Noise Pollution Monitoring System Using IoT

Objective:

A project to monitor and analyze noise pollution in various locations using Internet of Things (IoT) devices.

Components:

1.Noise Sensors: Deploy IoT-enabled noise sensors in targeted areas. These sensors will capture real-time noise levels and convert the data into digital signals.

2.Microcontrollers: Connect the noise sensors to microcontrollers (e.g., Arduino or Raspberry Pi) to process the collected data and communicate with the central server.

3.Central Server: Set up a central server to receive, store, and analyze the noise data from different locations. Use a database to organize and manage the information efficiently.

4.Internet Connectivity: Ensure that each IoT device has a reliable internet connection for seamless data transmission to the central server.

5.User Interface: Create a user-friendly interface, accessible through a web or mobile application, to display noise levels graphically. Users should be able to view real-time data for specific locations and historical trends.

6.Alert System: Implement an alert mechanism to notify relevant authorities or users when noise levels exceed predefined thresholds. This can help in addressing noise pollution promptly.

Hardware Specifications :

1.Air Pollution Sensors. 2.Sound Sensors. 3.Atmega 328 Microcontroller

4.Wifi Modem. 5.LCD Display. 6.LEDs. 7.Transformer

8.Resistors. 9.Capacitors. 10.Diodes

Software Specifications :

1.Arduino Compiler. 2.MC Programming Language: C

Innovation:

Data Collection: Noise sensors continuously collect data and send it to the microcontroller.

Data Processing: Microcontrollers process the data and transmit it to the central server.

Data Storage: The central server stores the received data in a database, organizing it based on location, time, and other relevant parameters.

User Interaction: Users can access the interface to view noise levels in different areas, set preferences, and receive alerts.

Alerts: If noise levels surpass acceptable limits, the system triggers alerts through the application or other communication channels.