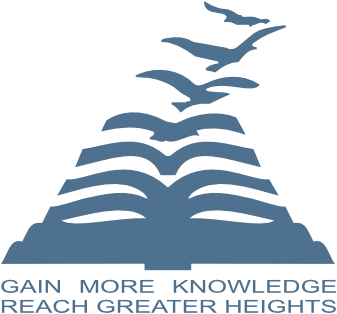
**Presidency University**



A Report on

**“Smart Agri using Arduino”**

A technical project work submitted in partial fulfilment of the requirement for the award of the degree of

**Bachelor of Technology in**

**Computer Science and Engineering**

**By**

**YEDUGURI CHANDRA KIRAN REDDY 20181CSE0804**

**YENDRU SRI HARSHA VARDHAN 20181CSE0806**

**DHARMAVARAM VIVEK SUNDER 20181CSE0837**

**GUDDITI MEGHANA 20181CSE0828**

**YAMUNA D 20181CSE0800**

Under the guidance of

**Mr. Mohammed Mujeer Ullah M. Tech**

Assistant Professor, Dept. of CS&E

**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**

**SCHOOL OF ENGINEERING**

**PRESIDENCY UNIVERSITY**

**2018-2022 (Batch)**

**Table of Contents**

1. **Aim**
2. **Components Used**
3. **Software Used**
4. **Abstract**
5. **Architecture**
6. **Pin Connections**
7. **Blynk Application Interface**
8. **Code**

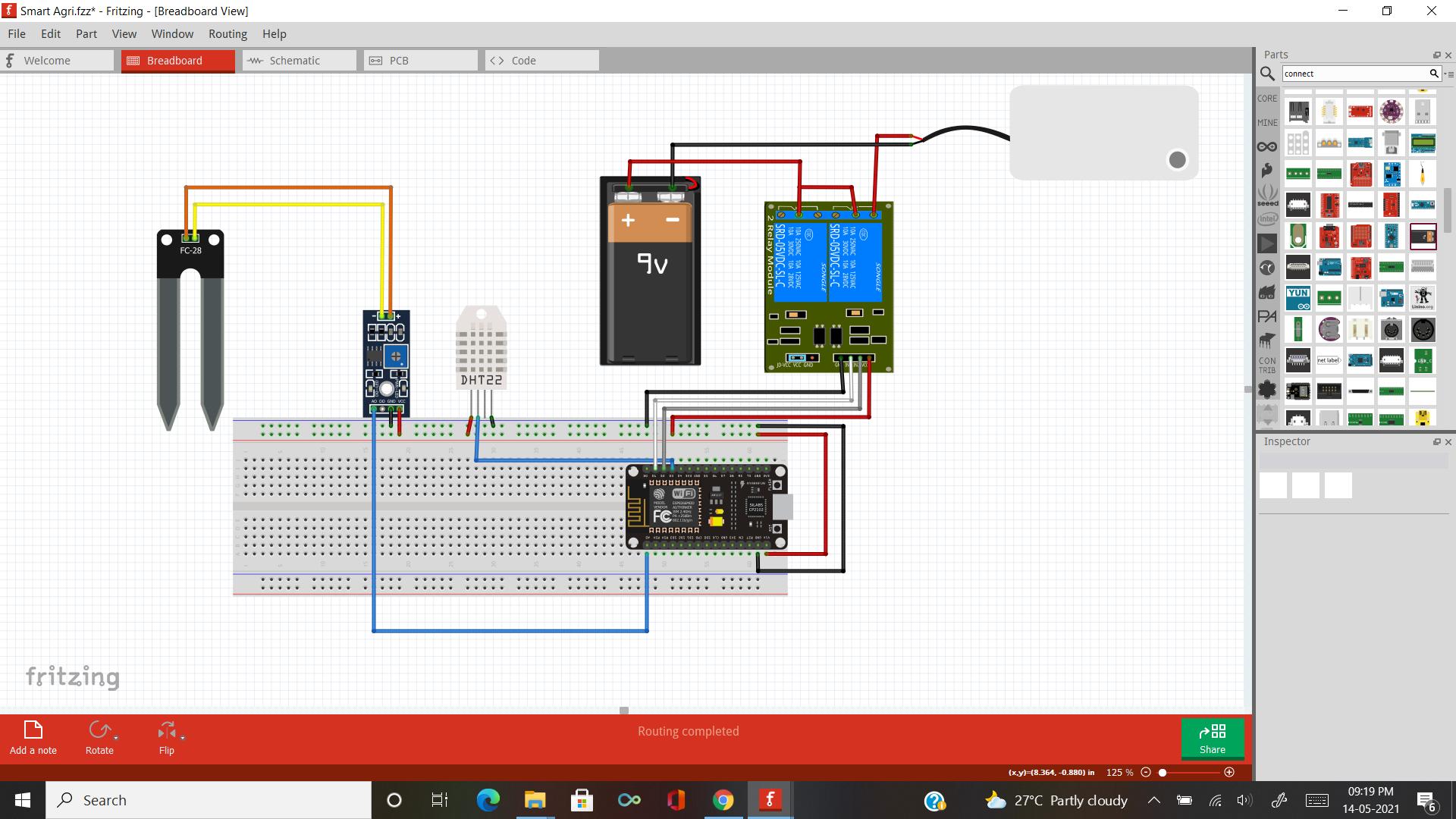
**Aim:** Smart Agri system using Arduino that will be used for farmers to helped farming based on weather conditions and water content in soil.

**Components Used:** Node MCU ESP8266, DHT 11 Temperature and Humidity Sensor, Soil Moisture Sensor, Relay Module, Submersible Pump, jumper cables, Bread Board.

**Software Used:** Arduino IDE, Blynk Application.

**Abstract:** Smart Agri is a system that will help the problem of the watering the fields. Based on the weather conditions and water contents in soil. Smart Agri will display the temperature values, Humidity values, and Soil Moisture values. The Main feature of Smart Agri is it can be controlled by you anywhere in the world. So that you can monitor the crops same as you are physically present in the field.

**Architecture:**



**Pin Connections:**

|  |  |
| --- | --- |
| **Sensor Pins** | **Node MCU board** |
| DHT 11 Ground pin | GND |
| DHT 11 VCC pin | +3.3v |
| DHT 11 Data pin | D3 |
| Soil Moisture Ground pin | GND |
| Soil Moisture VCC pin | +3.3v |
| Soil Moisture Data pin | A0 |
| Relay Module Ground pin | GND |
| Relay Module VCC pin | +5v |
| Relay Module Input pin | D1 |
| Submersible Pump +ve pin | Relay out |
| Submersible Pump -ve pin | Common to power supply |

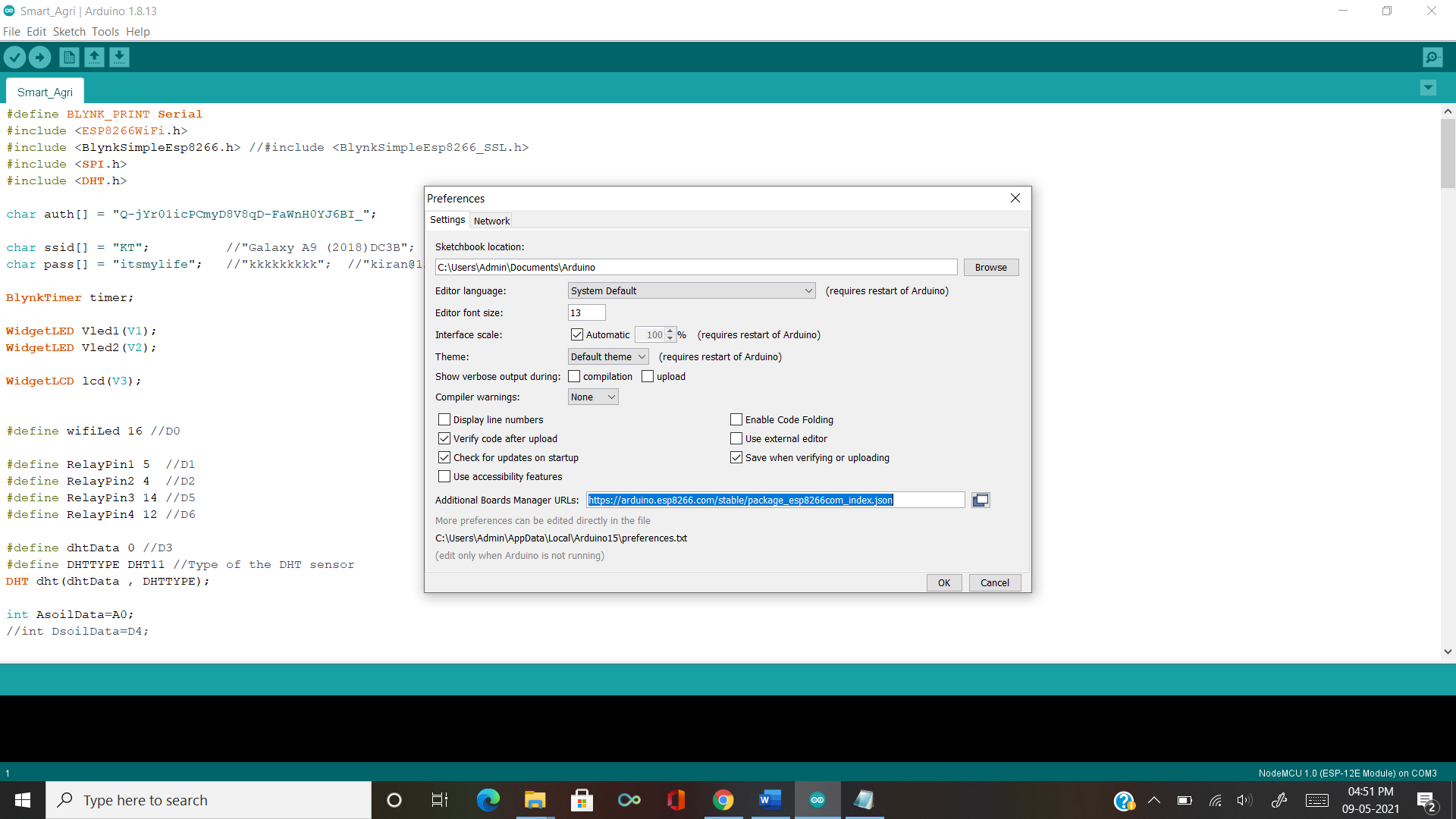
Note: VCC and GND pins are connected to breadboard using RED (VCC) and Black (GND) jumper cables and then to the board.

**INSTRUCTION TO SETUP ARDUINO IDE FOR ESP8266:**

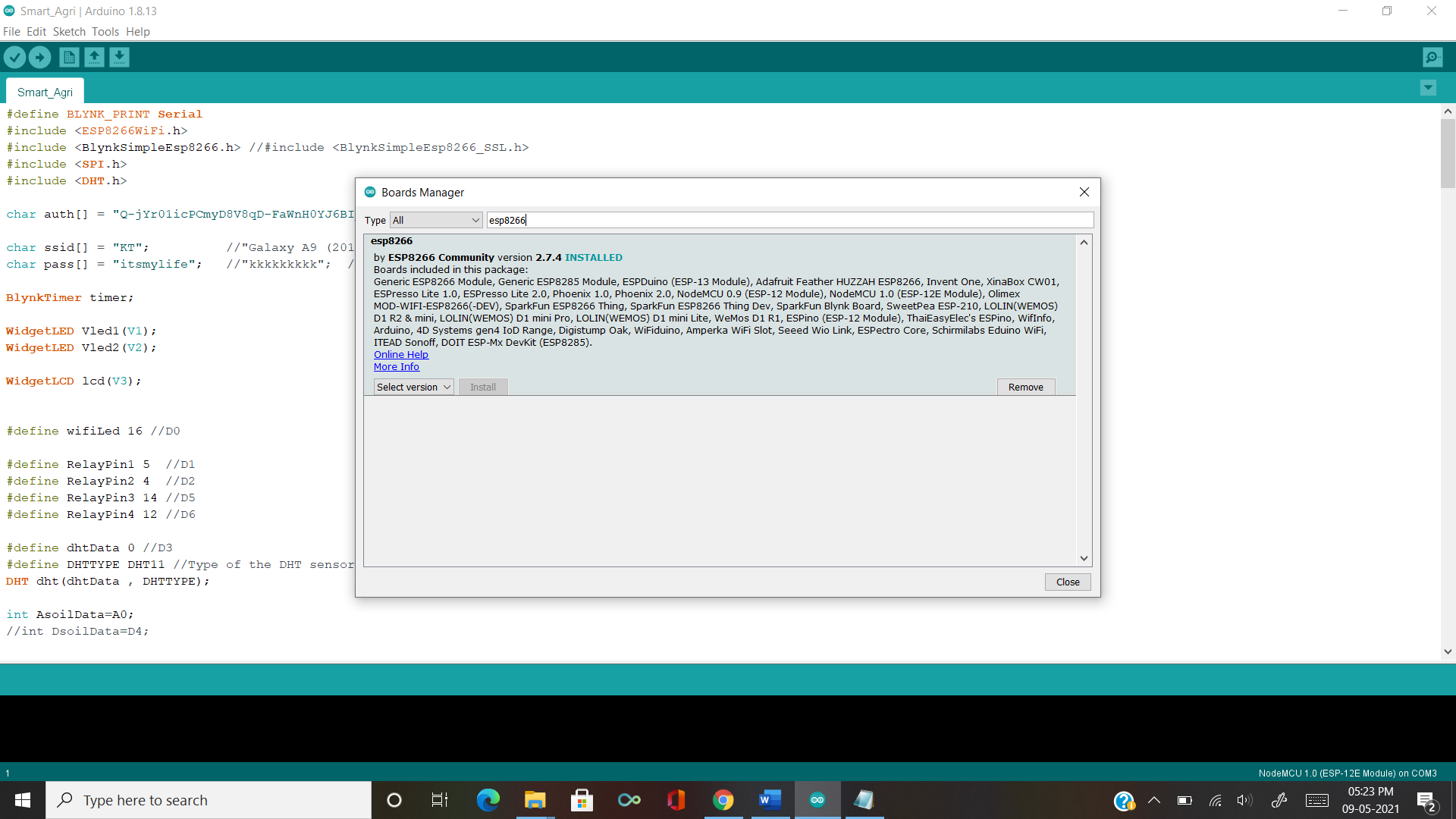
1. We need to Download and Install Arduino IDE from the browser.
2. After Installing open the IDE and Click on “File” located at top left corner and click on Preferences.
3. On Additional Board Manager URLs

https://arduino.esp8266.com/stable/package\_esp8266com\_index.json

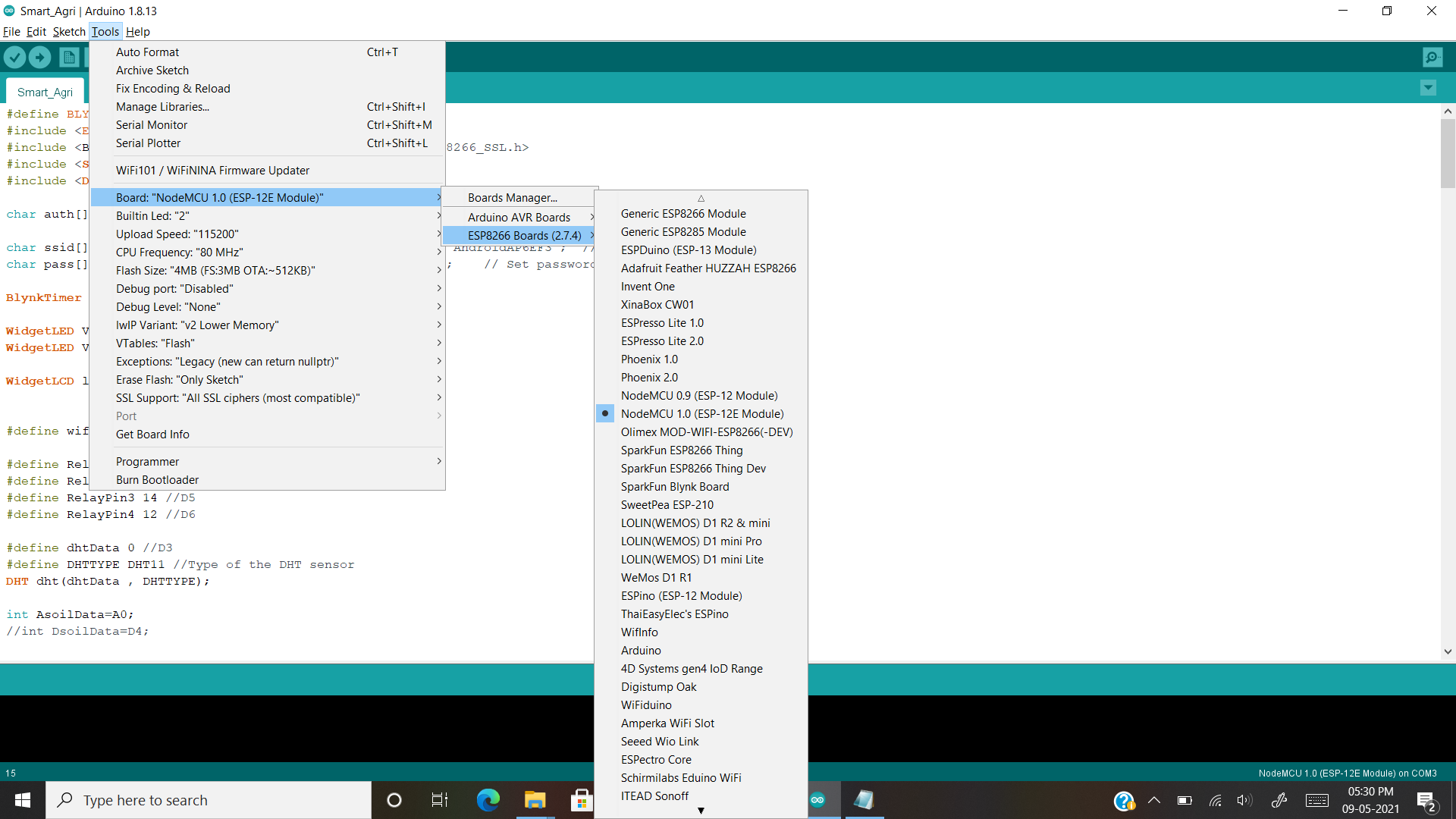
paste this above URL as shown in the below picture and click OK.



1. Now click “Tools” located at top left corner and click Board 🡪 Board Manager and Search for esp8266 and install it as shown in below picture.

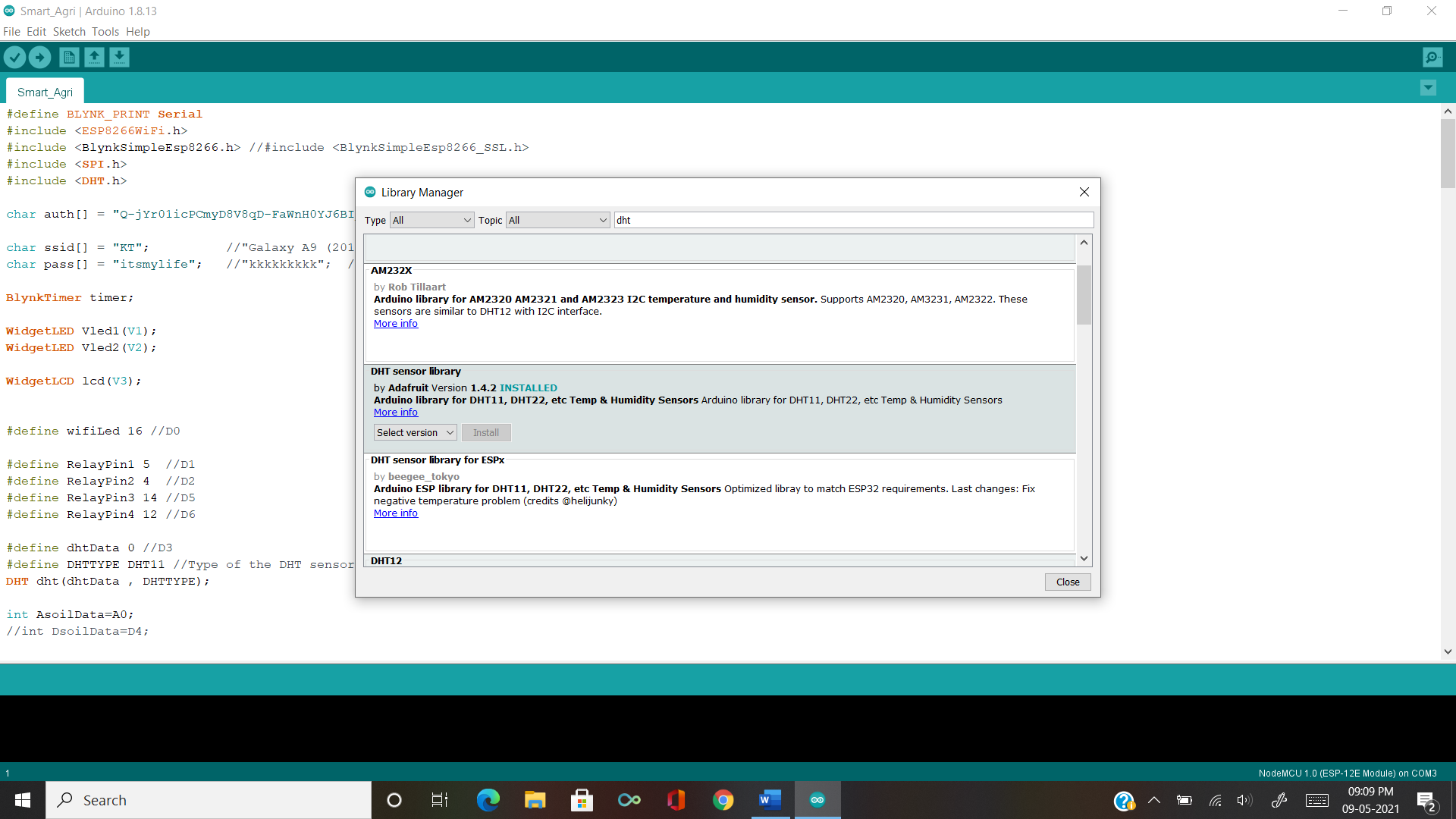


1. Now you can able to see the ESP8266 board as shown in below.



1. Now you need to add install one more library called “DHT sensor Library”

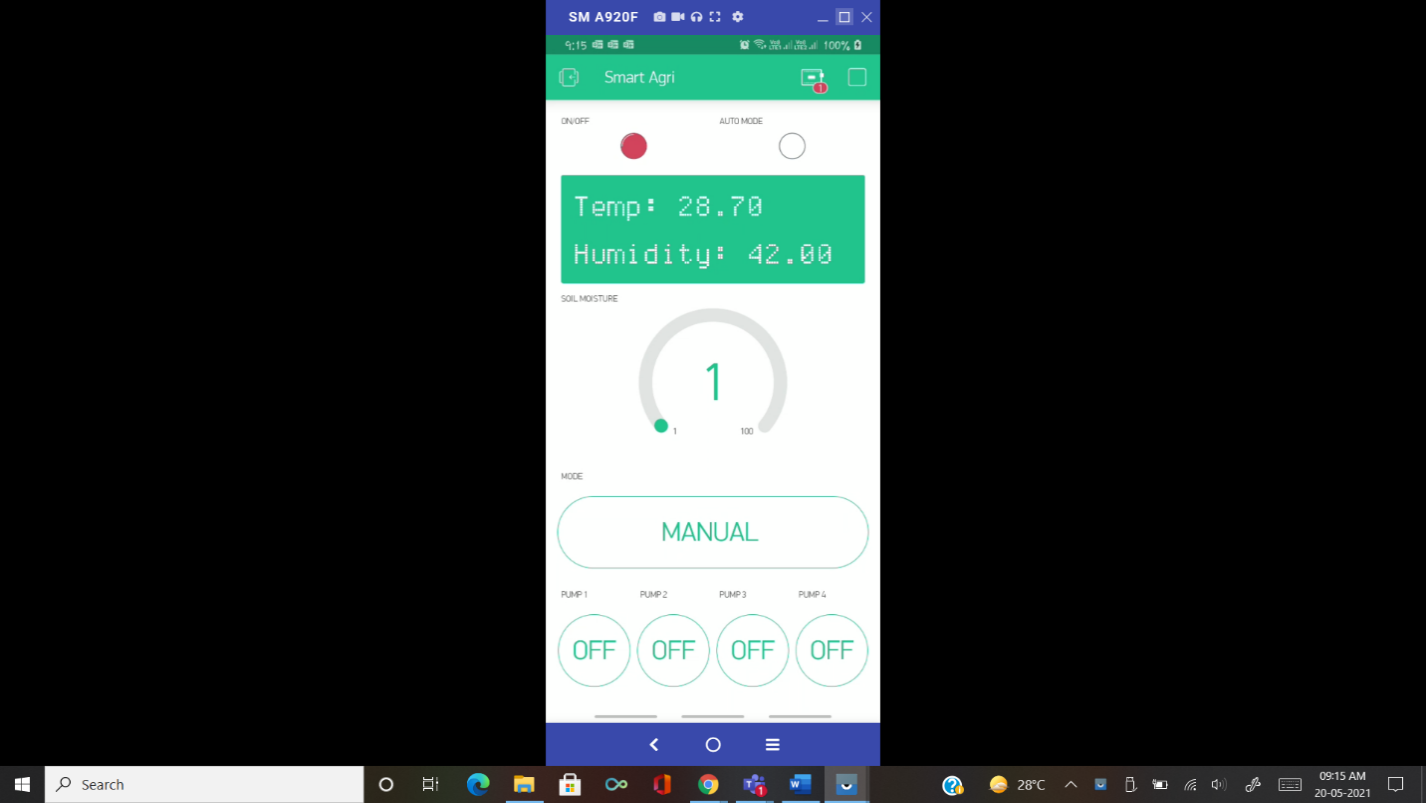
For this you need to click on “Tools” and then click on Manage Libraries there you need to search it and install as shown in below.

****

**Blynk Application Interface:**

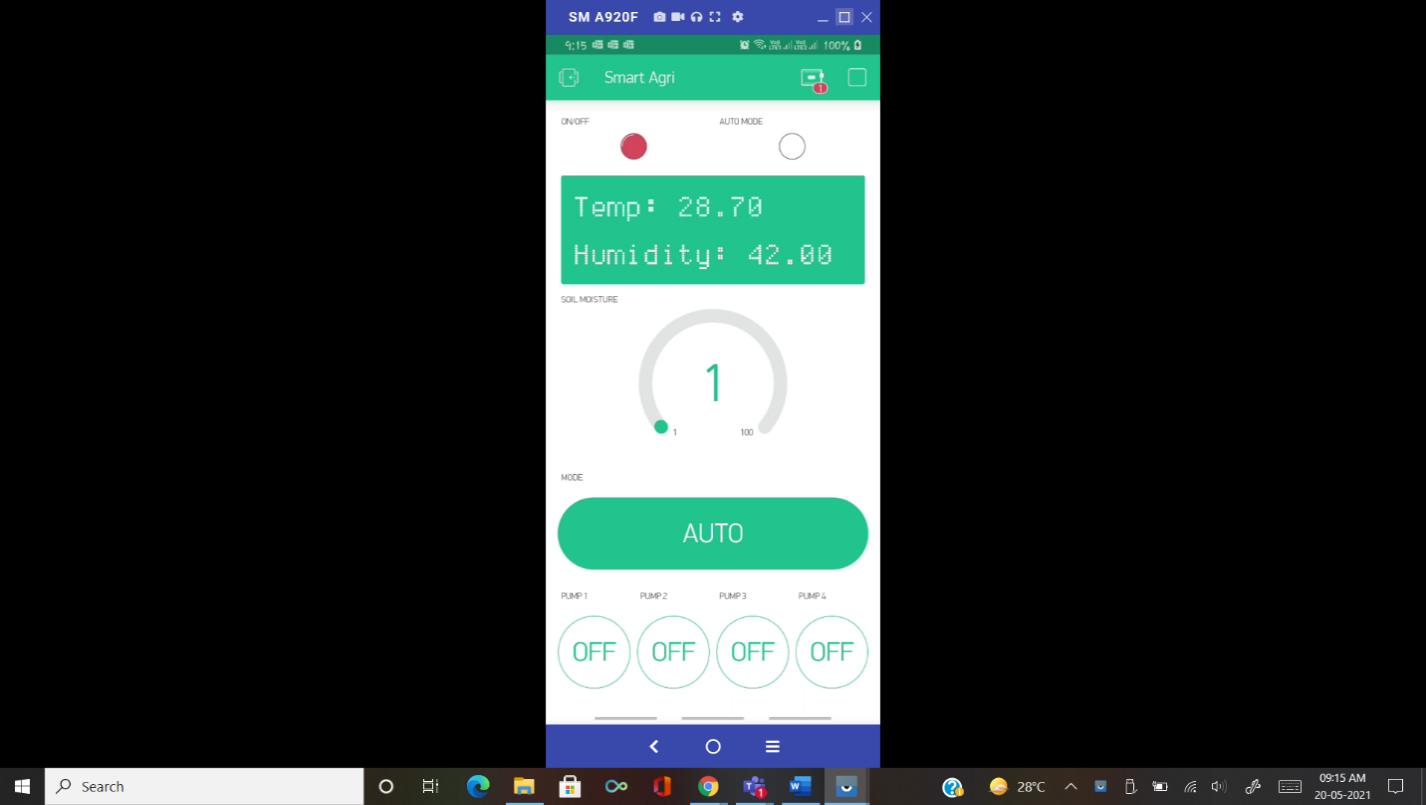
**Manual Mode:**

In manual mode you can control the pumps by youself

****

**Auto Mode:**

In auto mode you no need to control the pumps it will automatically detect the water content in soil and if water level undergoes the given level it will Turn On the pump.

****

**ARDUINO IDE CODE:**

#define BLYNK\_PRINT Serial

#include <ESP8266WiFi.h>

#include <BlynkSimpleEsp8266.h>

#include <SPI.h>

#include <DHT.h>

char auth[] = " "; //Your Authentication token

char ssid[] = "xxxx"; // Your WiFi credentials.

char pass[] = "xxxx"; // Set password to "" for open networks.

BlynkTimer timer;

WidgetLED Vled1(V1);

WidgetLED Vled2(V2);

WidgetLCD lcd(V3);

#define wifiLed 16 //D0

#define RelayPin1 5 //D1

#define RelayPin2 4 //D2

#define RelayPin3 14 //D5

#define RelayPin4 12 //D6

#define dhtData 0 //D3

#define DHTTYPE DHT11 //Type of the DHT sensor

DHT dht(dhtData , DHTTYPE);

int AsoilData=A0;

#define VPIN\_BUTTON\_1 V6 //Relay1

#define VPIN\_BUTTON\_2 V7 //Relay2

#define VPIN\_BUTTON\_3 V8 //Relay3

#define VPIN\_BUTTON\_4 V9 //Relay4

#define VPIN\_BUTTON\_5 V5 // Auto or Manual modes

int toggleState\_1=1;

int toggleState\_2=1;

int toggleState\_3=1;

int toggleState\_4=1;

int modeState;

int analog;

int analogPercent;

int waterLimit=35;

int wifiFlag=0;

BLYNK\_WRITE(VPIN\_BUTTON\_1) {

toggleState\_1 = param.asInt();

digitalWrite(RelayPin1, toggleState\_1);

}

BLYNK\_WRITE(VPIN\_BUTTON\_2) {

toggleState\_2 = param.asInt();

digitalWrite(RelayPin2, toggleState\_2);

}

BLYNK\_WRITE(VPIN\_BUTTON\_3) {

toggleState\_3 = param.asInt();

digitalWrite(RelayPin3, toggleState\_3);

}

BLYNK\_WRITE(VPIN\_BUTTON\_4) {

toggleState\_4 = param.asInt();

digitalWrite(RelayPin4, toggleState\_4);

}

BLYNK\_WRITE(VPIN\_BUTTON\_5) {

modeState = param.asInt();

}

BLYNK\_CONNECTED() {

// Request the latest state from the server

Blynk.syncVirtual(VPIN\_BUTTON\_1);

Blynk.syncVirtual(VPIN\_BUTTON\_2);

Blynk.syncVirtual(VPIN\_BUTTON\_3);

Blynk.syncVirtual(VPIN\_BUTTON\_4);

Blynk.syncVirtual(VPIN\_BUTTON\_5);

}

void checkBlynkStatus() { // called every 3 seconds by SimpleTimer

bool isconnected = Blynk.connected();

if (isconnected == false) {

wifiFlag = 1;

digitalWrite(wifiLed, HIGH);//Turn off WiFi LED

}

if (isconnected == true) {

wifiFlag = 0;

digitalWrite(wifiLed, LOW); //Turn on WiFi LED

}

}

void dhtSensor()

{

float h = dht.readHumidity();

float t = dht.readTemperature(); // or dht.readTemperature(true) for Fahrenheit

if (isnan(h) || isnan(t)) {

Serial.println("Failed to read from DHT sensor!");

return;

}

Serial.println(t);

Serial.println(h);

String text1= "Temp: " + String(t);

String text2= "Humidity: " +String(h);

lcd.clear();

lcd.print(0, 0, text1);

lcd.print(0, 1, text2);

}

void soilSensor()

{

analog = analogRead(AsoilData);

analogPercent = map(analog,0,1024,100,0);

//analogPercent = map(analog,0,1024,1024,0);

Blynk.virtualWrite(V4, analogPercent);

if(modeState==1 && analogPercent<waterLimit)

{

int toggle\_1=0;

int toggle\_2=0;

int toggle\_3=0;

int toggle\_4=0;

Blynk.virtualWrite(VPIN\_BUTTON\_1, toggle\_1);

digitalWrite(RelayPin1, toggle\_1);

Blynk.virtualWrite(VPIN\_BUTTON\_2, toggle\_2);

digitalWrite(RelayPin2, toggle\_2);

Blynk.virtualWrite(VPIN\_BUTTON\_3, toggle\_3);

digitalWrite(RelayPin3, toggle\_3);

Blynk.virtualWrite(VPIN\_BUTTON\_4, toggle\_4);

digitalWrite(RelayPin4, toggle\_4);

}

else if(modeState==1 && analogPercent>waterLimit)

{

int toggle\_1=1;

int toggle\_2=1;

int toggle\_3=1;

int toggle\_4=1;

Blynk.virtualWrite(VPIN\_BUTTON\_1, toggle\_1);

digitalWrite(RelayPin1, toggle\_1);

Blynk.virtualWrite(VPIN\_BUTTON\_2, toggle\_2);

digitalWrite(RelayPin2, toggle\_2);

Blynk.virtualWrite(VPIN\_BUTTON\_3, toggle\_3);

digitalWrite(RelayPin3, toggle\_3);

Blynk.virtualWrite(VPIN\_BUTTON\_4, toggle\_4);

digitalWrite(RelayPin4, toggle\_4);

}

else

{

Blynk.virtualWrite(VPIN\_BUTTON\_1, toggleState\_1);

digitalWrite(RelayPin1, toggleState\_1);

Blynk.virtualWrite(VPIN\_BUTTON\_2, toggleState\_2);

digitalWrite(RelayPin2, toggleState\_2);

Blynk.virtualWrite(VPIN\_BUTTON\_3, toggleState\_3);

digitalWrite(RelayPin3, toggleState\_3);

Blynk.virtualWrite(VPIN\_BUTTON\_4, toggleState\_4);

digitalWrite(RelayPin4, toggleState\_4);

}

}

void welcomeMSG()

{

lcd.clear();

lcd.print(4,0 ,"Wellcome");

delay(3000);

lcd.clear();

lcd.print(7,0, "TO");

delay(3000);

lcd.print(3,1, "Smart Agri");

delay(4000);

lcd.clear();

}

void setup()

{

Serial.begin(9600);

dht.begin();

Blynk.begin(auth, ssid, pass);

pinMode(RelayPin1, OUTPUT);

pinMode(RelayPin2, OUTPUT);

pinMode(RelayPin3, OUTPUT);

pinMode(RelayPin4, OUTPUT);

pinMode(wifiLed, OUTPUT);

//During Starting all Relays should TURN OFF

digitalWrite(RelayPin1, toggleState\_1);

digitalWrite(RelayPin2, toggleState\_2);

digitalWrite(RelayPin3, toggleState\_3);

digitalWrite(RelayPin4, toggleState\_4);

Vled1.off();

Vled2.off();

lcd.clear();

welcomeMSG();

timer.setInterval(3000L, checkBlynkStatus); // check if Blynk server is connected every 3 seconds

timer.setInterval(100L, dhtSensor);

timer.setInterval(100L, soilSensor);

}

void loop()

{

if(modeState==1)//Auto or manual control

{

Vled2.on();

}

else

{

Vled2.off();

}

if (WiFi.status() != WL\_CONNECTED)

{

Serial.println("WiFi Not Connected");

Vled1.off();

}

else

{

Serial.println("WiFi Connected");

Blynk.run();

Vled1.on();

}

timer.run();

}