

# Temperature & Humidity Monitoring System with Alert

## 1. Components Used

- Arduino Uno (or compatible board)
- DHT11 Temperature & Humidity Sensor
- 0.96" OLED Display (SSD1306, I2C)
- RGB LED (Common Cathode or Anode)
- Active Buzzer
- Passive Buzzer
- Resistors (220Ω for RGB LED)
- Jumper wires & breadboard / PCB

## 2. Functionality

This project monitors ambient temperature and humidity using the DHT11 sensor. The OLED screen displays live readings, while the RGB LED provides a visual status: - **Blue:** Normal (< 30°C) - **Green:** Warning (30–35°C) - **Red:** Danger (> 35°C) In warning/danger states, an active buzzer beeps at different intervals. In danger mode (> 35°C), a passive buzzer plays an emergency melody and the OLED shows **TEMP RISING ALERT!!**

## 3. Pin Mapping

Component	Arduino Pin
DHT11 Sensor	D2
OLED Display (SDA/SCL)	A4 (SDA), A5 (SCL)
RGB LED - Red	D9
RGB LED - Green	D10
RGB LED - Blue	D11
Active Buzzer	D6
Passive Buzzer	D7

## 4. Wiring Steps

- 1 Connect DHT11 VCC → 5V, GND → GND, Data → D2 (with 10k pull-up resistor).
- 2 Connect OLED VCC → 3.3V or 5V, GND → GND, SDA → A4, SCL → A5.
- 3 Connect RGB LED pins: Red → D9, Green → D10, Blue → D11 (with resistors).
- 4 Connect Active Buzzer positive → D6, negative → GND.
- 5 Connect Passive Buzzer positive → D7, negative → GND.
- 6 Ensure proper ground connections for all components.

## 5. Helpful Tips

- If the OLED does not display, check the I2C address (default 0x3C).

- Use resistors with RGB LED to prevent burning out.
- Ensure DHT11 has a pull-up resistor on the data pin for stable readings.
- Use separate power if many components cause voltage drops.
- Check buzzer polarity before wiring.

## **6. Troubleshooting**

- OLED blank → Verify I2C wiring and address.
- DHT11 shows NaN → Loose connection or missing pull-up resistor.
- Buzzers silent → Check pin mapping and wiring polarity.
- LED not lighting correctly → Verify resistor values and common cathode/anode type.
- Code uploads but nothing works → Ensure correct board & COM port in Arduino IDE.