AIR QUALITY MONITORING

PROBLEM STATEMENT

Set up IOT devices to measure air quality parameters like pollution levels and particulate matter. This data can be made publicly available to raise awareness about air quality and its impact on public health. Use any air quality sensor (Eg: VOC sensor) to monitor air pollution levels, particulate matters and various other parameters that define air quality. Connect these sensors to a processor such as Arduino or raspberry pi. Connect an esp8266 module to upload this data to cloud. Make this data publicly available. Create a blog to display this data along with visualizations such as graphs.

DESIGN THINKING

1. Define the Objectives:

Clearly define your project's objectives, such as which air quality parameters you want to monitor, the location(s) for deployment, and how often data should be collected and uploaded.

2. Gather Materials:

- ❖ Collect the necessary hardware components:
- ❖ Air quality sensors (e.g., VOC sensor, PM sensor)
- ❖ Arduino or Raspberry Pi
- ❖ ESP8266 module or other Wi-Fi/Cellular modules
- ❖ Power supply
- ❖ Enclosure for outdoor deployment
- ❖ Cables and connectors

3. Sensor Setup:

- ❖ Connect the air quality sensors to the Arduino or Raspberry Pi following the sensor's datasheet and instructions.
- ❖ Write code to read data from the sensors.

4. Data Processing:

- ❖ Use Arduino (C/C++) or Raspberry Pi (Python) to process and store the sensor data.
- ❖ You may want to perform data smoothing or filtering to reduce noise.
- ❖ Store the data in a structured format (e.g., JSON, CSV) on your device.

5. Connectivity:

- ❖ Connect the ESP8266 module to your Arduino or Raspberry Pi to enable Wi-Fi connectivity.
- ❖ Configure the ESP8266 module to connect to your Wi-Fi network.

6. Cloud Setup:

- ❖ Choose a cloud platform for data storage and visualization (e.g., AWS, Azure, Google Cloud).
- ❖ Write code on your Arduino/Raspberry Pi to send data to the cloud using MQTT, HTTP, or another suitable protocol.

7. Data Visualization:

- ❖ Create a web application or use existing platforms (e.g., ThingSpeak, Plotly, Grafana) to visualize the air quality data.
- ❖ Display data in real-time using graphs, charts, and maps.
- ❖ Add historical data storage and retrieval capabilities.

8. Public Availability:

❖ Host the web application on a publicly accessible server or cloud platform.

❖ Ensure the data is accessible to the public through a user-friendly interface.

9. Data Analysis:

❖ Consider implementing data analysis and alerting features to notify the public about critical air quality conditions.

10. Education and Outreach:

❖ Create a blog or website to explain the project's goals, the importance of air quality, and how to interpret the data.

❖ Share the project on social media and collaborate with local environmental organizations to increase awareness.

11. Maintenance:

❖ Regularly maintain and calibrate the sensors.

❖ Monitor the system for any issues and perform updates as needed.

12. Compliance:

❖ Ensure compliance with local regulations and privacy laws when collecting and sharing data.

13. Feedback and Improvement:

❖ Encourage public feedback and use it to improve the system and data presentation. By following these steps, you can create a successful IoT-based air quality monitoring system that raises awareness about air quality issues and empowers the public with valuable information to make informed decisions about their health and environment.