Environment Setup Instructions for Weather Station Project

Ravishan BBN

January 23, 2025

1 Introduction

This document provides a step-by-step guide to setting up the Weather Station project. The project integrates sensors, Raspberry Pi, Node-RED, and ThingSpeak to monitor environmental parameters such as temperature, pressure, CO levels, and hazardous gas levels. Follow these instructions carefully to configure your development environment.

2 Hardware Requirements

- **Raspberry Pi 4 B**: Running the latest Raspbian OS.
- **ADS1115 ADC Module**: For reading analog sensors (e.g., MQ7, MQ135).
- **BMP280 Sensor**: For temperature and pressure readings.
- **MQ7 and MQ135 Sensors**: For detecting CO and hazardous gases.
- **Breadboard and Jumper Wires**: For prototyping connections.
- **Internet Connection**: For uploading data to ThingSpeak.

3 Software Requirements

- **Python 3.7+**: Installed on Raspberry Pi.
- **Node-RED**: For creating a local dashboard.
- **ThingSpeak Account**: For cloud-based monitoring.
- **Mosquitto MQTT Broker**: For communication between Raspberry Pi and Node-RED.

4 Python Dependencies

4.1 Installation Commands

Run the following commands on the Raspberry Pi:

```
sudo apt update
sudo apt install python3-pip
pip3 install paho-mqtt requests adafruit-circuitpython-bmp280 \
    adafruit-circuitpython-ads1x15
```

5 Node-RED Configuration

- **Install Node-RED**:

  ```
  sudo apt install nodejs npm
  sudo npm install -g --unsafe-perm node-red
  ```

- **Install Necessary Nodes**:

  ```
  npm install node-red-dashboard node-red-contrib-mqtt
  ```

- **Start Node-RED**:

  ```
  node-red
  ```

6 ThingSpeak Integration

6.1 Setup Instructions

- Register at `https://thingspeak.com/`.
- Create a new channel and configure fields for:
  - Temperature
  - Pressure
  - CO Level
  - Hazardous Gas Level
- Obtain the API Write Key for the channel.

6.2 Python Integration

Ensure your Python script uploads data to ThingSpeak using:

```
url = "https://api.thingspeak.com/update"
payload = {
    "api_key": "<YOUR_API_KEY>",
    "field1": temperature,
    "field2": pressure,
    "field3": co_level,
    "field4": hazardous_level
}
requests.post(url, data=payload)
```

7 MQTT Broker Configuration

- Install Mosquitto:

```
sudo apt install mosquitto mosquitto-clients
```

- Start Mosquitto:

```
sudo systemctl start mosquitto
```

- Test MQTT:

```
mosquitto_sub -h localhost -t "sensor_data"
```

8 Dashboard Setup

8.1 Node-RED Dashboard

Create a dashboard with:

- Gauges for displaying temperature, pressure, CO levels, and hazardous levels.
- Conditional coloring for hazardous level classification (e.g., Good, Moderate, Bad).

8.2 Mobile Integration

Use services like Pushbullet to receive mobile alerts. Configure ThingSpeak React to trigger alerts:

- Set a React condition for high thresholds (e.g., hazardous level > 100).
- Configure a ThingHTTP request to send notifications using Pushbullet.

9 Verification

- Test Python scripts for sensor readings:

```
python3 read_bmp280.py
```

- Verify Node-RED dashboard displays live data.
- Check ThingSpeak for uploaded sensor data.
- Confirm mobile alerts are triggered on high thresholds.

10 Conclusion

With the setup complete, the Weather Station is ready for real-time monitoring and cloud integration. Make sure to regularly update dependencies and monitor system logs for errors.