PHASE 03 – IOT BASED NOISE POLLUTION MONITORING

PROJECT DEFINITION:

The project involves deploying IOT sensors to measure noise pollution in public areas and providing real-time noise level data accessible to the public through a platform or mobile app. The primary objective is to raise awareness about noise pollution and enable informed decision-making. This project includes defining objectives, designing the IOT sensor system, developing the noise pollution information platform, and integrating them using IOT technology and Python.

SOFTWARE NEED FOR IOT BASED NOISE POLLUTION MONITORING :

1. Data Acquisition Software

2. Data Processing and Analysis

3. Data Storage and Management

4. IoT Connectivity

5. User Interface (UI)

6. Alerting and Notification System

7.Geospatial Integration

8. Historical Data Analysis

9. Remote Monitoring and Control ..

HARDWARE NEED FOR IOT BASED NOISE POLLUTION MONITORING :

1.Noise Sensors or Microphones

2.Iot Connectivity Modules

3.Microcontrollers or Single-Board Computers (SBCs)

4.Power Supply

5.Weatherproof Enclosures

6.Antennas (for Wireless Connectivity)

7.GPS Modules

8.Cables and Wiring

9.Backup Power Source

ALGORITHM FOR IOT BASED NOISE POLLUTION MONITORING :

Step 1: Start

Step 2: open python

Step 3: write a program for Sensor Node Setup

Step 4: Data Acquisition

Step 5: Data Transmission

Step 6: Data Storage

Step 7: Data Analysis

Step 8: Alerting Mechanism

Step 9: Data Visualization

Step 10: Maintenance and Calibration

Step 11 : Security Measures

Step 12: Data Retention Policy

Step 13 :Scalability

Step 14 :Documentation and Reporting

Step 15 : Stop

PYTHON PROGRAM BASED IOT-Noise Pollution Monitoring:

1. Sensor Node Code :

```python
  Import time

  Import noise_sensor_library  # Replace with the actual library for your noise sensor

  Import mqtt_client  # Replace with the library for MQTT communication


 # Initialize the noise sensor and MQTT client
 Noise_sensor = noise_sensor_library.NoiseSensor()

Mqtt = mqtt_client.MQTTClient()

  Def monitor_noise():

    While True:

      # Read noise level from the sensor

      Noise_level = noise_sensor.read_noise_level()

      # Publish the noise level data to an MQTT topic

      Mqtt.publish("noise_level", str(noise_level))

      Time.sleep(60)  # Adjust the time interval as needed (e.g., once per minute)

  If __name__ == "__main__":

    Monitor_noise()

```

2. MQTT Client Code :

```python
```

```
Import paho.mqtt.client as mqtt

Class MQTTClient:

    Def __init__(self):

        Self.client = mqtt.Client()

        Self.client.connect("mqtt.broker.com", 1883)  # Connect to your MQTT broker

        Self.client.loop_start()

    Def publish(self, topic, message):

        Self.client.publish(topic, message)

If __name__ == "__main__":

    Client = MQTTClient()

```

3.Server/Cloud Application :

On your server or cloud platform, you would have a separate Python script or application to receive and process the data published by the IoT nodes. You can use MQTT libraries or frameworks to set up a server-side MQTT subscriber.

Here's a simplified example using the Paho MQTT library:

```python
```

```
Import paho.mqtt.client as mqtt

Import data_storage  # Replace with your data storage and analysis code

Def on_message(client, userdata, message):

   # Process incoming noise level data

   Noise_level = float(message.payload.decode(“utf-8”))

   Data_storage.store_data(noise_level)

If __name__ == “__main__”:

   Client = mqtt.Client()

   Client.on_message = on_message

   Client.connect(“mqtt.broker.com”, 1883)  # Connect to the same MQTT broker used by the nodes

   Client.subscribe(“noise_level”)

   Client.loop_forever()
```

CONCLUSION FOR IOT BASED NOISE POLLUTION MONITORING :

In conclusion, IoT-based noise pollution monitoring systems offer an innovative and effective solution for addressing the growing concern of noise pollution in urban environments. These systems provide real-time data collection, analysis, and reporting, enabling authorities and communities to take informed actions to mitigate noise pollution. By harnessing the power of interconnected sensors and data analytics, IoT-based noise monitoring not only helps in improving the quality of life but also supports the development of sustainable, quieter cities. It is a promising avenue for a quieter and healthier future, emphasizing the importance of technological advancements in environmental conservation…