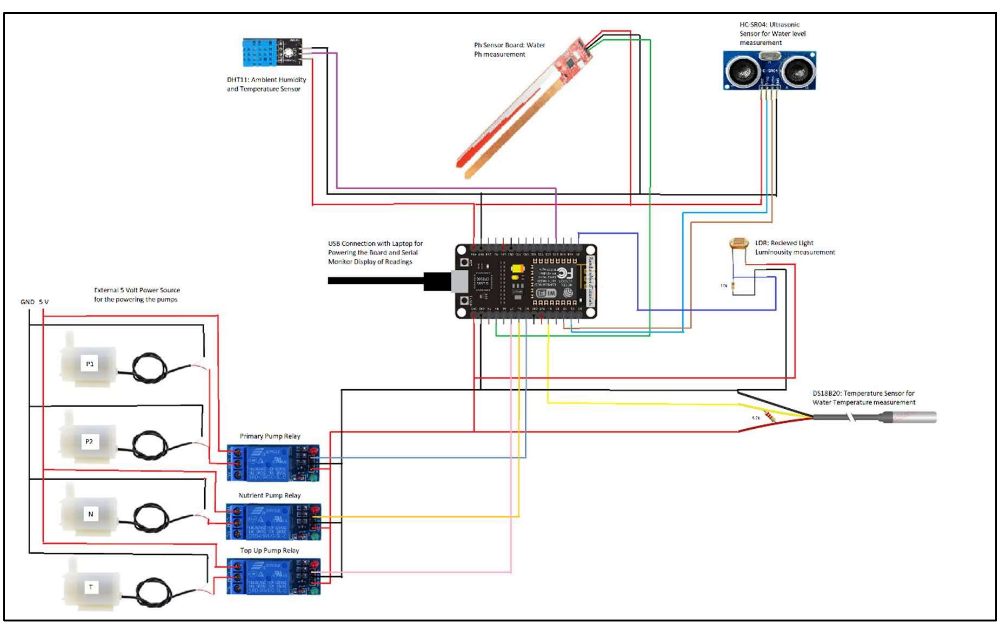
- 3-inch pipes with end caps and support pipes
- 5-volt relay
- 5-volt adapter for pumps
- 5-volt pump:
  - 2 – primary water supply
  - 1 – nutrient supply
  - 1 – water top up

- **Software**
  - Arduino Integrated Desktop Environment

  - ThingSpeak server

Methodology

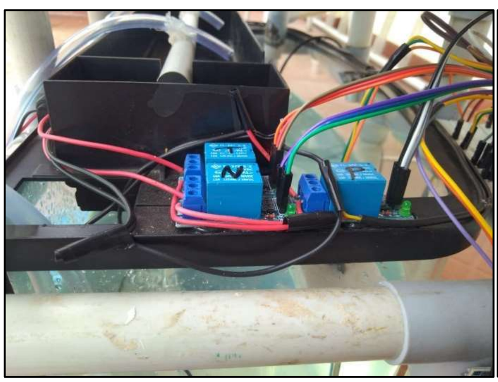
- **Circuit Diagram**

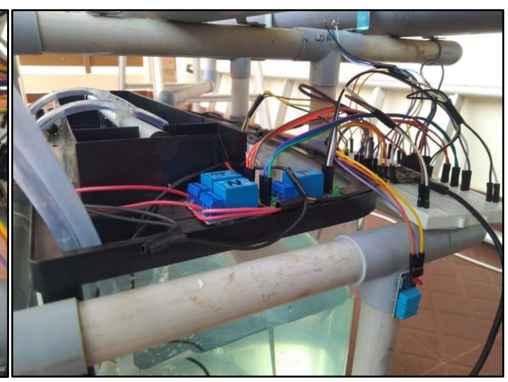


▪ Working Principle

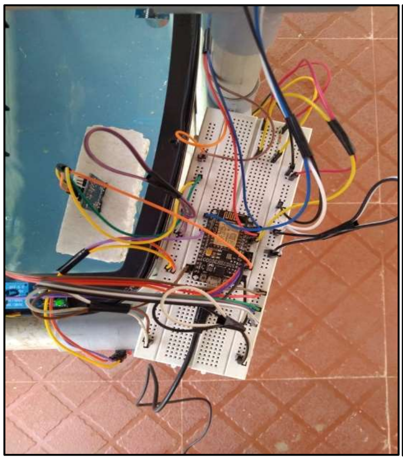
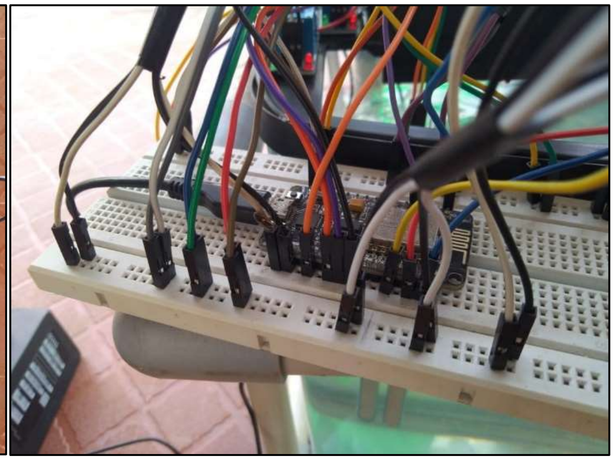
The connections made are as shown above. The connections for the primary, nutrient and top up pump are as shown in the above circuit diagram. The temperature sensor used is waterproof and is directly connected to the NodeMCU which send the readings to Thingspeak. The two primary pumps are connected to a single relay as it makes the controlling of the individual pumps a lot simpler. The nutrient and top up pumps are each connected to individual relays to ensure their isolated working and ease in fault detection.

Water is contained in a tank. Two primary motors have been installed here. These motors pump water from the tank to three-inch pipes. The plants are located inside the three-inch pipes.

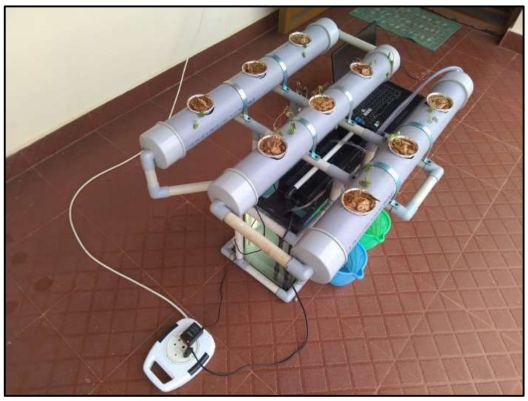
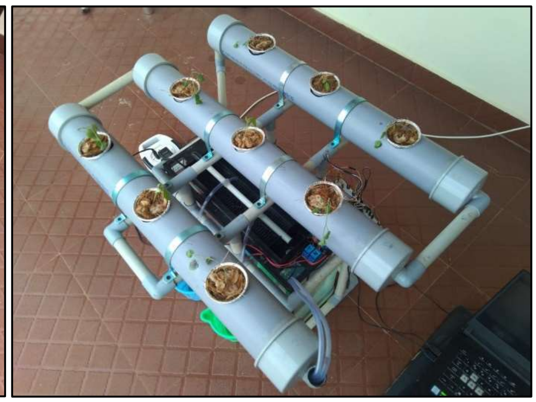




Two primary pumps are connected to a normally closed relay. A normally closed relay has been chosen as the pumps need to be running for a long period of time continuously. As long as power is available, the primary pump runs regardless of input from the microcontroller.

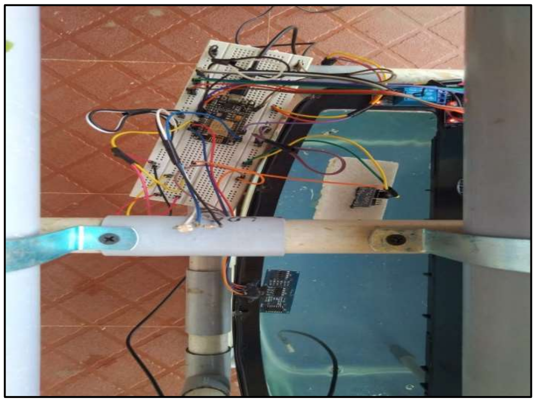
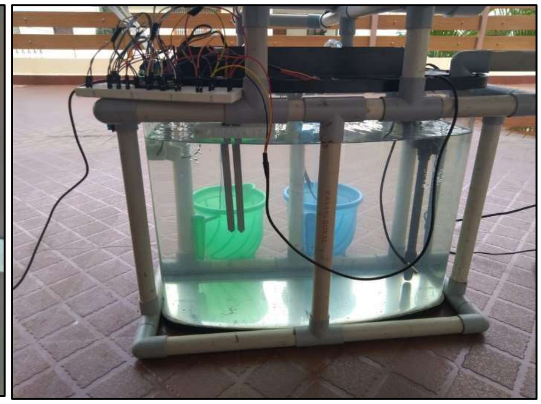
These pumps keep running unless the microcontroller signals it to stop, ensuring that there is continuous water circulation.

Water is supplied from the primary pump to the entire system which supplies the adequate nutrients and water to all the plants in the system. In this project, nine plants are placed in series. Once the water is utilized by the plants, it is taken back to the reservoir tank where it is stored. It is in this reservoir tank where the water is treated with an influx of oxygen and the required nutrients.

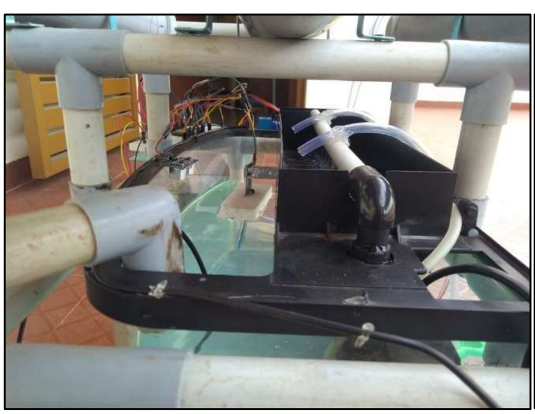
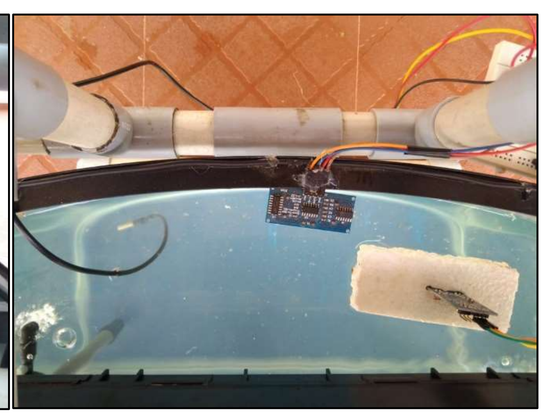
 

The aerator present in the reservoir tank, oxygenates the water in the reservoir tank before it is used once again for circulation, as plants require oxygen from the water as well to ensure proper growth.

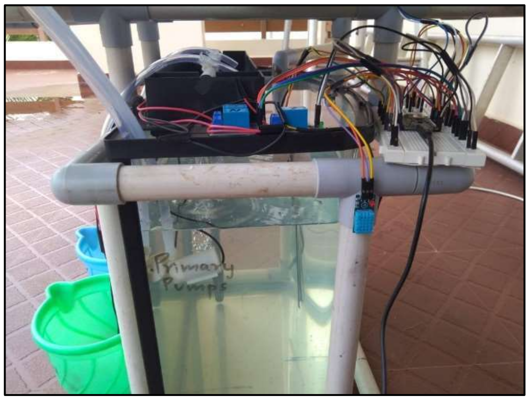
The pH sensor which is suspended on the water with the help of a floating piece of Styrofoam, continuously takes the pH reading of the water. Using this data, the corresponding nutrient level of the water is analyzed thoroughly


The readings from the Ph sensor, once analyzed decides the level of nutrients to be supplied to the water. The nutrient pump is activated accordingly, to supply the required nutrients into the water. The water temperature sensor present, continuously measures the temperature of the water as even a slight increase or decrease in the water temperature from the required ambient condition would impede the proper growth of the plant.

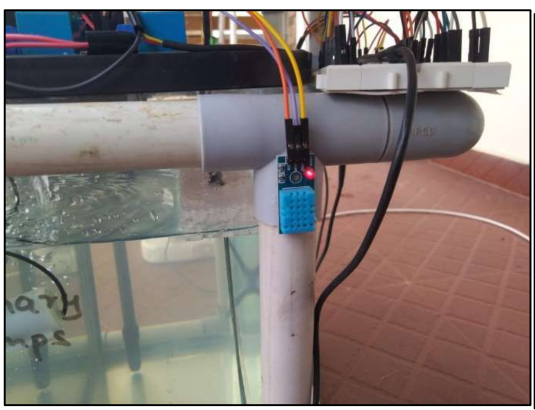
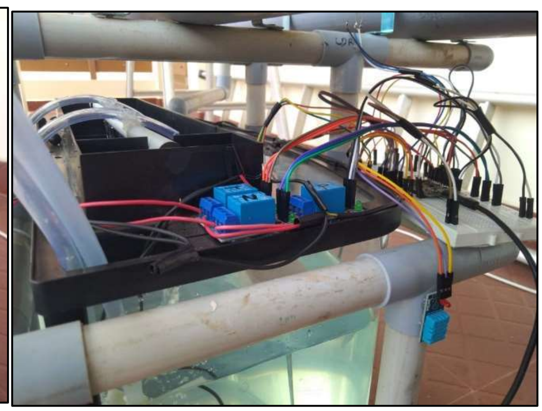



An ultrasonic sensor is used to measure the water level in the tank. The water level is crucial for the proper growth of plants as a slight increase in water levels would affect their growth. A drop of the water level below a particular set threshold would hinder the proper functioning of the primary pumps and the aerator. Hence, this is safety system has been put in place.

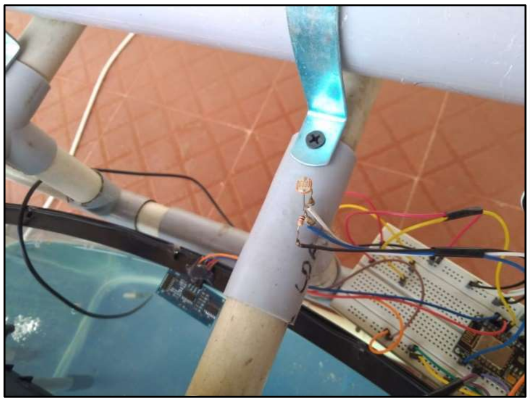
A top up pump is also present in the hydroponic setup. The purpose of this pump, primarily, is to ensure the smooth functioning of the primary pump and the aerator. The top up pump is activated whenever necessary to resume normal functioning of the setup.



If the water level exceeds a particular a particular threshold, the primary motor stops immediately. This is to avoid overflow. The primary pumps return to normal functioning once nominal water levels are detected.

The temperature of the environment in which the plants grow is a very important factor which affects their growth. Hence, a DHT11 humidity and temperature sensor is used to continuously record the ambient temperature of the environment of the hydroponic system in which the plants are present.



Sunlight is vital to the growth of plants and measuring the amount of sunlight that the plants are exposed is crucial to judge the health of the plants. An LDR sensor is being used to record the amount of sunlight received by the plant.

Each and every sensor used above measures parameters that are critical to the health of the plant. Having a visual representation of the data that we receive from these sensors will help us to know better about the health of the plant.

Hence the data received from each and every sensor present in the system is sent to a Thingpeak server. This tool takes data as input and depicts the data graphically. This aids in real-time monitoring of the plants. The graphs are updated every two minutes.

Arduino Code

```
#include <OneWire.h> // Library for DS18B20
#include <DallasTemperature.h> // Library for DS18B20
#include <DHT.h>  // Library for DHT11
#include <SPI.h> // Library for ThingsSpeak
#include <ESP8266WiFi.h> // Library for Wifi Contectivity

//*******************************************************************************************************//
String apiKey = "9SPBFR1UIHPXJQC9"; //  Enter your write API key from ThingSpeak
const char *ssid =  "Rajvihas"; // Replace with your wifi ssid and wpa2 key
const char *pass =  "bajrang1973";
const char* server = "api.thingspeak.com";

//*******************************************************************************************************//
const int trigPin = 5; //D1 Defines the Trigger Pin (HC-SR05)
const int echoPin = 4; //D2 Defines the Echo Pin (HC-SR05)

const int oneWireBus = 2; // D4 GPIO where the DS18B20 is connected to
OneWire oneWire(oneWireBus); // Setup a oneWire instance to communicate with any OneWire devices
DallasTemperature sensors(&oneWire); // Pass our oneWire reference to Dallas Temperature sensor

#define DHTPIN 10 // The pin where the DHT11 is connected
DHT dht(DHTPIN, DHT11);

const int LDR = A0; // Defining LDR PIN

const int relayP = 14; // D5
const int relayN = 12; // D6
const int relayT = 13; // D7

//*******************************************************************************************************//
float duration = 0;
float distance = 0;
String phdata;
String ph;
```

```
float phvalue = 0;
float watertempC = 0;
float hum = 0;
float temp = 0;
int lum = 0;

//*******************************************************************************************************//
WiFiClient client;
void setup() {

  Serial.begin(9600); // Start the Serial Monitor
  pinMode(trigPin, OUTPUT); // Sets the trigPin as an Output (HC-SR05)
  pinMode(echoPin, INPUT); // Sets the echoPin as an Input (HC-SR05)
  sensors.begin(); // Start the Water Temperature sensor (DS18B20)
  dht.begin(); // Start the Humidity and Temperature sensor (DHT11)
  pinMode(relayP,OUTPUT);
  pinMode(relayN,OUTPUT);
  pinMode(relayT,OUTPUT);

//*******************************************************************************************************//
  // Connecting to Wifi with ssid and password
  Serial.println("Connecting to ");
  Serial.println(ssid);
  WiFi.begin(ssid, pass);
  while (WiFi.status() != WL_CONNECTED)
     {
       delay(500);
       Serial.print(".");
     }
   Serial.println("");
   Serial.println("WiFi connected");

}
```

```
void loop() {

  digitalWrite(relayN,HIGH);
  digitalWrite(relayT,HIGH);
  Serial.println("Nutrient and Top up pump are off");

//*****************************************************************************************************//
  // The Ultrasonic sensor measurement (HC-SR05)
  digitalWrite(trigPin, LOW); // Clears the trigPin
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH); // Sets the trigPin on HIGH state for 10 micro seconds
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  duration = pulseIn(echoPin, HIGH); // Reads the echoPin, returns the sound wave travel time in microseconds
  distance= duration*0.034/2; // Calculating the distance
  Serial.print("Water Level: "); // Prints the distance on the Serial Monitor
  Serial.println(distance);
  if (distance>2 && distance<7) {
    Serial.println("Water level is Nominal");
    digitalWrite(relayP,LOW);
    Serial.println("Primary pump on");
    delay(6000);
  }
  else if (distance<2) {
    Serial.println("Water level is too High");
    digitalWrite(relayP,HIGH);
    Serial.println("Primary pump off");
    delay(6000);
  }
```

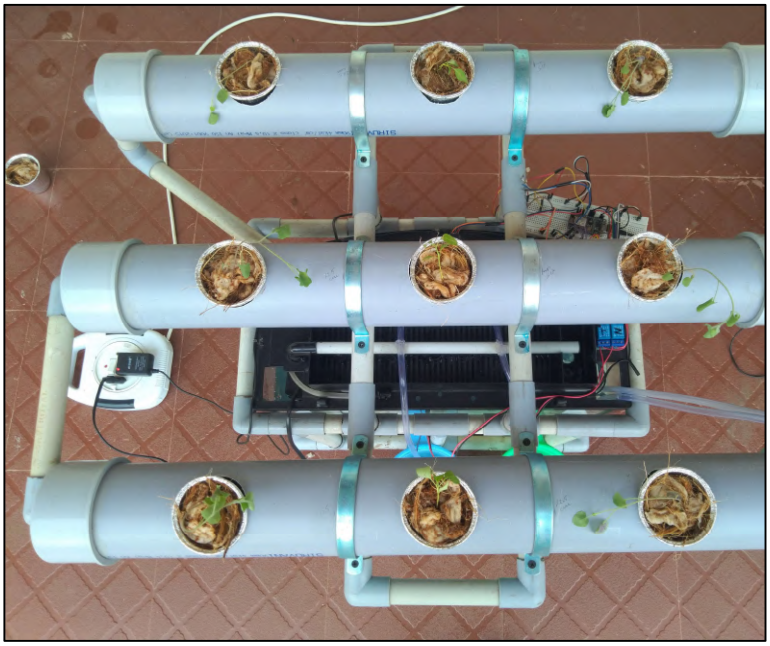
```
  else if (distance>7) {
    Serial.println("Water level is too Less");
    digitalWrite(relayP,LOW);
    digitalWrite(relayT,LOW);
    Serial.println("Primary pump on");
    Serial.println("Top up pump on");
    delay(6000);
    digitalWrite(relayT,HIGH);
    Serial.println("Top up pump off");
  }

//*****************************************************************************************************//
  // The PH Sensor Board measurement
  for(int i=1; i<=10; i++) {
        phdata = Serial.readStringUntil(':');
        if(phdata != ""){
            ph = Serial.readStringUntil('$');
            phvalue = ph.toFloat();
        }
  }
  Serial.println("Ph Sensor Board Value: ");
  Serial.println(ph);
  Serial.println("Ph Value: ");
  Serial.println(phvalue);
  if (phvalue>7 && phvalue<10) {
    Serial.println("Ph value of water is Nominal");
    delay(6000);
  }
```

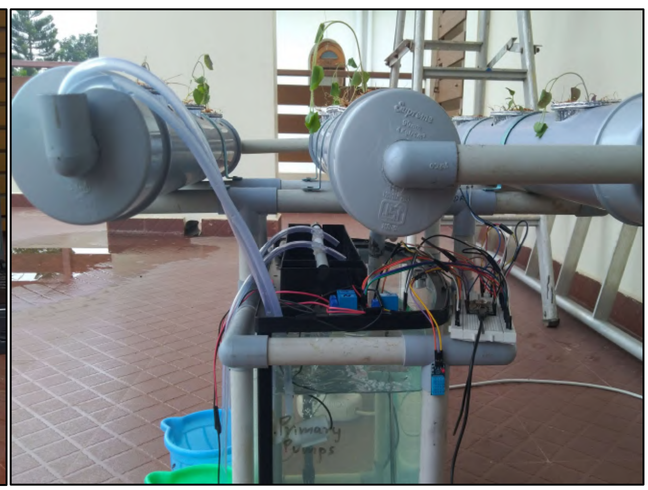
```
  else if (phvalue>0 && phvalue<7) {
    Serial.println("Ph value of water is Acidic");
    digitalWrite(relayT,LOW);
    Serial.println("Top up pump on");
    delay(6000);
    digitalWrite(relayT,HIGH);
    Serial.println("Top up pump off");
  }
  else if (phvalue>10 && phvalue<14) {
    Serial.println("Ph value of water is Basic");
    digitalWrite(relayN,LOW);
    Serial.println("Nutrient pump on");
    delay(6000);
    digitalWrite(relayN,HIGH);
    Serial.println("Nutrient pump off");
  }
  else {
    Serial.println("Ph Sensor Board is out of the water");
    delay(6000);
  }

//*****************************************************************************************************//
  // The Water Temperature measurement (DS18B20)
  sensors.requestTemperatures();
  watertempC = sensors.getTempCByIndex(0);
  Serial.println("Water Temperature: ");
  Serial.print(watertempC);
  Serial.println("°C");
  delay(6000);
```

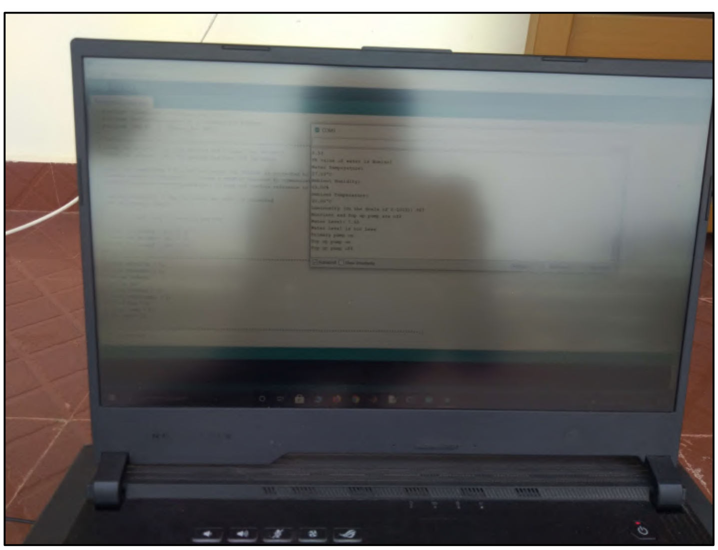
```
//*************************************************************************************************************//
  // The Humidity and Temperature measurement (DHT11)
  hum = dht.readHumidity();
  temp = dht.readTemperature();
  Serial.println("Ambient Humidity: ");
  Serial.print(hum);
  Serial.println("%");
  Serial.println("Ambient Temperature: ");
  Serial.print(temp);
  Serial.println("°C");
  delay(6000);

//*************************************************************************************************************//
  // The LDR sensor measurement
  lum = analogRead(LDR); // Reading Input
  Serial.print("Luminosity (On the Scale of 0-1023): " );
  Serial.println(lum); // Writing input on serial monitor.
  delay(6000);

//*************************************************************************************************************//
```

```
 // Connecting to ThingSpeak server
 if (client.connect(server,80))   //   "184.106.153.149" or api.thingspeak.com
 {
     // The field specification for each sensor value
     String postStr = apiKey;
     postStr +="&field1=";
     postStr += String(phvalue);
     postStr +="&field2=";
     postStr += String(watertempC);
     postStr +="&field3=";
     postStr += String(hum);
     postStr +="&field4=";
     postStr += String(temp);
     postStr +="&field5=";
     postStr += String(distance);
     postStr +="&field6=";
     postStr += String(lum);
     postStr += "\r\n\r\n";

     // The ThingSpeak credentials
     client.print("POST /update HTTP/1.1\n");
     client.print("Host: api.thingspeak.com\n");
     client.print("Connection: close\n");
     client.print("X-THINGSPEAKAPIKEY: "+apiKey+"\n");
     client.print("Content-Type: application/x-www-form-urlencoded\n");
     client.print("Content-Length: ");
     client.print(postStr.length());
     client.print("\n\n");
     client.print(postStr);
  }
 client.stop();
 Serial.println("Waiting...");
 delay(15000); // ThingSpeak needs minimum 15 sec delay between updates
 }
```

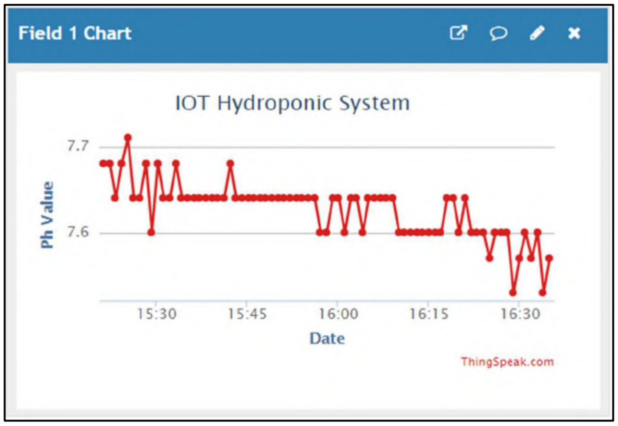
Outcome of the Project



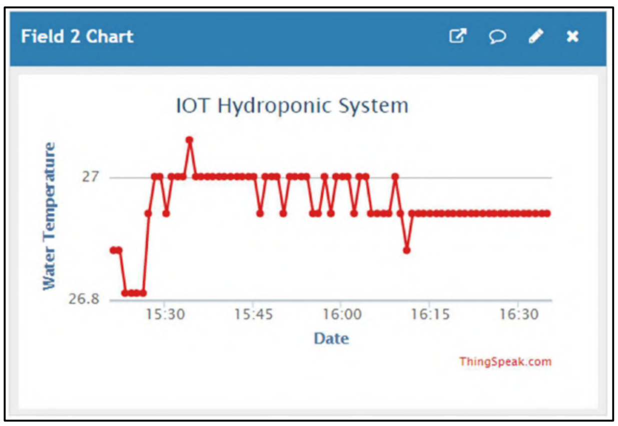




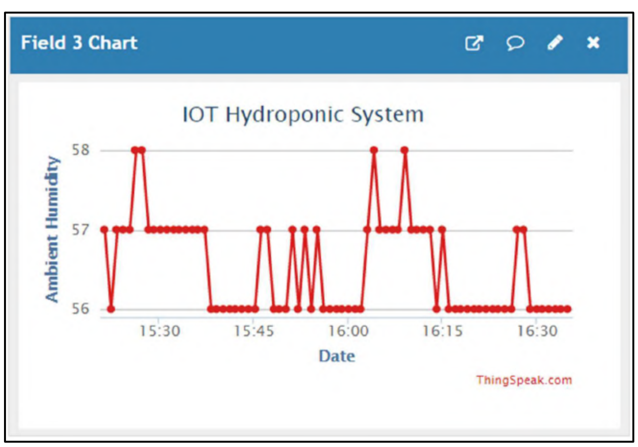
Graphs Obtained



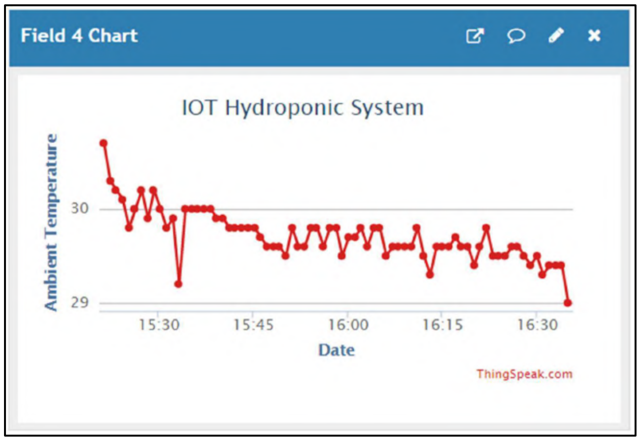
The above graph is a plot of the Ph value of the water that is being continuously measured. A part of the measurement graph is shown here.



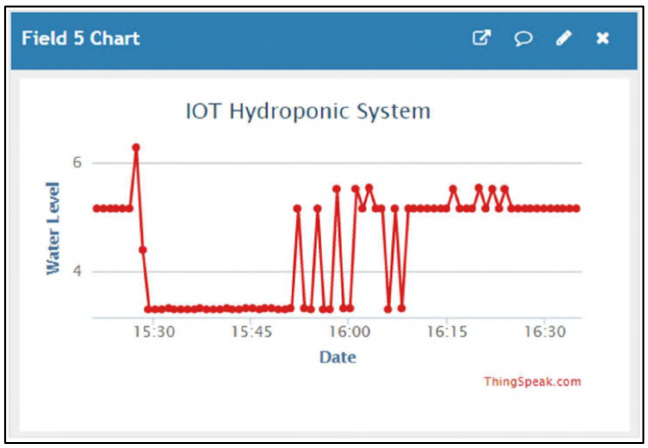
The above graph is a plot of the water temperature that is being continuously monitored. A part of the measurement is shown here.



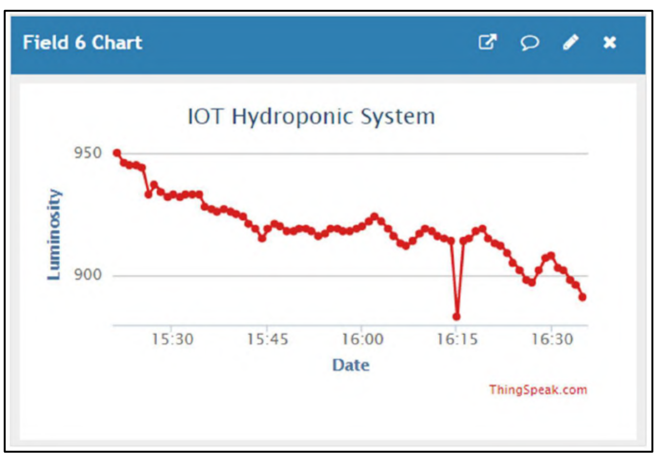
The above graph is a plot of ambient humidity that is being monitored continuously. A part of the measurement is shown here.



The above graph is a plot of ambient temperature that is being monitored continuously. A part of the measurement is shown here.



The above graph is a plot of the water level that is being continuously monitored. A part of the measurement is shown here.



The above graph is a plot of the luminosity that is being continuously monitored. A part of the measurement is shown here.

Innovative Applications

- Hydroponic plant systems reduce the amount of water consumed as the water is continuously recycled. This is a more convenient way to grow plants in a closed environment or in malls, restaurants etc.
- Hydroponic systems can also be used for water treatment as has been successfully demonstrated by researchers from Singapore.
- Since the entire setup has been made, all we have to see is the growth of plants in our system which will take some weeks
- With the current modern technique, this will help the users to easily monitor the overall stats via their system and not be a hindrance to the system.
- Here as you would have observed 'No soil' has been involved. Since hydroponic farming involves growing crops without soil, it is an ideal option for anyone who has limited accessibility to land.
- Further our setup will ensure faster growth rate and zero weed requiring less labour and producing higher yields at the same time.

References

- **Dr Prathima V.R, Vandhana Shree G S, Lavanya R, Mounika J, "SMART HYDROPONIC FRAMING SYSTEM USING IOT", International Research Journal of Engineering and Technology (IRJET), Volume: 07, Issue: 07, July 2020.**
- **Ahsan Ullah, Sadia Aktar, Nipa Sutar, Rafsan Kabir, Afzal Hossain, "Cost Effective Smart Hydroponic Monitoring and Controlling System Using IoT", Intelligent Control and Automation, Vol. 10, Pg. 142-154, October 2019.**
- **Ravi Lakshmanan, Mohamed Djama, Sathish Kumar Selvaperumal, Raed Abdulla, "Automated smart hydroponics system using internet of things", International Journal of Electrical and Computer Engineering (IJECE), Vol. 10, No. 6, pp. 6389~6398, December 2020.**
- **Falmata Modu, Adam Adam, Farouq Aliyu, Audu Mabu, Mahdi Musa, "A Survey of Smart Hydroponic Systems", Advances in Science, Technology and Engineering Systems Journal, Vol. 5, No. 1, Pg: 233-248, January 2020.**
- **R.Rajkumar, R.Dharmaraj, "A Novel Approach for Smart Hydroponic Farming Using IoT", International Journal of Engineering Research in Computer Science and Engineering (IJERCSE), Vol. 5, Issue. 5, May 2018.**
- **Nianpu Li, Yimeng Xiao, Lei Shen, Zhuoyue Xu, Botao Li, Chongxuan Yin, "Smart Agriculture with an Automated IoT-Based Greenhouse System for Local Communities", Advances in Internet of Things, Vol. 9, Pg. 15-31, April 2019.**
- **freshwatersystems.com**
- **homehydrosystems.com**
- **purehydroponics.com**
- **diynetwork.com**
- **verticalroots.com**