

SMART IRRIGATION SYSTEM USING BLYNK APP MINI PROJECT CODE

```
#define BLYNK_TEMPLATE_ID          "TMPL6Ys20MNQH"
#define BLYNK_TEMPLATE_NAME        "miniproject"
#define BLYNK_AUTH_TOKEN           "G_rTS0iGb6mCS5iMQz5h2IQpnYDdn19p"

/* Comment this out to disable prints and save space */
#define BLYNK_PRINT Serial

#include <LiquidCrystal_I2C.h>
#include <WiFi.h>
#include <WiFiClient.h>
#include <BlynkSimpleEsp32.h>
#include <DHT.h>
LiquidCrystal_I2C lcd(0x27, 16, 2);

// Your WiFi credentials.
// Set password to "" for open networks.
char ssid[] = "azizi";
char pass[] = "012345678";

#define DHTPIN 4                  // What digital pin we're connected to
#define sensor 33
#define waterPump 12

// Uncomment whatever type you're using!
#define DHTTYPE DHT11            // DHT 11
// #define DHTTYPE DHT22          // DHT 22, AM2302, AM2321
// #define DHTTYPE DHT21          // DHT 21, AM2301

DHT dht(DHTPIN, DHTTYPE);
BlynkTimer timer;

bool Relay = 0;

// This function sends Arduino's up time every second to Virtual Pin
// (5).
// In the app, Widget's reading frequency should be set to PUSH. This
// means
// that you define how often to send data to Blynk App.
void sendSensor()
{
    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;
    }
    // You can send any value at any time.
    // Please don't send more that 10 values per second.
    Blynk.virtualWrite(V5, h);
    Blynk.virtualWrite(V6, t);
}

void setup()
{
    // Debug console
    Serial.begin(9600);
    pinMode(waterPump, OUTPUT);
    digitalWrite(waterPump, HIGH);
    lcd.begin();
    lcd.backlight();
    dht.begin();

    Blynk.begin(BLYNK_AUTH_TOKEN, ssid, pass);
    // You can also specify server:
    //Blynk.begin(BLYNK_AUTH_TOKEN, ssid, pass, "blynk.cloud", 80);
    //Blynk.begin(BLYNK_AUTH_TOKEN, ssid, pass, IPAddress(192,168,1,100),
    8080);

    dht.begin();

    lcd.setCursor(1, 0);
    lcd.print("System Loading");
    for (int a = 0; a <= 15; a++) {
        lcd.setCursor(a, 1);
        lcd.print(".");
        delay(500);
    }
    lcd.clear();

    timer.setInterval(100L, DHT11sensor);
    timer.setInterval(100L, soilMoisture);

    // Setup a function to be called every second
    timer.setInterval(1000L, sendSensor);
}

BLYNK_WRITE(V7) {
    Relay = param.asInt();
}

```

```

    if (Relay == 1) {
        digitalWrite(waterPump, LOW);

    } else {
        digitalWrite(waterPump, HIGH);
    }
}

//Get the soil moisture values
void soilMoisture() {
    int value = analogRead(sensor);
    value =(100- ((value/4095)*100));
    Blynk.virtualWrite(V8, value);
    Serial.println(value);
    lcd.setCursor(0, 0);
    lcd.print("Moisture :");
    lcd.print(value);
    lcd.print(" ");
}

void DHT11sensor() {
    float h = dht.readHumidity();
    float t = dht.readTemperature();

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

    lcd.setCursor(0, 1);
    lcd.print("T:");
    lcd.print(t);

    lcd.setCursor(8, 1);
    lcd.print("H:");
    lcd.print(h);
}

void loop()
{
    Blynk.run();
    timer.run();
}

```

