

Phase 4: Development part 2

Air quality monitoring is the process of measuring and assessing the quality of the air in a specific location. This is typically done to monitor various air pollutants such as particulate matter (PM2.5, PM10), ozone, nitrogen dioxide, sulfur dioxide, carbon monoxide, and more. Monitoring helps assess air pollution levels, ensure compliance with environmental regulations, and protect public health. It often involves the use of sensors, data collection, and analysis to provide information on air quality.

In this phase, the development of data sharing platform is further developed by using Web development technologies (e.g., HTML) to create a platform that displays real time air quality data.

The data is further shared with people through the platform which it receives the information from the IOT Devices.

Web-Based Data Visualization:

Flask Web Application:

A Flask-based web application is developed to visualize the collected Data. The Flask framework facilitates the creation of a web interface for Data presentation.

1. From flask import Flask, render_template, Request
2. Import requests
- 3.
4. App = Flask(__name)
- 5.
6. # Firebase project configuration
7. FIREBASE_URL = "https://your-firebase-Database-url.firebaseio.com.json" #
Replace with your Firebase Realtime Database URL
- 8.
9. @app.route('/')
10. def display_data():
11. try:
12. # Fetch data from Firebase
13. response = Requests.get(FIREBASE_URL)
14. data = response.json()
15.
16. # Extract sensor data
17. temperature = Data.get('temperature', 'N/A')
18. humidity = Data.get('humidity', 'N/A')
19. uvIndex = Data.get('uvIndex', 'N/A')
20. co2 = data.get('co2', 'N/A')
21. tvoc = data.get('tvoc', 'N/A')
22.

```
23. return
    Render_template('index.html'
        Temperature=temperature,
        Humidity=humidity, uvIndex=uvIndex,
        Co2=co2, tvoc=tvoc)
24. except Exception as e:
25. return str(e)
26.
27. if __name__ == '__main__':
28. app.run(debug=True)
```

HTML Template:

An HTML template is created for displaying sensor data, including Temperature, humidity, UV index, CO2 levels, and TVOC readings.

Dynamic Data Rendering:

The Flask application dynamically fetches data from the Firebase Database and renders it on the web interface.

Creating a platform to share air quality data involves several steps and considerations. Building an air quality data platform can be a complex endeavor, so it's essential to plan carefully, gather the necessary resources, and stay committed to maintaining and improving the platform over time.