VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

**Fall Semester 2022-23**

**School of Electronics Engineering**

# Title: Smart Plant monitoring system

**Course Code:** ECE3501

**Course Title:** IOT Fundamentals

**Slot:** L3+L4

**Ankit Raj - 20BEC0766**

**Devika Nair M - 20BEC0241**

**Ganesh Nuthalapati - 20BEC0146**

## Abstract

Nowadays, most families start to do gardening because they can grow vegetables and fruits or any other plants that they want in their day-to-day life. So, they can survive without spending money on online grocery shopping for fruits and vegetables. Most of the families were trying to do gardening for their needs. The major concern is people cannot monitor their plants every time and protect their gardens. So, we decided to automate the garden work. With our new solution, gardeners can monitor some important factors like the

Plant's healthiness, soil moisture level, air humidity level, and the surrounding temperature and water their garden from anywhere in the world at any time by using our app and website. Plant health has a significant impact on plant development, production, and quality of agricultural goods. Users can also make a watering schedule so that our unit will automatically water the plants by considering several factors and they do not have to manually water the plants every day.

## Introduction:

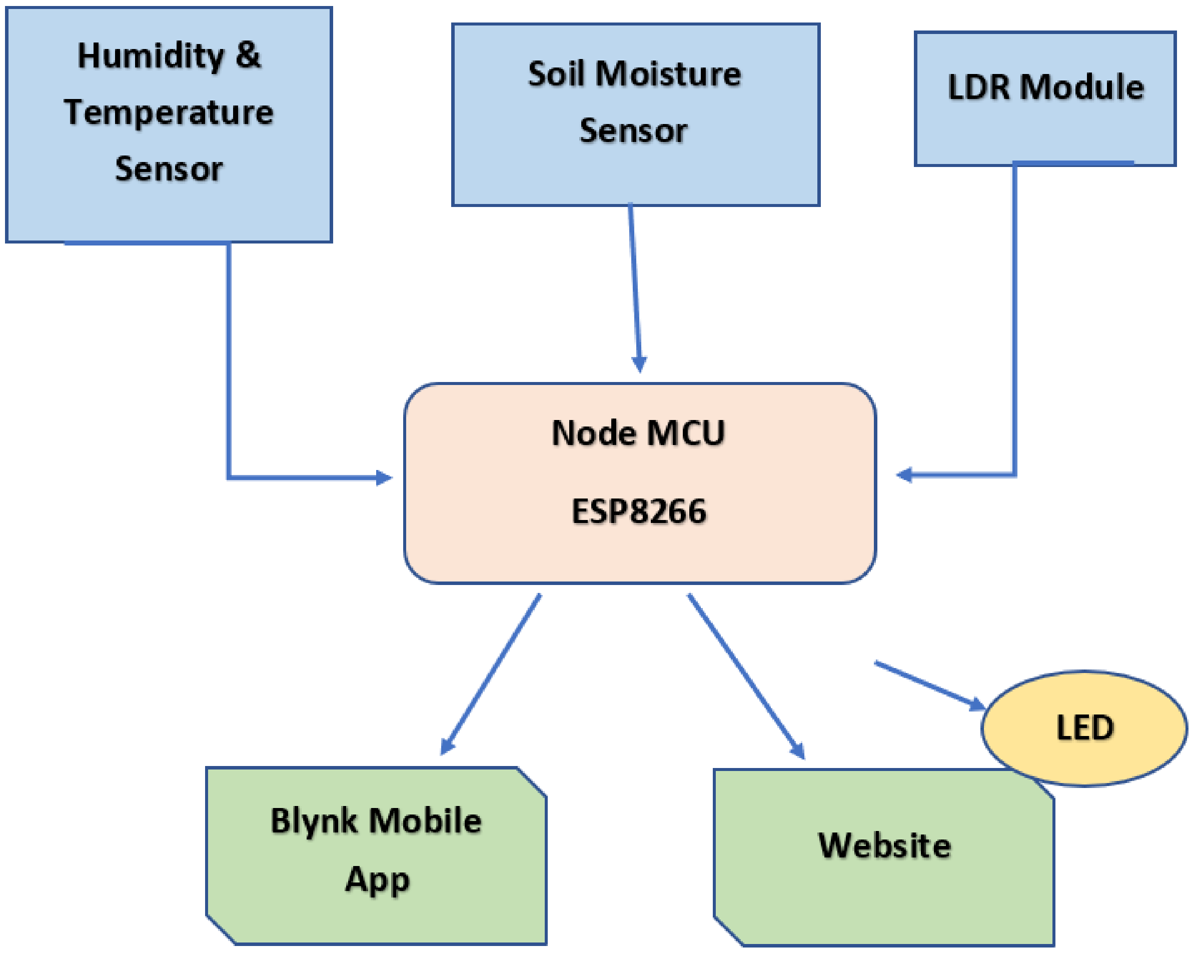
Most families nowadays started gardening since they may grow whatever veggies, fruits, or other plants they desire in their daily lives. As a result, they will be able to make it through this period without having to spend money on online grocery shopping for fruits and vegetables. We discussed the issue and attempted to develop a solution that would allow them to continue working while doing so. The main issue is that individuals cannot constantly watch their plants to safeguard their gardens. As a result, we decided to assist them by providing a plant monitoring system. Gardeners may use our innovative solution to monitor critical parameters such as plant health and illness, soil moisture level, air humidity level, and surrounding temperature, as well as light up the surroundings when the light goes down. The current approach is based on observation with the naked eye, which is a time-consuming process. Users can manually water the plants by checking the soil moisture level at any time and from anywhere in the world, the user may also create a watering schedule so that our device will automatically water the plants based on several criteria, eliminating the need to water the plants manually every day.

## Literature Review

The smartphone's processing and connection components, such as the Wi-Fi network, were controlled directly by an Android App. The smartphone is activated by the mobile App, which wakes it up according to user-defined settings.

In India, about 35% of the land was reliably irrigated. And the 2/3rd part of the land is dependent on monsoon for the water. Irrigation reduces dependence on monsoon, improves food security, and improves the productivity of agriculture and it offers more opportunities for jobs in rural areas. Farmers are facing problems related to the watering system. how much water has to be supplied and at what time? Sometimes over watering causes damage to crops as well as the waste of water. Hence to avoid such damage we need to maintain an approximate water level in the soil.

## BLOCK DIAGRAM:



## BLOCK DESCRIPTION:

### Humidity & Temperature Sensor:

Measure the temperature and humidity of surrounding air and send it to Node MCU.

### Soil Moisture Sensor:

Measure moisture present in soil and send it to Node MCU.

### LDR Module:

To provide the proper amount of lighting to plants and to carry out projects like indoor gardening a plant lighting system is required.

### Blynk Mobile APP:

Mobile Application used to present real-time values to moisture, humidity, Temperature & and amount of light present.

### Firebase - firestore real-time database:

We use the firebase real-time database free tier to upload the real-time data from our node MCU to firebase using google auth.

### Website:

Real-time data collected by sensors and transmitted through Node MCU is displayed using Firebase Website.

## Procedure:

A DHT11 sensor is used to measure the temperature and humidity of the surrounding air. Soil moisture sensors measure moisture present in the soil. The LDR module is used to sense light present in the surroundings and is needed to detect day and night. At night, LED lights are automatically turned ON.

All the data collected by sensors are sent to the Node MCU ESP8266 module which processes and displays an alert message. The Blynk mobile app is used to display the real-time values collected by the sensors. The same data is transmitted over the internet to firebase's real-time database. From the firebase database, the data is fetched using javascript and is displayed on the website. CSS is used in designing the website. In this way, users can check the condition of soil from anywhere via the internet via mobile or web application.

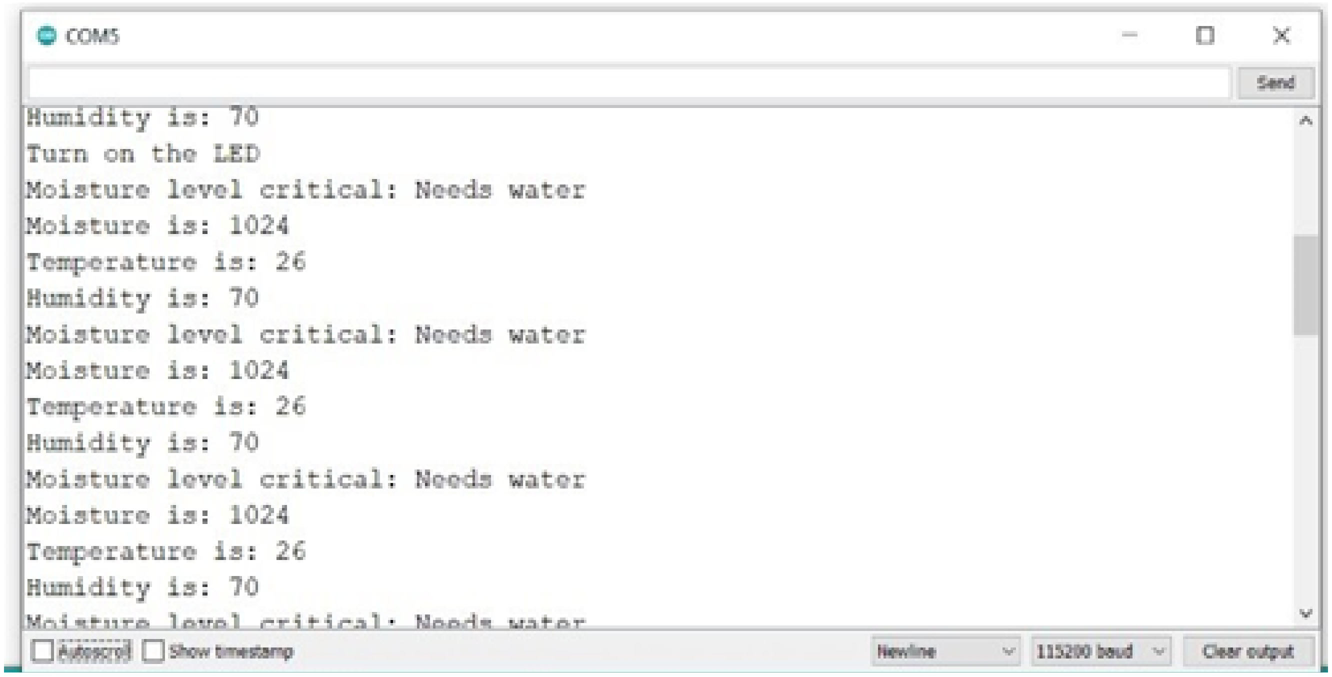
.

## CODE USED TO PROGRAM NODE MCU:

```
// IOT Smart Plant Monitoring System
#define BLYNK_TEMPLATE_ID "TMPLR3JIQe6F"
#define BLYNK_DEVICE_NAME "Smart Plant Monitoring"
#define BLYNK_PRINT Serial
#include <SPI.h>
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
#include <SimpleTimer.h>
#include <DHT.h>
#include <FirebaseArduino.h>
#define BLYNK_PRINT Serial

#define PROJECT_ID "plantmonitoring-f8a5c"   // Your Firebase Project ID.
Can be found in project settings.
```

```
#define FIREBASE_HOST
"plantmonitoring-f8a5c-default-rtdb.asia-southeast1.firebasedatabase.app"
#define FIREBASE_AUTH "eNE5ADWm7FJcjuFyiuS7eUT8ubcTj9yVnT4skY2Q"
#define WIFI_SSID "Devika"
#define WIFI_PASSWORD "pqrsdevika"

char auth[] ="szrkMwc7NpFX1s_POXNROPLFz2z5YtNS";
//Authentication code sent by Blynk
char SSID[] ="abcd";                                        //WiFi SSID
char PASSWORD[] ="1234";                                      //WiFi Password

#define soilWet 600
#define soilDry 900
int sensor=A0;
int h,t,m,h1,t1,m1;
const int ldrPin = D5;
const int ledPin = D0;
#define DHTPIN 2
#define DHTTYPE DHT11
DHT dht(DHTPIN, DHTTYPE);
SimpleTimer timer;
//Firebase firebase(PROJECT_ID);    // SLOW BUT HASTLE-FREE METHOD FOR LONG
TERM USAGE. DOES NOT REQUIRE PERIODIC UPDATE OF FINGERPRINT
void setup()
{
  Serial.begin(9600);
  Blynk.begin(auth, SSID, PASSWORD);
  pinMode(A0, INPUT);
  dht.begin();
  timer.setInterval(1000L, sendSensor);
  Serial.begin(115200);
  Blynk.begin(auth, SSID, PASSWORD);

 WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
  Serial.print("connecting");
  while (WiFi.status() != WL_CONNECTED) {
    Serial.print(".");
    delay(500);
  }
  Serial.println();
  Serial.print("connected: ");
  Serial.println(WiFi.localIP());

  Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);
}

void sendSensor()
{
  h = (int) dht.readHumidity();
  t = (int) dht.readTemperature();
  m = analogRead(sensor);

  if (isnan(h) || isnan(t)) {
    Serial.println("Failed to read from DHT sensor!");
    return;
  }

  Blynk.virtualWrite(V5, h);  //V5 is for Humidity
  Blynk.virtualWrite(V6, t);  //V6 is for Temperature
  Blynk.virtualWrite(V2, m);  //V6 is for Moisture
}

void loop()
{
  Blynk.run();
  timer.run();
  h1 = (int) dht.readHumidity();
  t1 = (int) dht.readTemperature();
  m1 = analogRead(sensor);
  Serial.print("Moisture is: ");
  Serial.println(m1);
  Serial.print("Temperature is: ");
  Serial.println(t1);
  Serial.print("Humidity is: ");
  Serial.println(h1);

  int ldrValue = digitalRead(ldrPin);
  if (ldrValue==HIGH) {
    Serial.println("Turn on the LED");
    digitalWrite(ledPin, HIGH);
  }
  else{
    digitalWrite(ledPin, LOW);
  }

  if(m1>soilDry){
    Serial.println("Moisture level critical: Needs water");
    delay(1000);
  }
  else if(m1>=soilWet && m1<soilDry ){
    //do nothing, has not been watered yet
    Serial.println("has not been watered yet");
    delay(1000);
  }
  else{
    Serial.println("does not need water");
    delay(1000);
  }


  Firebase.setInt ("Moisture",m1);
  Firebase.setInt ("Humidity",h1);
  Firebase.setInt ("Temperature",t1);

 if(Firebase.failed())
{
  Serial.println("Firebase log sending failed");
  Serial.println(Firebase.error());
  return;
}
  delay(100);
}
```

## ARDUINO IDE RESULT:

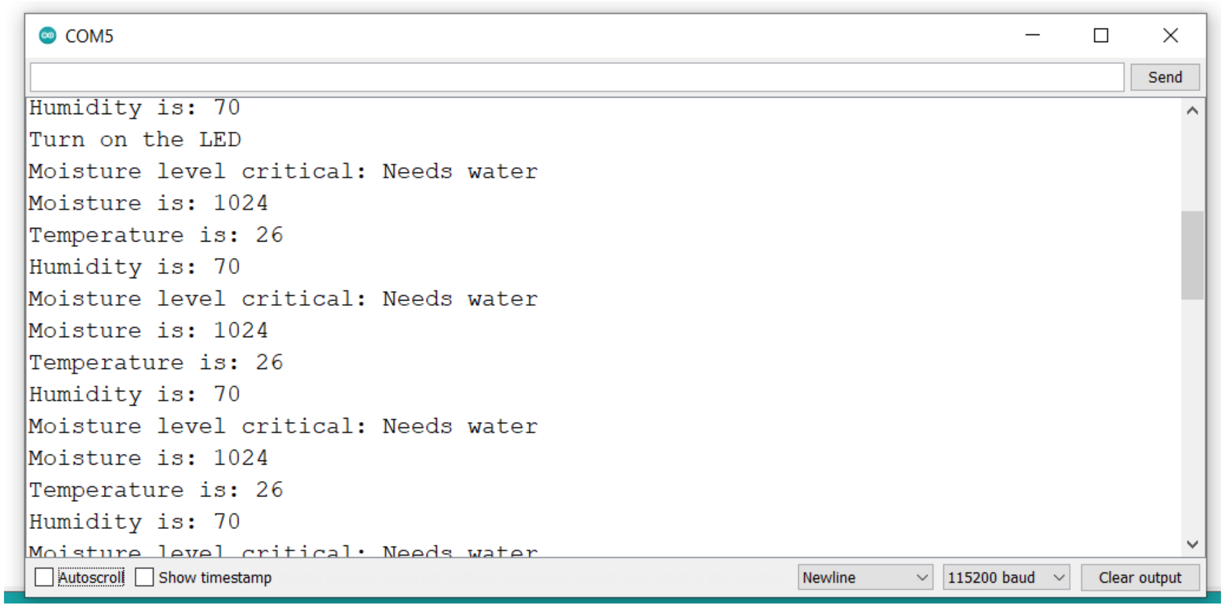


COM5

Send

```
Humidity is: 70
Turn on the LED
Moisture level critical: Needs water
Moisture is: 1024
Temperature is: 26
Humidity is: 70
Moisture level critical: Needs water
Moisture is: 1024
Temperature is: 26
Humidity is: 70
Moisture level critical: Needs water
Moisture is: 1024
Temperature is: 26
Humidity is: 70
Moisture level critical: Needs water
```

Autoscroll  Show timestamp  Newline  115200 baud  Clear output



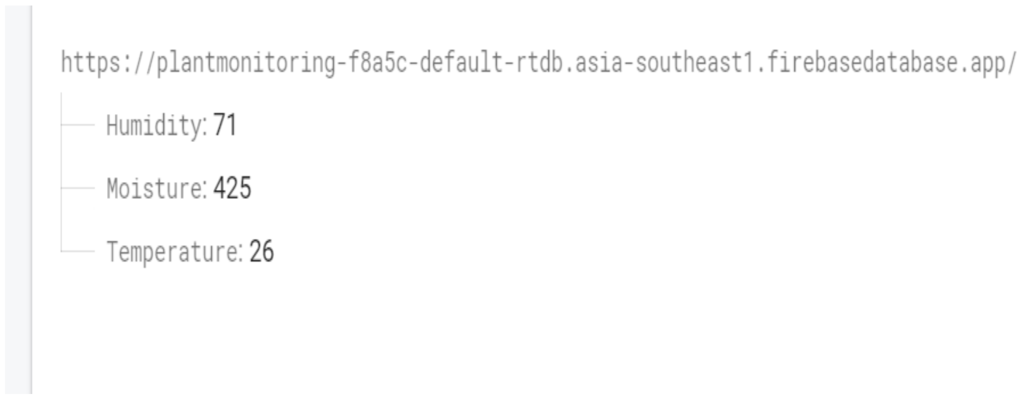
COM5

Send

```
Humidity is: 70
Turn on the LED
Moisture level critical: Needs water
Moisture is: 1024
Temperature is: 26
Humidity is: 70
Moisture level critical: Needs water
Moisture is: 1024
Temperature is: 26
Humidity is: 70
Moisture level critical: Needs water
Moisture is: 1024
Temperature is: 26
Humidity is: 70
Moisture level critical: Needs water
```

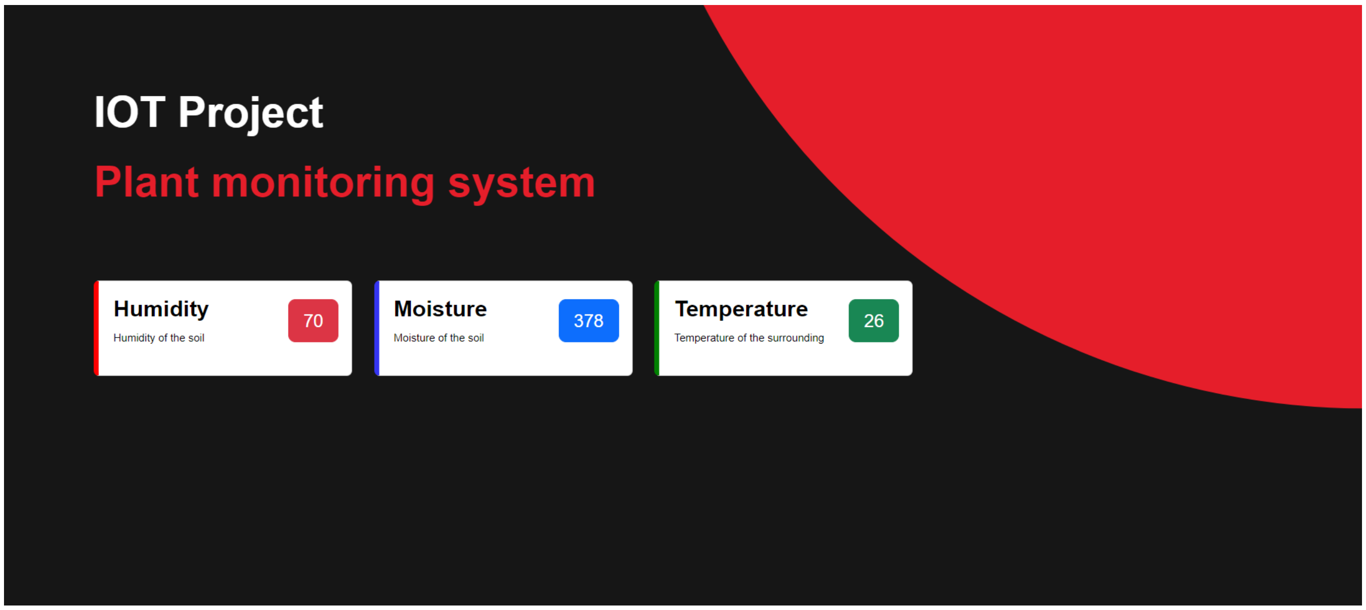
Autoscroll  Show timestamp  Newline  115200 baud  Clear output

## Firebase Realtime database:



```
https://plantmonitoring-f8a5c-default-rtdb.asia-southeast1.firebasedatabase.app/
├── Humidity: 71
├── Moisture: 425
└── Temperature: 26
```

## Web Application:



IOT Project

Plant monitoring system

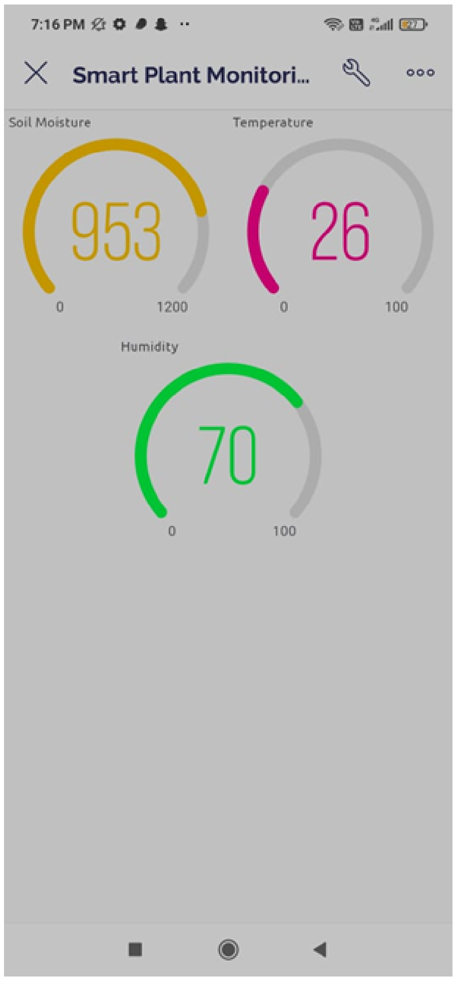
| | | |
|---|---|---|
| Humidity | Humidity of the soil | 70 |
| Moisture | Moisture of the soil | 378 |
| Temperature | Temperature of the surrounding | 26 |

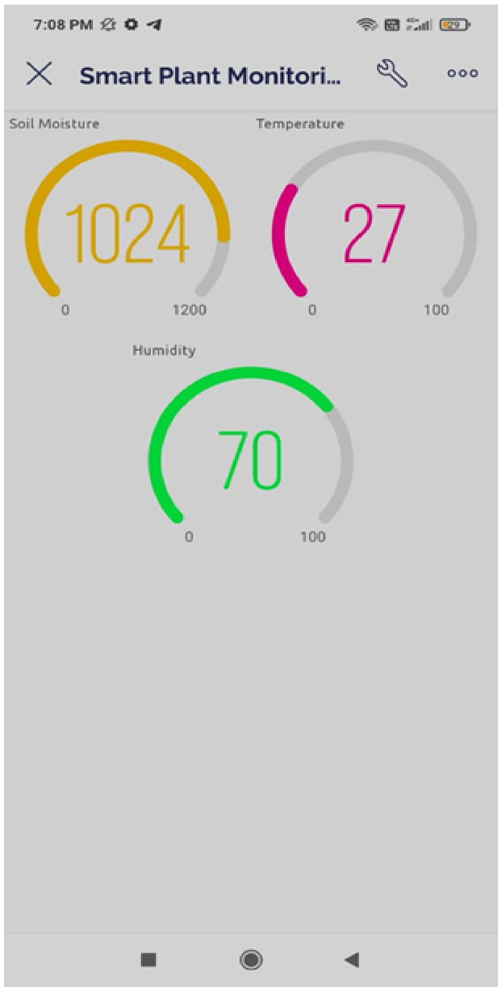
## Code for connecting Firebase and HTML to display the content:

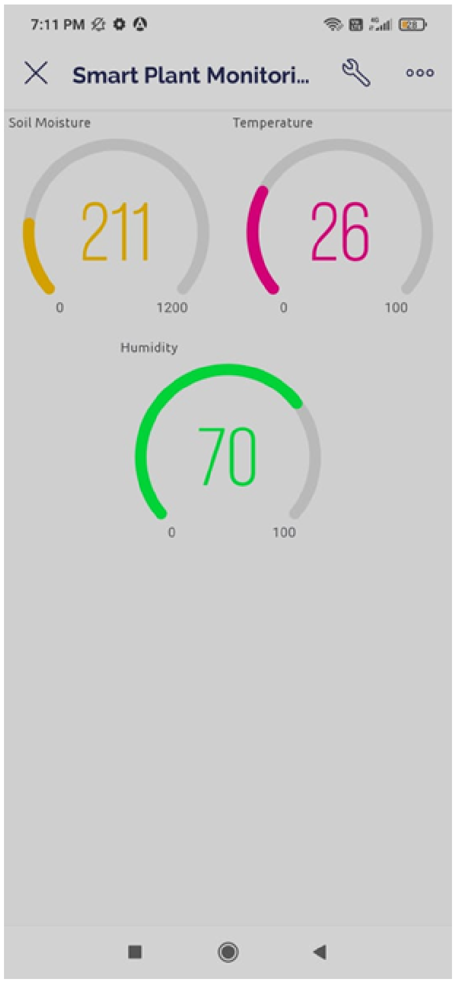
```
<script type="module">
        // Import the functions you need from the SDKs you need
        import { initializeApp } from
"https://www.gstatic.com/firebasejs/9.14.0/firebase-app.js";

        const firebaseConfig = {
          apiKey: "AIzaSyCEgp0CzZIb4gF0fEQAuhxr_m4ysxYNGzg",
          authDomain: "plantmonitoring-f8a5c.firebaseapp.com",
          databaseURL:
"https://plantmonitoring-f8a5c-default-rtdb.asia-southeast1.fireba
sedatabase.app",
          projectId: "plantmonitoring-f8a5c",
          storageBucket: "plantmonitoring-f8a5c.appspot.com",
          messagingSenderId: "936151290472",
          appId: "1:936151290472:web:54d98dd8c26c11609421dd",
          measurementId: "G-WFZL8DPRKR"
        };

        // Initialize Firebase
        const app = initializeApp(firebaseConfig);

        import {getDatabase,ref,get,set,child,update,remove} from
"https://www.gstatic.com/firebasejs/9.14.0/firebase-database.js";
        const db = getDatabase();
            const dbref = ref(db);
            get(child(dbref,"Humidity")).then(snapshot => {
                if(snapshot.exists()){
                    const info = snapshot.val()
                    document.getElementById("humidity").innerHTML
= info;
```

```
                    console.log(info);
                }
                else{
                    console.log("notok");
                }
            })
            get(child(dbref,"Moisture")).then(snapshot => {
                if(snapshot.exists()){
                    const info = snapshot.val()
                    document.getElementById("moisture").innerHTML
= info;
                    console.log(info);
                }
                else{
                    console.log("notok");
                }
            })
            get(child(dbref,"Temperature")).then(snapshot => {
                if(snapshot.exists()){
                    const info = snapshot.val()

document.getElementById("temperature").innerHTML = info;
                    console.log(info);
                }
                else{
                    console.log("notok");
                }
            })

        </script>
```

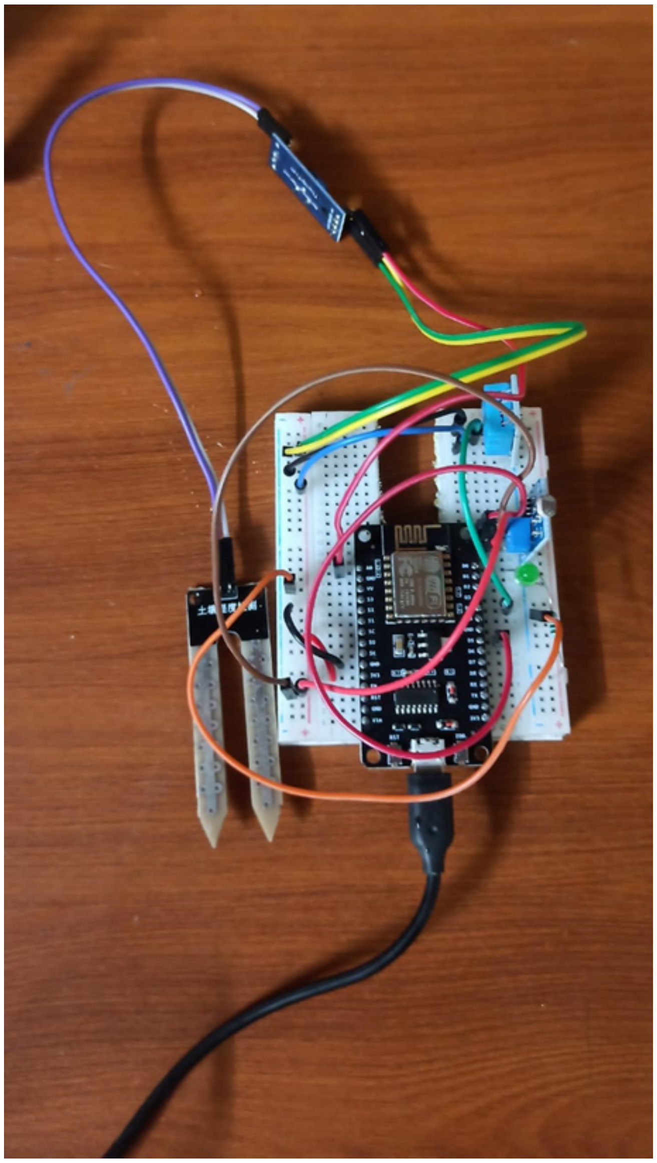
## Blynk mobile application:

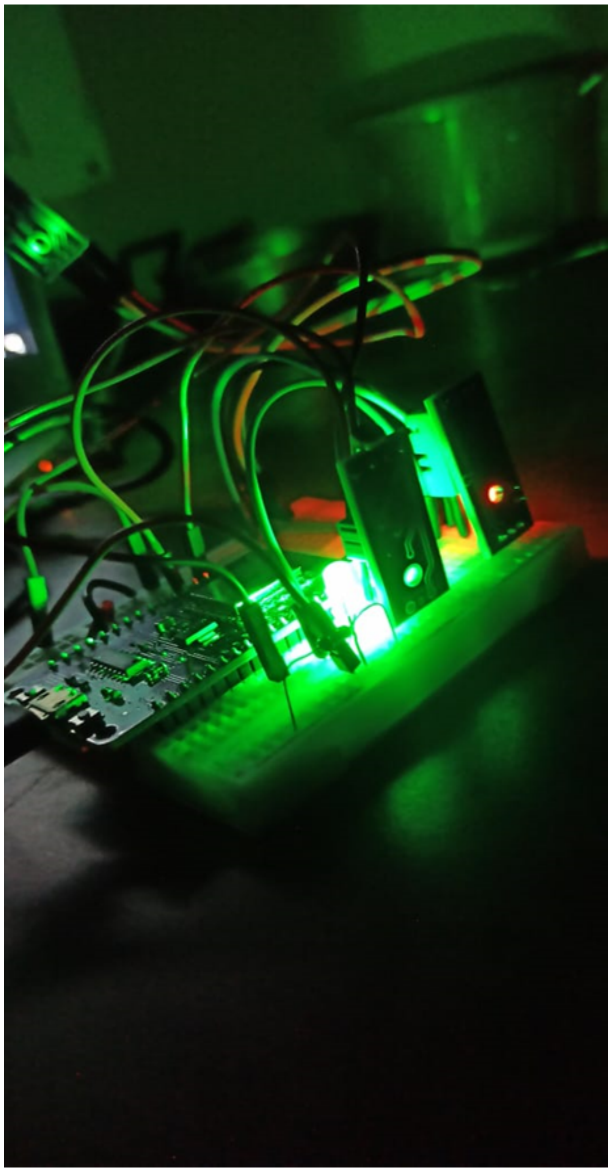






## Hardware model:





## Pros and cons:

**PROS:**

`  Smart Irrigation systems save water, time, and money. Studies show that up to 50% of water usage for landscape irrigation can be saved with IOT-based Smart Plant monitoring systems. As a result, Smart systems typically pay for themselves in water savings within two years. When the weather temperatures increase or the rainfall scale decrease, smart irrigation controller considers the special factor of the place such as soil type and control the watering number and duration

**CONS**

Smart watering systems are a bit expensive. Depending on the size of your property, you will need more systems. Of course saving on water bills will lead to less cost. If you want to use this system for lawn watering, it's better to fix it under the ground before planting.

## Conclusion:

As a result, the "IoT Smart Plant Monitoring, Watering and Security System" has been successfully planned and constructed. It was created by combining the features of all of the hardware components used. Every module's presence has been carefully considered and positioned, resulting in the best possible operation of the unit. The system was thoroughly tested to ensure that it will run on its own.The system automatically comes to a halt when the required moisture level is attained, and then the water pump is switched off. Other than that, if the humidity and temperature levels are changed, it alerts the user through the app. Users have the ability to monitor and control the units from anywhere in the world at any time. Also, they have the opportunity to make a watering schedule according to their preference so that they can reduce the manual human interactions and water the plant automatically by checking the moisture levels of the soil. Users can also manually water the plant simply by using the mobile app which controls the IOT device.

## Future Scope:

According to our project, we implemented our IOT device for a one plant as a prototype. As further work, we can develop our IoT device for home gardens which can check all the plants and the soil. Also, we can develop this IOT device which can be used for greenhouses. Then the green house will be automated, and it will help to get the expected outcome easily or monitor the greenhouse from anywhere in the world. At the industrial level, we can develop our IOT device which can be used for farmers to monitor their farm from their mobile phones.

## References:

[1] Prof. Kawale Jayashri , Sanjay More, Akshay Bankar, Ganesh Dongre, Pooja Patil, “I0T BASED

[2] SMART PLANT MONITORING SYSTEM”, International Journal of Advance Research in Science and Engineering, Vol. No. 7. [2] Prof. Prachi Kamble, “IOT Based Plant Monitoring System”, ITIIRD, Vol. No. 2.

[3] Yogendra Parihar, “Internet of Things and Node MCU”, JETIR, Vol. No. 6. Gabriel, “Monitoring Moisture and Humidity using Arduino Nano”, IJERT, Vol. No. 9

[4] J. G. Jaguey, "Smartphone irrigation sensor," Sensors Journal, vol. 15, 2015.

[5] D. Chaparro, MerceVall-llossera, M. Piles, A. Camps, C. R ̈udiger and R. Riera-Tatch, "Predicting the Extent of Wildfires Using Remotely Sensed Soil Moisture and Temperature Trends," IEEE journal of selected topics in applied earth observations and remote sensing, vol. 9, 2016.

[6] A. Anil, "Project HARITHA - An Automated Irrigation System for Home Gardens," 2012.

[7] J. gutie ́rrez, "Automated irrigation system using a wireless sensor network and gprs," ieee transactions on instrumentation and measurement, vol. 63, 2014

[8] J. Ruan, P. Liao and C. Dong, "The Design and Research on IntelligentFertigation System," in 7th International Conference on Intelligent Human-Machine Systems and Cybernetics, 2015.