

Air Quality Monitoring

Hardware Selection:

Choose suitable sensors for measuring air quality parameters, such as PM2.5, PM10, CO2, temperature, and humidity. Select an IoT development board like Raspberry Pi, Arduino, or specialized IoT hardware.

Hardware Setup:

Connect the selected sensors to the IoT device following their datasheets and pinouts. Make sure the IoT device has an internet connection method, such as Wi-Fi or Ethernet.

Install Required Libraries:

Install necessary Python libraries for sensor data collection. For example, if you're using a Raspberry Pi, you might use libraries like Adafruit_DHT, smbus, or pandas.

Collect Sensor Data:

Write Python code to read data from the connected sensors. Ensure that the data is accurate and calibrated.

Here's an example code snippet for collecting data from a DHT22 temperature and humidity sensor on a Raspberry Pi:

```
import Adafruit_DHT

sensor = Adafruit_DHT.DHT22
pin = 4

humidity, temperature = Adafruit_DHT.read_retry(sensor, pin)

if humidity is not None and temperature is not None:
    print(f'Temperature: {temperature:.2f}°C, Humidity: {humidity:.2f}%')
else:
    print('Failed to retrieve data from the sensor.')
```

Data Processing:

Process the collected sensor data if necessary. You might need to filter, aggregate, or format the data.

Data Sharing Platform:

Choose a data-sharing platform where you want to send the data. Options include cloud platforms like AWS, Google Cloud, Azure, or dedicated IoT platforms like ThingSpeak or Ubidots.

Here's a Python script to read data from an SDS011 sensor using the pms5003 library, which is a common library for working with these sensors:

```
import time

import serial

# Set the serial port for your sensor (may vary depending on your setup)
ser = serial.Serial('/dev/ttyUSB0', baudrate=9600, timeout=2)

def read_sensor_data(ser):
    while True:
        data = ser.read(10)

        if data[0] == 170 and data[1] == 192:
            pm25 = int.from_bytes(data[2:4], byteorder='little') / 10.0
            pm10 = int.from_bytes(data[4:6], byteorder='little') / 10.0
            print(f'PM2.5: {pm25} µg/m³, PM10: {pm10} µg/m³')

    try:
        read_sensor_data(ser)
    except KeyboardInterrupt:
        ser.close()
```