// Viral Science www.viralsciencecreativity.com www.youtube.com/c/viralscience

// Blynk IOT Smart Plant Monitoring System

/\* Connections

Relay. D3

Btn. D7

Soil. A0

PIR. D5

SDA. D2

SCL. D1

Temp. D4

\*/

//Include the library files

#include <LiquidCrystal\_I2C.h>

#define BLYNK\_PRINT Serial

#include <ESP8266WiFi.h>

#include <BlynkSimpleEsp8266.h>

#include <DHT.h>

//Initialize the LCD display

LiquidCrystal\_I2C lcd(0x3F, 16, 2);

char auth[] = " "; //Enter your Blynk Auth token

char ssid[] = " "; //Enter your WIFI SSID

char pass[] = " "; //Enter your WIFI Password

DHT dht(D4, DHT11);//(DHT sensor pin,sensor type) D4 DHT11 Temperature Sensor

BlynkTimer timer;

//Define component pins

#define soil A0 //A0 Soil Moisture Sensor

#define PIR D5 //D5 PIR Motion Sensor

int PIR\_ToggleValue;

void checkPhysicalButton();

int relay1State = LOW;

int pushButton1State = HIGH;

#define RELAY\_PIN\_1 D3 //D3 Relay

#define PUSH\_BUTTON\_1 D7 //D7 Button

#define VPIN\_BUTTON\_1 V12

//Create three variables for pressure

double T, P;

char status;

void setup() {

Serial.begin(9600);

lcd.init();

lcd.backlight();

pinMode(PIR, INPUT);

pinMode(RELAY\_PIN\_1, OUTPUT);

digitalWrite(RELAY\_PIN\_1, LOW);

pinMode(PUSH\_BUTTON\_1, INPUT\_PULLUP);

digitalWrite(RELAY\_PIN\_1, relay1State);

Blynk.begin(auth, ssid, pass, "blynk.cloud", 80);

dht.begin();

lcd.setCursor(0, 0);

lcd.print(" Initializing ");

for (int a = 5; a <= 10; a++) {

lcd.setCursor(a, 1);

lcd.print(".");

delay(500);

}

lcd.clear();

lcd.setCursor(11, 1);

lcd.print("W:OFF");

//Call the function

timer.setInterval(100L, soilMoistureSensor);

timer.setInterval(100L, DHT11sensor);

timer.setInterval(500L, checkPhysicalButton);

}

//Get the DHT11 sensor values

void DHT11sensor() {

float h = dht.readHumidity();

float t = dht.readTemperature();

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

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

return;

}

Blynk.virtualWrite(V0, t);

Blynk.virtualWrite(V1, h);

lcd.setCursor(0, 0);

lcd.print("T:");

lcd.print(t);

lcd.setCursor(8, 0);

lcd.print("H:");

lcd.print(h);

}

//Get the soil moisture values

void soilMoistureSensor() {

int value = analogRead(soil);

value = map(value, 0, 1024, 0, 100);

value = (value - 100) \* -1;

Blynk.virtualWrite(V3, value);

lcd.setCursor(0, 1);

lcd.print("S:");

lcd.print(value);

lcd.print(" ");

}

//Get the PIR sensor values

void PIRsensor() {

bool value = digitalRead(PIR);

if (value) {

Blynk.logEvent("pirmotion","WARNNG! Motion Detected!"); //Enter your Event Name

WidgetLED LED(V5);

LED.on();

} else {

WidgetLED LED(V5);

LED.off();

}

}

BLYNK\_WRITE(V6)

{

PIR\_ToggleValue = param.asInt();

}

BLYNK\_CONNECTED() {

// Request the latest state from the server

Blynk.syncVirtual(VPIN\_BUTTON\_1);

}

BLYNK\_WRITE(VPIN\_BUTTON\_1) {

relay1State = param.asInt();

digitalWrite(RELAY\_PIN\_1, relay1State);

}

void checkPhysicalButton()

{

if (digitalRead(PUSH\_BUTTON\_1) == LOW) {

// pushButton1State is used to avoid sequential toggles

if (pushButton1State != LOW) {

// Toggle Relay state

relay1State = !relay1State;

digitalWrite(RELAY\_PIN\_1, relay1State);

// Update Button Widget

Blynk.virtualWrite(VPIN\_BUTTON\_1, relay1State);

}

pushButton1State = LOW;

} else {

pushButton1State = HIGH;

}

}

void loop() {

if (PIR\_ToggleValue == 1)

{

lcd.setCursor(5, 1);

lcd.print("M:ON ");

PIRsensor();

}

else

{

lcd.setCursor(5, 1);

lcd.print("M:OFF");

WidgetLED LED(V5);

LED.off();

}

if (relay1State == HIGH)

{

lcd.setCursor(11, 1);

lcd.print("W:ON ");

}

else if (relay1State == LOW)

{

lcd.setCursor(11, 1);

lcd.print("W:OFF");

}

Blynk.run();//Run the Blynk library

timer.run();//Run the Blynk timer

}