#include <ESP8266WiFi.h>

#include <ThingSpeak.h>

#include <Wire.h>

#include <LiquidCrystal\_I2C.h>

#define MOISTURE\_PIN A0

#define RELAY\_PIN D5

#define MOISTURE\_THRESHOLD 600

const char\* ssid = "iPhone";

const char\* password = "12345678";

WiFiClient client;

unsigned long myChannelNumber = 2970757;

const char\* myWriteAPIKey = "LLDWWEKGVHPDOW3P";

LiquidCrystal\_I2C lcd(0x27, 16, 2);

void setup() {

  Serial.begin(9600);

  WiFi.begin(ssid, password);

  lcd.init();  // Corrected from lcd.begin() to lcd.init()

  lcd.backlight();

  lcd.setCursor(0, 0);

  lcd.print("Connecting WiFi");

  while (WiFi.status() != WL\_CONNECTED) {

    delay(500);

    Serial.print(".");

  }

  Serial.println("WiFi connected");

  lcd.clear();

  lcd.print("WiFi Connected");

  ThingSpeak.begin(client);

  pinMode(RELAY\_PIN, OUTPUT);

  digitalWrite(RELAY\_PIN, HIGH); // Pump OFF

  delay(2000);

  lcd.clear();

}

void loop() {

  int moisture = analogRead(MOISTURE\_PIN);

  bool pumpStatus;

  lcd.setCursor(0, 0);

  lcd.print("Moisture: ");

  lcd.print(moisture);

  lcd.print("   ");

  if (moisture > MOISTURE\_THRESHOLD) {

    digitalWrite(RELAY\_PIN, LOW);  // Pump ON

    pumpStatus = 1;

    lcd.setCursor(0, 1);

    lcd.print("Pump: ON         ");

  } else {

    digitalWrite(RELAY\_PIN, HIGH); // Pump OFF

    pumpStatus = 0;

    lcd.setCursor(0, 1);

    lcd.print("Pump: OFF        ");

  }

  // Send data to ThingSpeak

  ThingSpeak.setField(1, moisture);

  ThingSpeak.setField(2, pumpStatus);

  int x = ThingSpeak.writeFields(myChannelNumber, myWriteAPIKey);

  if (x == 200) {

    Serial.println("Channel update successful.");

  } else {

    Serial.print("Problem updating channel. HTTP error code ");

    Serial.println(x);

  }

  delay(15000); // ThingSpeak update interval

}