GOVERNMENT COLLEGE OF ENGINEERING, ERODE

அரசினர் பொறியியல் கல்லூரி, ஈரோடு
Government College of Engineering, Erode
(Approved by AICTE, New Delhi and Affiliated to Anna University, Chennai)

B.E Electronics and Communication Engineering

AIR QUALITY MONITORING

Done By

MITHUN KUMARA N G - 731121106032

Under the mentor of **Dr.M.SATHYAKALA**

Department of Information Technology (IT)

Department of Electronics and Communication Engineering.

Government College of Engineering

Erode ,PO ,near Vasavi College,TamilNadu-638316,Affiliated to Anna University ,Chennai.

AIR QUALITY MONITORING

Introduction:

Air Quality Monitoring refers to the collection and measurement of ambient air pollution samples. The data from these samples is compared to clean air standards and historical information regarding air quality levels, along with data reflecting its health and environmental impacts, to determine the state of the air.

Web Development in AQM:

The monitoring system is developed for transmission and reception of the information received from various data-sources with the use of sensors integrated with microcontroller. The wireless sensing real-time data are transmitted into desired form across the network through internet connection. AQMS is able to monitor concentration of CO2, CO, NO2, temperature and relative humidity and stores the concentration values in the database.

- ✓ Data Collection: Establishing data collection systems to gather information from the sensors. This can involve IOT devices, drones, satellites, or stationary monitoring stations.

- ✓ Data Processing: Developing algorithms and software for data processing, quality assurance, and real-time analysis to generate accurate air quality indices and pollutant concentration measurements.

- ✓ Communication: Implementing methods to transmit data to a central server or database. This may include cellular networks, Wi-Fi, or other wireless communication technologies.

- ✓ User Interface: Creating user-friendly interfaces, such as mobile apps or web platforms, to display air quality information to the public or relevant authorities.

- ✓ Data Visualization: Developing graphical representations and maps to present air quality data in an understandable and actionable way.

- ✓ Alerts and Warnings: Integrating alert systems that notify users when air quality reaches unhealthy levels, enabling them to take appropriate precautions.

- ✓ Air Quality Modeling: Utilizing advanced computational models to simulate and predict air quality patterns based on historical data and meteorological conditions. This helps in forecasting air quality and assessing the effectiveness of potential interventions.

- ✓ Mobile Monitoring Units: Developing mobile monitoring units that can be deployed to specific locations or events to assess air quality in real time. These units are especially useful for tracking air quality during emergencies or large gatherings.

- ✓ Big Data and Machine Learning: Harnessing big data and machine learning techniques to analyse vast datasets and identify patterns, trends, and correlations in air quality data. This can lead to more precise pollution source identification and targeted interventions.

- ✓ Mobile Apps and Wearables: The development of mobile apps and wearable devices equipped with air quality sensors provides individuals with real-time data on the air they are breathing. This empowers people to make informed decisions about outdoor activities and can contribute to personal health and well-being.

- ✓ Historical Data Analysis: Historical air quality data can provide valuable insights into long-term trends and the effectiveness of past interventions. Analysing this data can inform future policy decisions and environmental planning.

- ✓ Real-time Feedback Loops: In some cases, air quality monitoring systems are integrated with other systems, like traffic management or heating and cooling systems, to create real-time feedback loops for reducing emissions and improving air quality.

WEB DEVELOPMENT CODE (Python):

Main Script:

```
import sensors # Import your sensor library
import sqlite3# Import your database library
import analysis # Import your data analysis library
import visualization # Import your data visualization library
# Initialize sensors (replace with actual sensor initialization
code)
air_quality_sensor = sensors.AirQualitySensor()
temperature_sensor = sensors.TemperatureSensor()
humidity_sensor = sensors.HumiditySensor()

conn = sqlite3.connect('your_database.db')
cursor = conn.cursor()

# Create the sensor_data table if it doesn't exist
cursor.execute('''
    CREATE TABLE IF NOT EXISTS sensor_data (
        id INTEGER PRIMARY KEY,
        air_quality REAL,
        temperature REAL,
        humidity REAL
    )
''')
conn.commit()
```

```
    # Main data collection loop
    while True:
     # Read sensor data
      air_quality_data = air_quality_sensor.read()
      temperature_data = temperature_sensor.read()
      humidity_data = humidity_sensor.read()
     # Store data in a database (e.g., SQLite, MySQL,
    InfluxDB)
      cursor.execute("INSERT INTO sensor_data (air_quality,
    temperature, humidity) VALUES (?, ?, ?)",
                     (air_quality_data, temperature_data,
    humidity_data))
      conn.commit()
     # Analyze data for air quality index (AQI)
      aqi = analysis.calculate_aqi(air_quality_data)
     # Visualize data (e.g., on a dashboard)
      visualization.update_dashboard(air_quality_data,
    temperature_data,humidity_data, aqi)mperature_data,
humidity_data)
```

Modules:

- **sensors module :**

```
import random
class AirQualitySensor:
    def read(self):
        # Simulate air quality data (replace with actual
implementation)
        return random.randint(0, 100)

class TemperatureSensor:
    def read(self):
        # Simulate temperature data (replace with actual
implementation)
        return random.uniform(20.0, 30.0)

class HumiditySensor:
    def read(self):
        # Simulate humidity data (replace with actual implementation)
        return random.uniform(30.0, 60.0)
```

- **analysis module:**

```
def calculate_aqi(air_quality_data):
    # Your AQI calculation logic here
    aqi = air_quality_data * 2  # Replace with your actual AQI calculation
    return aqi
```

- **visualization module:**

```
        def update_dashboard(air_quality_data, temperature_data,
        humidity_data, aqi):
            # Your visualization logic here
            print(f"Air Quality: {air_quality_data}")
            print(f"Temperature: {temperature_data}")
            print(f"Humidity: {humidity_data}")
            print(f"AQI: {aqi}")
    # Add your visualization code here (e.g., updating a dashboard)
```

Conclusion:

Wifi, Apps and online sites are included to monitor the quality of air which are measured by the Air Quality Monitoring system installed in the appropriate geographical area to monitor the quality of air. It will be helpful for peoples to know about the Quality of air they are breathing. It should be mainly installed in areas surrounded by industries to often check the air quality to avoid severe consequences.