REAL TIME WATER QUALITY MONITORING AND ANALYSIS SYSTEM

TEAM MEMBERS:

JOVITHA S

MAADHINI K

REAL TIME WATER QUALITY MONITORING AND ANALYSIS SYSTEM

AIM:

To design and continuously monitor, collect and analysis data on various parameter of the water quality using sensors.

COMPONENTS REQUIRED:

1. ESP 8266(NODE-MCU)
2. pH sensor
3. Temperature sensor (DSI8B20)
4. Turbidity sensor(SEN0189)
5. TDS(Total Dissolved Solids)
6. EC(Electrical Conductivity sensor)
7. DO(Dissolved Oxygen sensor)

FUNCTION:

1. ESP8266

Low-cost Wi-Fi enabled system on a chip (SoC). The **NodeMCU** is a development board that includes the ESP8266 chip, USB support, and makes programming easier.

Uses:

- Smart home devices (lights, fans, switches)
- Weather stations (with temperature/humidity sensors)

Range: 2.4 GHz, 3.3V

2. pH Sensor

Scientific instrument designed to measure the acidity or alkalinity of a liquid.

Uses:

- Water quality monitoring
- Food and beverage industry

Range: 0–14

3. Temperature Sensor

Device that detects and measures the temperature of an object or environment, and converts it into an electrical signal. Example: DS18B20 – digital temperature sensor.

Uses:

- Electronics & devices
- Automobiles

Range: -55 °C to 125 °C

4. Turbidity Sensor

Measures the cloudiness or haziness of a liquid, which indicates the concentration of suspended particles (e.g., SEN0189 – gravity analog turbidity sensor).

Uses:

- Water quality monitoring
- Environmental studies

Range: 0–1000 NTU

5. TDS Sensor

Measures total dissolved solids such as salts, minerals, and metals in water.

Uses:

- Industrial applications
- Agriculture & hydroponics

Range: 0–1000 ppm

6. EC Sensor

Electrical conductivity sensor that measures the ability of a solution (commonly water) to conduct electricity.

Uses:

- Agriculture
- Aquaculture

Range: 0–20 mS/cm

7. DO Sensor (Dissolved Oxygen)

Device used to measure the concentration of molecular oxygen dissolved in water.

Uses:

- Water quality monitoring
- Wastewater treatment

Range: 0–20 mg/L

PIN CONFIGURATION:

S.NO	COMPONENTS	ESP8266 PIN
1.	OLED VCC	3.3V
2.	OLED GND	GND
3.	OLED SDA	D2(GP104)
4.	OLED SCL	D1(GP105)
5.	DS18B20 VDD	3.3V
6.	DS18B20 GND	GND
7.	DS18D20 DQ	D4(GP102)
8.	LED anode	D5(GP1014)
9.	LED cathode	GND

PROCEDURE:

Identify Parameters → pH, temperature, turbidity, DO, conductivity, etc. (based on use case).

Select Sensors & Hardware → water quality probes + microcontroller (ESP32/RPi) + power (battery/solar) + enclosure.

Connect & Calibrate Sensors → signal conditioning, temperature compensation, calibration with standard solutions.

Acquire Data → microcontroller reads sensors periodically, applies calibration & filtering.

Transmit Data → send via Wi-Fi/LoRa/GSM to cloud or local server.

Store & Process → save in time-series DB; apply thresholds, rules, or WQI index.

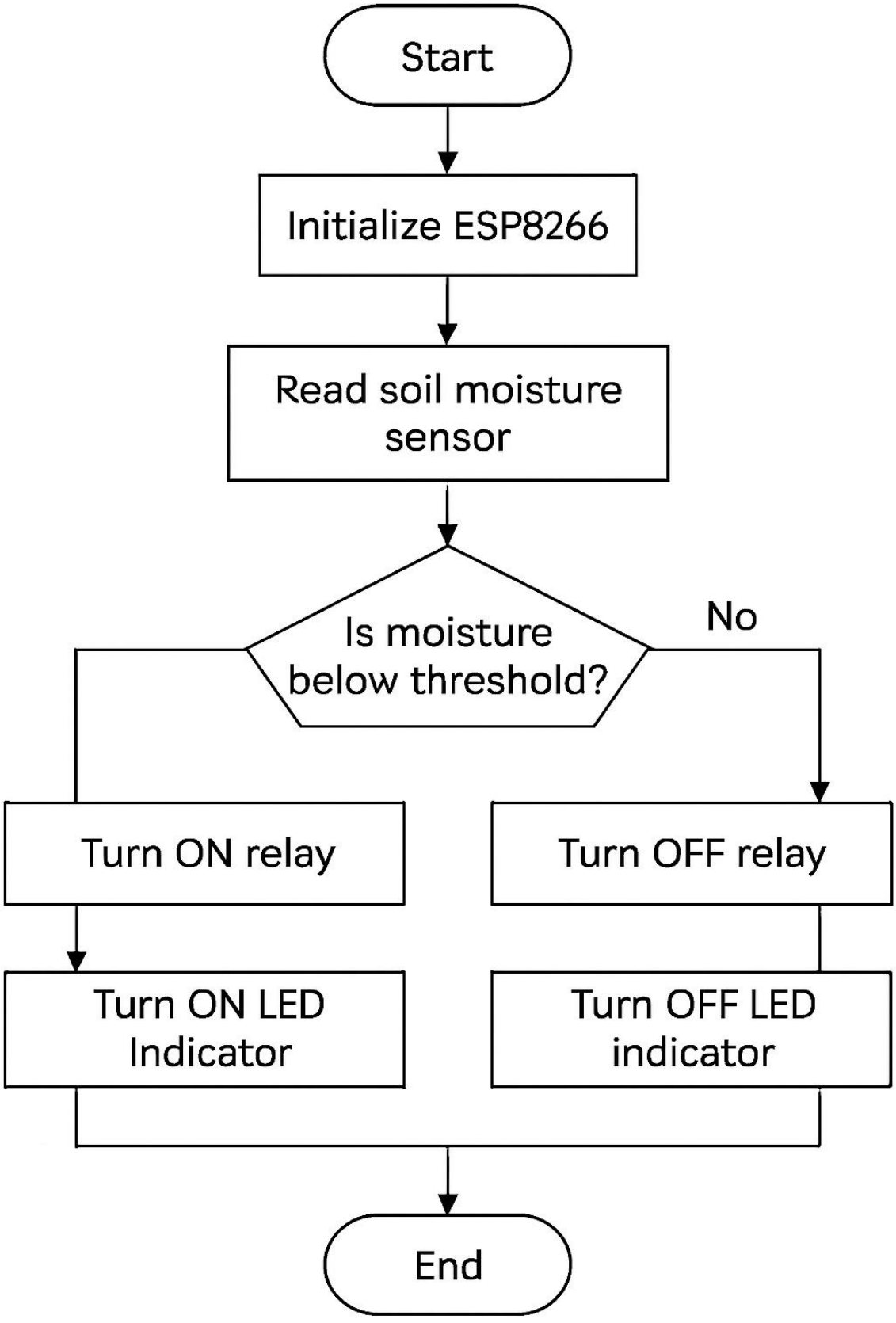
Visualize & Analyze → dashboard (Grafana/Web app) with real-time graphs & alerts.

Alerts → SMS/Email/Push notification when quality crosses safe limits.

Maintenance → regular sensor cleaning, calibration, and probe replacement.

Validation & Deployment → field testing, compare with reference devices, install at site.

FLOW CHART:


Start
Initialize ESP8266
Read soil moisture sensor
Is moisture below threshold?
No
Turn ON relay
Turn OFF relay
Turn ON LED Indicator
Turn OFF LED indicator
End

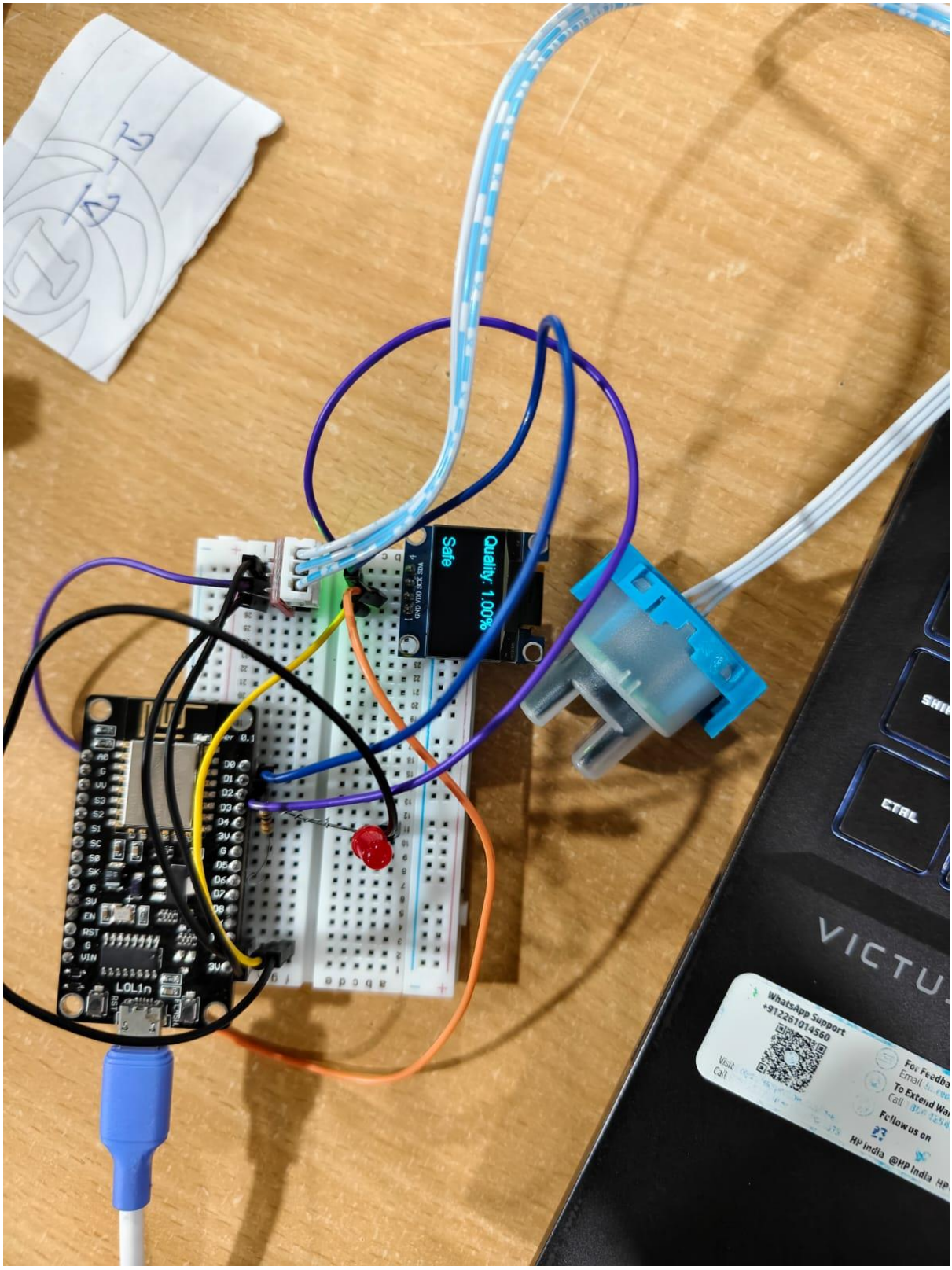
PROGRAM CODE:

```
// Pin definitions using GPIO numbers
#define RELAY_PIN 5    // GPIO5 (D1) -> Relay IN1
#define SCL_PIN   14   // GPIO14 (D5) -> OLED SCL
#define SDA_PIN   12   // GPIO12 (D6) -> OLED SDA
#define SOIL_PIN  A0   // Analog pin (A0) -> Soil sensor

// Extra LEDs
#define LED1_PIN  4    // GPIO4 (D2)
#define LED2_PIN  0    // GPIO0 (D3)
#define LED3_PIN  2    // GPIO2 (D4)

void setup() {
  Serial.begin(115200);

  // Relay setup
  pinMode(RELAY_PIN, OUTPUT);
  digitalWrite(RELAY_PIN, HIGH); // Relay OFF initially (active LOW)

  // I2C pins (just declared, not used without OLED library)
  pinMode(SCL_PIN, OUTPUT);
  pinMode(SDA_PIN, OUTPUT);

  // LED pins
  pinMode(LED1_PIN, OUTPUT);
  pinMode(LED2_PIN, OUTPUT);
```

```
  pinMode(LED3_PIN, OUTPUT);

  // Start with all LEDs OFF
  digitalWrite(LED1_PIN, LOW);
  digitalWrite(LED2_PIN, LOW);
  digitalWrite(LED3_PIN, LOW);

  Serial.println("System Ready");
}

void loop() {
  int soilValue = analogRead(SOIL_PIN);

  // Convert to percentage (0=wet, 1023=dry)
  int moisturePercent = map(soilValue, 1023, 0, 0, 100);

  // Print values
  Serial.print("Soil Value: ");
  Serial.print(soilValue);
  Serial.print(" | Moisture: ");
  Serial.print(moisturePercent);
  Serial.println("%");

  // Relay control: pump ON if soil moisture < 40%
  if (moisturePercent < 40) {
    digitalWrite(RELAY_PIN, LOW);  // Relay ON
```

```
  Serial.println("Pump: ON");
 } else {
  digitalWrite(RELAY_PIN, HIGH); // Relay OFF
  Serial.println("Pump: OFF");
 }

 // LED indicators (example usage)
 if (moisturePercent < 30) {
  digitalWrite(LED1_PIN, HIGH); // Very dry
  digitalWrite(LED2_PIN, LOW);
  digitalWrite(LED3_PIN, LOW);
 } else if (moisturePercent < 60) {
  digitalWrite(LED1_PIN, LOW);
  digitalWrite(LED2_PIN, HIGH); // Moderate
  digitalWrite(LED3_PIN, LOW);
 } else {
  digitalWrite(LED1_PIN, LOW);
  digitalWrite(LED2_PIN, LOW);
  digitalWrite(LED3_PIN, HIGH); // Wet
 }

 delay(2000);
}
```

CIRCUIT DIAGRAM:



EXECUTION STEPS:

This program initializes the ESP8266's pins for relay, LEDs, and soil moisture sensing, prints "System Ready" once set up, and then enters a continuous loop where it reads the soil moisture sensor value, calculates moisture percentage, and displays these readings via serial output. Based on this moisture level, the relay switches the pump ON if soil moisture drops below 40%, helping maintain adequate watering. Three LEDs visually indicate the current soil condition: LED1 signals very dry soil (moisture < 30%), LED2 signals moderately moist soil (30–59%), and LED3 indicates wet soil (60% and above).The loop repeats every 2 seconds to continuously monitor and control the system

RESULT:

The system reads soil moisture, displays values, and controls the pump—activating it when dryness is detected below 40%, with LEDs showing moisture status Automated watering and real-time feedback optimize irrigation and make monitoring simple