AIR QUALITY MONITORING(AQM)

Subbmitted by: k. Reddy Anusha

Reg no:723921104021

E-Mail:anushareddyk227@gmail.com

To configure IoT devices to measure air quality parameters and

develop a Python script to send the collected data to a data-sharing

platform, you'll need the following components:

IoT Device (e.g., Raspberry Pi, Arduino, ESP8266/ESP32, etc.) Air quality sensors (e.g., particulate matter, gas sensors) Python (Python 3.x) installed on the IoT device An Internet connection (Wi-Fi, Ethernet, etc.) A data-sharing platform or server (e.g., MQTT broker, HTTP server, or a cloud platform like AWS IoT, Google Cloud IoT, or Azure IoT) Here's a step-by-step guide to achieve this:

Set Up Hardware:

Connect the air quality sensors to the IoT device as per the manufacturer's instructions. Make sure your IoT device is connected to the internet. Install Required Libraries: Depending on the sensors you're using, you might need to install specific Python libraries. For instance, if you're using the BME280 sensor for temperature, humidity, and pressure measurements, you can install the Adafruit Circuit Python library:

bash

Copy code

pip install adafruit-circuitpython-bme280

Create a Python Script:

Create a Python script on the IoT device to read data from the sensors and send it to your data-sharing platform. Below is a basic example using MQTT as the data-sharing protocol. You'll need to modify it according to your specific sensor and platform:

Python

Copy code

import time

import board

import bus io

import adafruit\_bme280

import pah.mqtt.client as mqtt

# Sensor setup

i2c = busio.I2C(board.SCL, board.SDA)

sensor = adafruit\_bme280.Adafruit\_BME280\_I2C(i2c)

# MQTT configuration

mqtt\_broker = "your\_mqtt\_broker"

mqtt\_port = 1883

mqtt\_topic = "air\_quality\_data"

mqtt\_client\_id = "air\_quality\_sensor"

mqtt\_username = "your\_username"

mqtt\_password = "your\_password"

def on\_connect(client, userdata, flags, rc):

print("Connected to MQTT broker with result code " + str(rc)) client = mqtt.Client(client\_id=mqtt\_client\_id)

client.on\_connect = on\_connect client.username\_pw\_set(username=mqtt\_username, password=mqtt\_password)

client.connect(mqtt\_broker, mqtt\_port, 60)

while True:

temperature = sensor.temperature

humidity = sensor.humidity

pressure = sensor.pressure

# Send data to MQTT broker

payload = f"Temperature: {temperature}°C, Humidity: {humidity}%, Pressure: {pressure} hPa"

client.publish(mqtt\_topic, payload)

time.sleep(60) # Send data every 60 seconds

client.loop\_forever()

Replace the placeholders:

Replace your\_mqtt\_broker, mqtt\_topic, mqtt\_username, and mqtt\_password with your MQTT broker information.

Adjust the sensor setup to your specific sensor.

Run the Script:

Run the Python script on your IoT device.

This is a basic example, and the actual code will depend on your specific sensors and data-sharing platform. Make sure to secure your IoT device and consider error handling and data format as per your platform's requirements. Additionally, you may need to handle sleep and error conditions more robustly for a production environment.

Was this response better or worse?

Better

Worse

Same