**Noise Pollution Monitoring**

**Design thinking:**

**Name:** Kajjuru tirumalesh

**Reg no.:**711121106038

Monitoring noise pollution is crucial for understanding its impact on human health and the environment. There are several methods and technologies available for noise pollution monitoring, ranging from basic to advanced. Here's an overview of some common approaches:

**Sound Level Meters (SLMs):** Sound level meters are portable devices designed to measure sound levels in decibels (dB). They typically consist of a microphone, an amplifier, and a display. SLMs are commonly used for spot measurements and are useful for assessing noise levels at specific locations and times.

**Noise Dosimeters:** Noise dosimeters are wearable devices that measure an individual's exposure to noise over a specified period. They are often used in occupational settings to assess the risk of hearing damage for workers. Dosimeters provide data on both the level and duration of noise exposure.

**Permanent Noise Monitoring Stations:** These are fixed installations that continuously monitor noise levels in specific areas. They are commonly used in urban environments, near industrial facilities, or along transportation routes. Data from these stations can be used for long-term trend analysis and to assess compliance with noise regulations.

**Community Noise Monitoring:** Some cities and regions deploy networks of noise monitoring stations to assess noise pollution at the community level. This data can help identify noise hotspots, plan urban development, and implement noise mitigation measures.

**Smartphone Apps:** Various smartphone apps are available for noise monitoring. While these apps may not be as accurate as dedicated instruments, they can provide valuable data when used in large numbers. Some apps allow users to contribute data to noise pollution maps.

**Noise Mapping:** Advanced noise mapping software combines data from multiple monitoring points to create detailed noise maps of an area. These maps can help urban planners, policymakers, and researchers understand the spatial distribution of noise pollution and develop effective mitigation strategies.

**Machine Learning and AI:** Machine learning and AI techniques can be employed to process and analyze large datasets from noise monitoring stations. They can help identify patterns, trends, and correlations in noise data, making it easier to understand and address noise pollution issues.

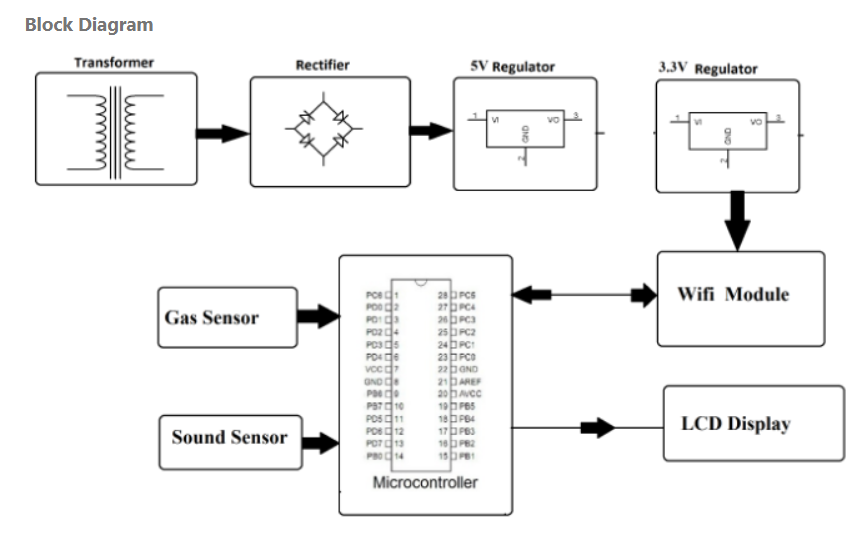
**Remote Sensing:** Remote sensing technologies, such as satellite imagery and aerial surveys, can provide insights into noise sources and their impact on the environment. For example, they can help identify transportation corridors with high noise levels.

**Environmental Impact Assessments (EIAs):** Noise monitoring is a critical component of EIAs for infrastructure projects. EIAs assess the potential noise impacts of construction, transportation, and industrial activities and recommend mitigation measures.

**Noise Barriers and Absorption Materials:** In addition to monitoring, physical interventions like noise barriers and sound-absorbing materials can be used to reduce noise pollution in specific areas.

Effective noise pollution monitoring involves a combination of these methods, depending on the specific objectives and the scale of the problem. It plays a vital role in protecting human health, preserving the environment, and ensuring compliance with noise regulations.

IoT Sensor Design



Using IoT (Internet of Things) noise sensors in various public areas to measure noise levels is an effective and efficient way to monitor noise pollution and gather data for analysis and mitigation efforts. Here are some key considerations for implementing IoT noise sensors in public areas:

**Sensor Selection:** Choose appropriate noise sensors that are capable of accurately measuring noise levels and are designed for outdoor use. These sensors should be capable of capturing data in real-time and be rugged enough to withstand varying weather conditions.

**Sensor Placement:** Determine strategic locations for sensor placement. High-traffic areas, residential neighborhoods, near industrial zones, and transportation hubs are typical locations for noise sensors. Ensure that the sensors are securely mounted and protected from vandalism.

**Data Transmission:** IoT noise sensors should be equipped with wireless communication capabilities (e.g., Wi-Fi, cellular, LoRaWAN) to transmit data to a central server or cloud platform. Make sure there is reliable network connectivity in the chosen deployment areas.

**Power Supply:** Consider the power source for the sensors. Some sensors may be battery-powered, while others can be solar-powered or connected to the electrical grid. The choice depends on the availability of power and the desired maintenance schedule.

**Data Storage and Management:** Establish a central database or cloud-based platform to collect, store, and manage the data generated by the sensors. Ensure data security and backup mechanisms are in place.

**Real-Time Monitoring:** Implement real-time monitoring capabilities to provide immediate feedback on noise levels. This can be valuable for responding quickly to noise complaints or incidents.

**Data Analysis:** Employ data analytics tools to process and analyze the collected noise data. This analysis can reveal patterns, trends, and areas with consistently high noise levels, helping urban planners and policymakers make informed decisions.

**Integration with GIS:** Integrate noise sensor data with Geographic Information Systems (GIS) to create noise maps. This spatial representation of noise levels can aid in visualizing noise pollution hotspots.

**Alerts and Notifications:** Implement alert systems that notify relevant authorities or stakeholders when noise levels exceed predefined thresholds. This can facilitate timely responses and enforcement of noise regulations.

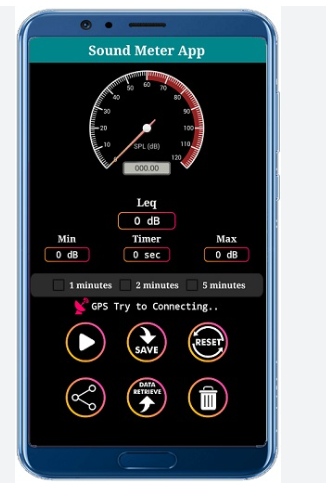
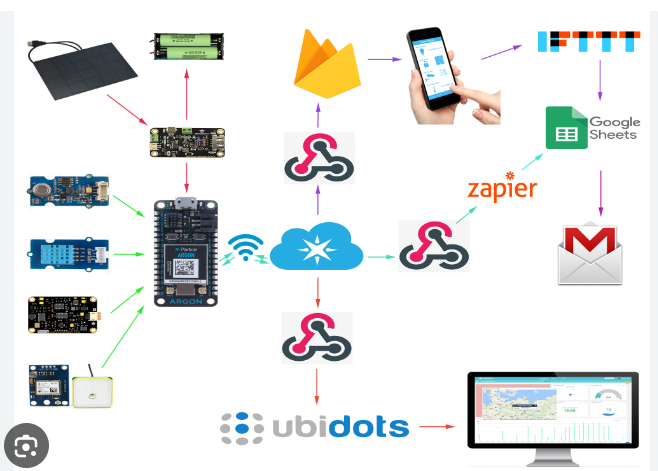
**Public Access:** Consider making some or all of the noise data accessible to the public through websites or mobile apps. Transparent access to this information can raise awareness about noise pollution and engage the community in noise reduction efforts.

**Compliance and Regulation:** Ensure that the deployment of IoT noise sensors complies with local regulations and privacy laws. Data collection and storage should adhere to legal requirements and protect individuals' privacy.

**Maintenance and Calibration:** Regularly maintain and calibrate the sensors to ensure accurate measurements. Scheduled maintenance should be part of the deployment plan to maximize the longevity and reliability of the sensors.

By deploying IoT noise sensors in public areas and effectively managing the data they generate, cities and municipalities can gain valuable insights into noise pollution, implement targeted mitigation strategies, and improve the overall quality of life for residents.

Noise Pollution Information Platform:



Designing a web-based platform and mobile app to display real-time noise level data to the public requires careful planning, user-friendly interfaces, and robust backend infrastructure. Here's a step-by-step guide to create such a system:

**1. Define Objectives and Features:**

Determine the primary goals of the platform and app (e.g., public awareness, noise monitoring, reporting).

Identify the key features, such as real-time noise level data, noise maps, historical data, and user-generated reports.

Decide whether the platform will be used for a specific city, region, or cover a broader area.

**2. Backend Development:**

Develop a robust backend system to collect and store real-time noise data from IoT sensors.

Implement data processing and analysis algorithms to calculate noise levels and identify trends.

Set up a secure database to store historical noise data.

Establish APIs for data retrieval by the web platform and mobile app.

**3. Web-Based Platform:**

Design a user-friendly web interface accessible via standard web browsers.

Include a map displaying noise levels in real time. Users can zoom in and out to view specific areas.

Implement filters and overlays to display noise levels in different colors or contours.

Provide historical noise data charts and graphs for trends and comparisons.

Create a user registration and login system to enable user-generated reports and personalized features.

Allow users to customize notification settings based on noise thresholds or specific locations.

Include a reporting feature for users to submit noise complaints or issues directly through the platform.

**4. Mobile App:**

Develop a mobile app (iOS and Android) with a similar interface and functionality to the web platform.

Optimize the app for various screen sizes and orientations.

Implement geolocation to provide users with noise data relevant to their current location.

Allow users to receive real-time noise level alerts and notifications.

Enable users to take and upload photos or videos as part of noise reports.

Incorporate a "My Account" section for user profiles and personalized settings.

Ensure offline functionality for basic data access in case of network issues.

**5. Data Visualization:**

Use interactive charts and graphs to visualize historical noise data.Implement tooltips and legends to help users interpret noise level colors or values.

Consider integrating a heatmap overlay to provide a quick overview of noise hotspots.

**6. Security and Privacy:**

Prioritize data security and user privacy, adhering to relevant data protection regulations.

Implement user authentication and authorization mechanisms.

Secure data transmission between the app, platform, and backend server using encryption protocols.

**7. Testing and Quality Assurance:**

Conduct thorough testing, including functional, usability, and performance testing.Address any bugs or issues identified during testing.

**8. Deployment and Scalability:**

Deploy the web platform and mobile app to reliable hosting and app stores.Plan for scalability to accommodate increasing user numbers and data volume.

**9. User Engagement:**

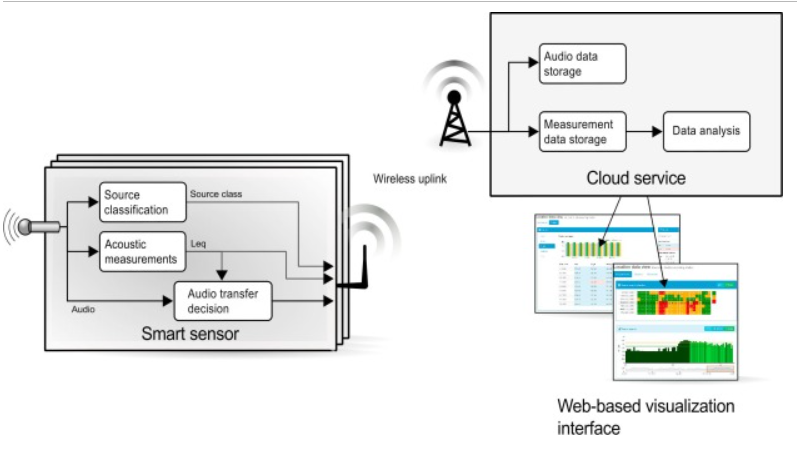
Promote the platform and app through marketing and social media to encourage public usage.

Collect user feedback and make continuous improvements based on user suggestions.

Creating a web-based platform and mobile app to display real-time noise level data to the public is a valuable resource for raising awareness about noise pollution and engaging the community in noise reduction efforts. It requires collaboration with sensor providers, data analysts, and user experience designers to ensure a comprehensive and user-friendly solution.

Top of Form

Integration Approach:



IoT sensors can send data to the noise pollution information platform through various communication methods and protocols. The choice of communication technology depends on factors like sensor location, power constraints, data volume, and network availability. Here are some common methods for IoT sensors to transmit noise level data to the platform:

**Wi-Fi:** If the IoT sensors are located in areas with reliable Wi-Fi coverage, they can transmit data over local Wi-Fi networks. This is a common choice for sensors installed in indoor environments, urban areas, or places with existing Wi-Fi infrastructure.

**Cellular Networks:** IoT sensors can use cellular data connections (3G, 4G, or 5G) to send data to the platform. Cellular connectivity is suitable for remote or mobile sensors that need to transmit data over long distances or in areas without Wi-Fi coverage.

**LoRaWAN:** Low Range Wide Area Network (LoRaWAN) is a low-power, long-range wireless communication technology designed for IoT devices. LoRaWAN is ideal for sensors deployed in remote locations or where long battery life is crucial.

**Satellite Communication:** In areas with no cellular coverage, such as remote rural locations or oceans, IoT sensors can use satellite communication to transmit data to the platform. This method ensures global coverage but may be more expensive.

**Bluetooth:** For short-range communication, IoT sensors can use Bluetooth to connect to a gateway device that forwards the data to the platform. This is common in applications where sensors are in close proximity to each other.

**Mesh Networks:** In some cases, IoT sensors can form mesh networks where they relay data through other sensors until it reaches a gateway or hub connected to the platform. Mesh networks can extend the range of sensor deployments.

**Narrowband IoT (NB-IoT):** NB-IoT is a cellular technology designed for low-power, wide-area IoT applications. It provides efficient data transmission for sensors with low bandwidth requirements.

**MQTT (Message Queuing Telemetry Transport):** MQTT is a lightweight and efficient messaging protocol commonly used for IoT data communication. Sensors can publish data to MQTT brokers, which then relay it to the platform.

**HTTP/HTTPS:** Sensors can send data directly to the platform's server using HTTP or HTTPS protocols. This method is suitable for web-based platforms where sensors have internet access.

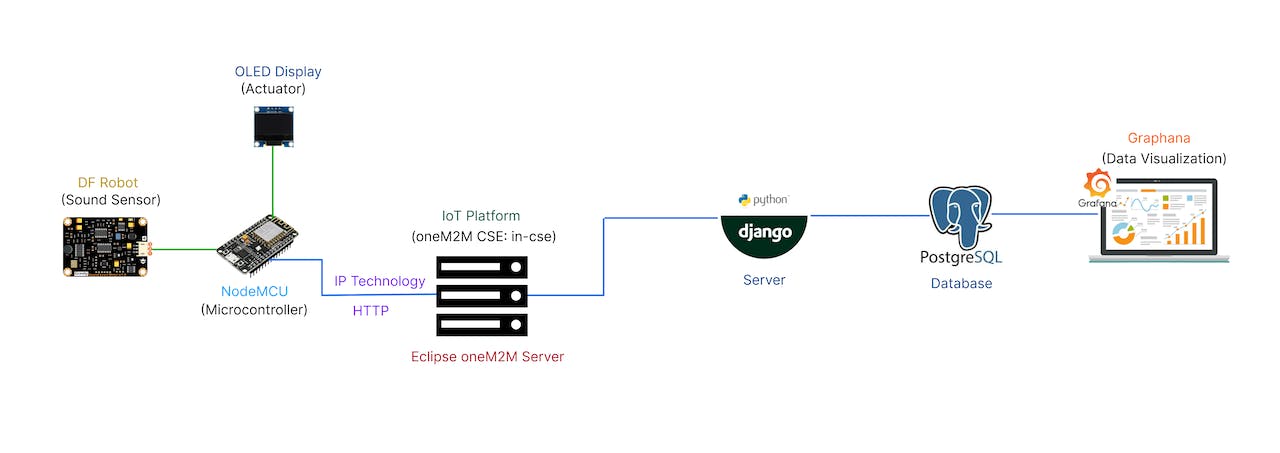
**UDP/TCP:** Sensors can transmit data using User Datagram Protocol (UDP) or Transmission Control Protocol (TCP) to dedicated servers or cloud platforms. UDP is often used for real-time data streaming.

**Local Storage and Batch Upload:** In scenarios where network connectivity is intermittent, sensors can locally store data and upload it in batches when a connection becomes available.

The choice of communication method should consider factors such as sensor power consumption, data security, network availability, and the specific requirements of the noise pollution information platform. Additionally, robust error handling and data validation mechanisms should be in place to ensure data integrity during transmission. The platform's backend should be capable of receiving data from different communication protocols and devices, processing it, and making it available for visualization and analysis.

Top of Form

**Innovation on noise pollution**



**1. Data Collection:**

Sensor Networks: Deploy noise sensors across urban areas to collect real-time noise data. These sensors can be placed strategically to cover different zones.

Open Data Sources: Utilize existing open data sources like traffic patterns, construction schedules, and event calendars, which can help correlate noise levels with specific events or activities.

**2. Data Processing and Analysis:**

Data Integration: Combine noise data with other relevant data sources such as weather data, road traffic data, and land use data. Integration provides a comprehensive view of the factors influencing noise pollution.

Data Cleaning: Noise data often contains outliers and inconsistencies. Implement data cleaning techniques to ensure accuracy in analysis.

Pattern Recognition: Utilize machine learning algorithms to recognize patterns in noise data. For example, patterns related to specific times of the day, days of the week, or seasons.

Predictive Analytics: Build predictive models to forecast noise levels based on historical data and external factors. This can help in proactive planning and management.

**3. Visualization and Interpretation:**

Interactive Dashboards: Create interactive dashboards for policymakers and citizens. Visualization tools can help in understanding noise trends and making data-driven decisions.

Heatmaps: Utilize heatmaps to represent noise levels geospatially. This visual representation can pinpoint noise hotspots in urban areas.

Temporal Analysis: Visualize noise data over time to identify long-term trends and seasonal variations.

**4. Community Engagement:**

Crowdsourced Data: Encourage citizens to contribute noise data through mobile apps. Crowdsourced data can supplement official data and provide a broader perspective.

Public Awareness: Use data visualization to create awareness campaigns. When people can see the noise pollution patterns, they are more likely to actively support noise reduction initiatives.

**5. Policy and Intervention:**

Policy Formulation: Base noise pollution regulations and policies on data-driven insights. For instance, implement specific regulations for noise-intensive activities during sensitive hours.

Targeted Interventions: Use analytics to identify the most critical areas. Target interventions such as noise barriers, altered traffic routes, or stricter regulations in these areas.

Impact Assessment: Analyze the effectiveness of implemented interventions. Data analytics can help assess whether noise levels have decreased after implementing specific measures.

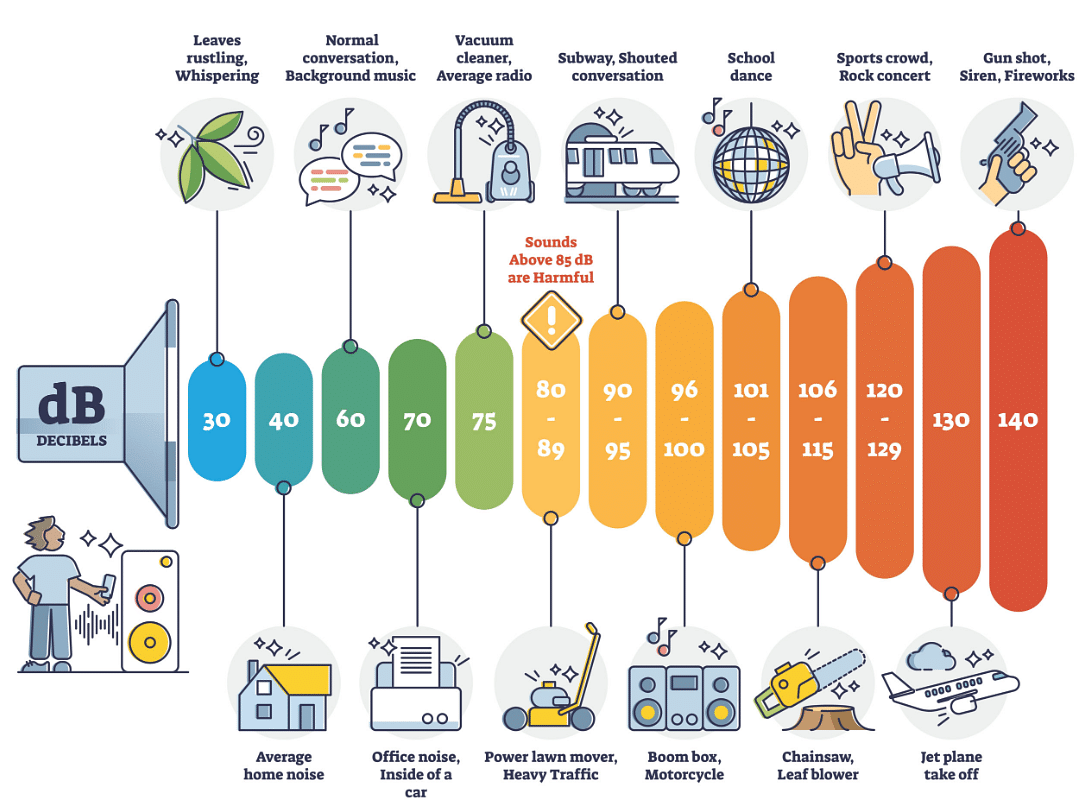
**6. Continuous Improvement:**

Feedback Loops: Establish feedback mechanisms from implemented interventions back into the data analytics system. This feedback loop ensures continuous improvement and adaptation of strategies.

Advanced Analytics: Explore advanced techniques such as deep learning for more accurate predictions and identifying nuanced patterns in noise data.

By incorporating data analytics at every stage, from collection to interpretation, we can develop effective strategies to mitigate noise pollution, leading to healthier and more sustainable urban environments.

**noise areas**



**1. Categorization Criteria:**

Low Noise Areas: Areas where noise pollution levels are below the established threshold. These areas are relatively quiet and suitable for residential living and sensitive activities.

Moderate Noise Areas: Areas where noise pollution levels are slightly above the threshold but do not cause significant disturbance. These areas might include mixed residential and commercial zones.

High Noise Areas: Areas where noise pollution levels exceed the threshold significantly, causing disturbance to residents. These areas often include busy roads, industrial zones, and commercial centers.

Very High Noise Areas: Areas with extremely high noise pollution levels, typically found near major highways, airports, or heavy industrial areas. These areas can have severe health implications for residents.

**2. Data Analysis and Categorization:**

Data Segmentation: Analyze noise data collected from sensors and categorize different locations based on noise levels recorded over time.

Statistical Analysis: Use statistical methods to identify areas where noise levels consistently exceed the thresholds. This analysis helps in categorizing areas accurately.

Machine Learning Classification: Implement machine learning algorithms like decision trees or support vector machines to automatically classify areas into different noise pollution categories based on historical data patterns.

**3. Visualization and Mapping:**

Color-Coded Maps: Create maps with color-coded zones representing different noise pollution categories. This visual representation makes it easy to identify high noise areas at a glance.

GIS Mapping: Utilize Geographic Information Systems (GIS) to create interactive maps that display noise pollution categories. GIS tools provide detailed spatial analysis and visualization options.

**4. Regular Monitoring and Updates:**

Continuous Monitoring: Implement a continuous monitoring system to keep track of noise pollution levels in different areas. Regular updates ensure that new high noise areas are promptly identified.

Dynamic Thresholds: Consider dynamic thresholds based on time of day or specific events. For example, noise pollution limits could be lower during nighttime when residents expect a quieter environment for sleep.

**5. Public Awareness:**

Communication: Inform residents about noise pollution categories in their area and what each category means in terms of potential disturbances.

Community Engagement: Involve communities in understanding noise pollution data. Citizen engagement can lead to more effective noise reduction initiatives.

Categorizing high noise areas based on noise pollution levels provides a clear framework for urban planners and policymakers to prioritize interventions. It ensures that efforts are concentrated on areas where noise pollution has the most significant impact on residents' quality of life.

**Potential sources of noise pollution**

**1. Transportation:**

Road Traffic: Vehicles, especially heavy trucks and motorcycles, can produce significant noise, especially at high speeds.

Rail Traffic: Trains passing through urban areas can create substantial noise, especially at intersections.

Air Traffic: Noise from aircraft taking off and landing, especially near airports, contributes to noise pollution.

**2. Industrial Activities:**

Factories: Machinery, manufacturing processes, and equipment in factories produce continuous noise pollution.

Construction Sites: Heavy machinery, drilling, and construction activities generate high levels of noise, especially in urban areas.

**3. Public Gatherings and Events:**

Concerts and Sporting Events: Large gatherings in stadiums or concert halls can create intense noise levels.

Festivals and Parades: Celebratory events with music, fireworks, and cheering crowds contribute to temporary noise pollution.

**4. Recreational Activities:**

Amusement Parks: Rides, music, and excited crowds can create noisy environments.

Nightclubs and Bars: Loud music and patrons’ conversations contribute to noise pollution, especially at night.

**5. Commercial Activities:**

Restaurants and Cafes: Busy eateries with outdoor seating often generate noise, particularly in the evenings.

Shopping Centers: Large shopping malls with bustling crowds and music can be significant noise sources.

**6. Household Activities:**

Appliances: Household appliances such as vacuum cleaners, blenders, and washing machines contribute to indoor noise pollution.

Pets: Barking dogs and other noisy pets can disturb the neighborhood.

**7. Infrastructure:**

Roadwork and Repairs: Maintenance activities on roads and bridges involve heavy machinery and construction noise.

Demolition: Demolishing old buildings or structures generates loud noise.

**8. Natural Sources:**

Thunderstorms: Thunder and lightning during storms can create loud noises.

Waterfalls and Rivers: Natural water bodies can produce noise, especially in hilly terrains.

**9. Equipment and Machinery:**

Generators: Backup generators used during power outages produce noise pollution.

HVAC Systems: Heating, ventilation, and air conditioning systems in buildings can be noisy, especially if not properly maintained.

**10. Traffic Signals and Alarms:**

Horns and Sirens: Vehicle horns and emergency sirens contribute to noise pollution, especially in urban areas with high traffic.

**11. Communication Systems:**

Public Address Systems: Used in public transportation, events, and commercial spaces, these systems can create noise pollution if too loud.

**12. Agricultural Activities:**

Farm Machinery: Tractors, harvesters, and other agricultural machinery can generate noise, especially in rural areas.

Understanding these diverse sources of noise pollution is vital for policymakers and urban planners to develop effective strategies to mitigate the adverse effects of noise pollution on public health and well-being.

**Things used in this project:**

**1.Hardware components**

**a. LM393 sound sensor**

**Sound Detection Sensor Module Features & Specifications**

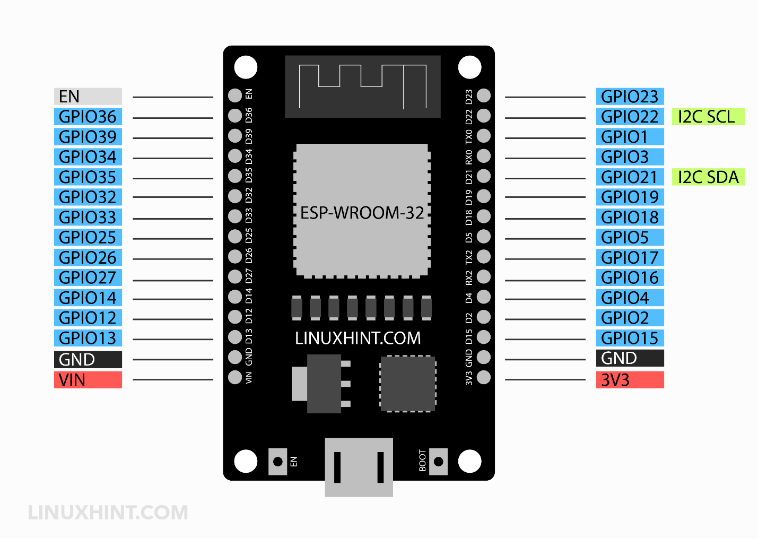
* Operating Voltage: 3.3V to 5V DC
* LM393 comparator with threshold preset
* PCB Size: 3.4cm \* 1.6cm
* Induction distance: 0.5 Meter
* Operating current:  4~5 mA
* Microphone Sensitivity (1kHz): 52 to 48 dB
* Easy to use with Microcontrollers or even with normal Digital/Analog IC

### **How to Use Sound Detection Sensor Module**

Sound Detection Sensor Module consists of four pins i.e. VCC, GND, DO, AO. Digital out pin is connected to the output pin of LM393 comparator IC while the Analog pin is connected to Microphone.

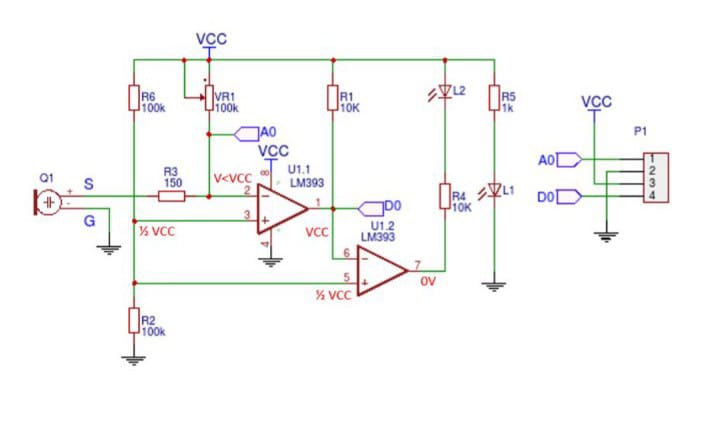
Using Sound Detection Sensor Module with a microcontroller is very easy. Connect the Analog/Digital Output pin of the module to the Analog/Digital pin of Microcontroller. Connect VCC and GND pins to 5V and GND pins of Microcontroller. When the sound level exceeds the setpoint, an LED on the module is illuminated and the output is set low.

**b.ESP32 Development board**

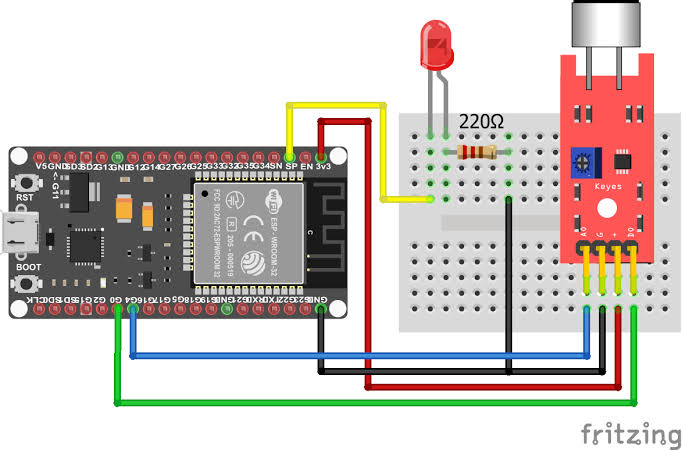


The ESP32 microcontroller has Wi-Fi and Bluetooth functionality, Bluetooth Low Energy (BLE) communication, independent timers, analog to digital and digital to analog converters (ADCs and DACs), capacitive touch sensors, and a Hall effect sensor. The ESP32 microcontroller includes two 240MHz cores, each with a Tensilica Xtensa 32-bit LX6 microprocessor. The ESP32 microcontroller is incorporated in several formats, ranging from a development board to an integrated watch with touchscreen and GPS.

**Circuit diagram**

****

**Schematic diagram :**



Code for execution:

import machine

import time

import urequests

import ujson

import network

import math

# Define your Wi-Fi credentials

wifi\_ssid = 'Wokwi-GUEST'

wifi\_password = ''  # Replace with the actual Wi-Fi password

# Connect to Wi-Fi

wifi = network.WLAN(network.STA\_IF)

wifi.active(True)

wifi.connect(wifi\_ssid, wifi\_password)

# Wait for Wi-Fi connection

while not wifi.isconnected():

    pass

# Define ultrasonic sensor pins (Trig and Echo pins)

ultrasonic\_trig = machine.Pin(15, machine.Pin.OUT)

ultrasonic\_echo = machine.Pin(4, machine.Pin.IN)

# Define microphone pin

microphone = machine.ADC(2)

calibration\_constant = 2.0

noise\_threshold = 60  # Set your desired noise threshold in dB

# Firebase Realtime Database URL and secret

firebase\_url = 'https://noise-pollution-bd0ab-default-rtdb.asia-southeast1.firebasedatabase.app/'  # Replace with your Firebase URL

firebase\_secret = 'nBsgyQFTqHUe4qExlaZX6VL3mpf5gn6BlpnMiuR0'  # Replace with your Firebase secret

def measure\_distance():

    # Trigger the ultrasonic sensor

    ultrasonic\_trig.value(1)

    time.sleep\_us(10)

    ultrasonic\_trig.value(0)

    # Measure the pulse width of the echo signal

    pulse\_time = machine.time\_pulse\_us(ultrasonic\_echo, 1, 30000)

    # Calculate distance in centimeters

    distance\_cm = (pulse\_time / 2) / 29.1

    return distance\_cm

def measure\_noise\_level():

    # Read analog value from the microphone

    noise\_level = microphone.read()

    noise\_level\_db = 20 \* math.log10(noise\_level / calibration\_constant)

    return noise\_level, noise\_level\_db

# Function to send data to Firebase

def send\_data\_to\_firebase(distance, noise\_level\_db):

    data = {

        "Distance": distance,

        "NoiseLevelDB": noise\_level\_db

    }

    url = f'{firebase\_url}/sensor\_data.json?auth={firebase\_secret}'

    try:

        response = urequests.patch(url, json=data)  # Use 'patch' instead of 'put'

        if response.status\_code == 200:

            print("Data sent to Firebase")

        else:

            print(f"Failed to send data to Firebase. Status code: {response.status\_code}")

    except Exception as e:

        print(f"Error sending data to Firebase: {str(e)}")

try:

    while True:

        distance = measure\_distance()

        noise\_level, noise\_level\_db = measure\_noise\_level()

        print("Distance: {} cm, Noise Level: {:.2f} dB".format(distance, noise\_level\_db))

        if noise\_level\_db > noise\_threshold:

            print("Warning: Noise pollution exceeds threshold!")

        # Send data to Firebase

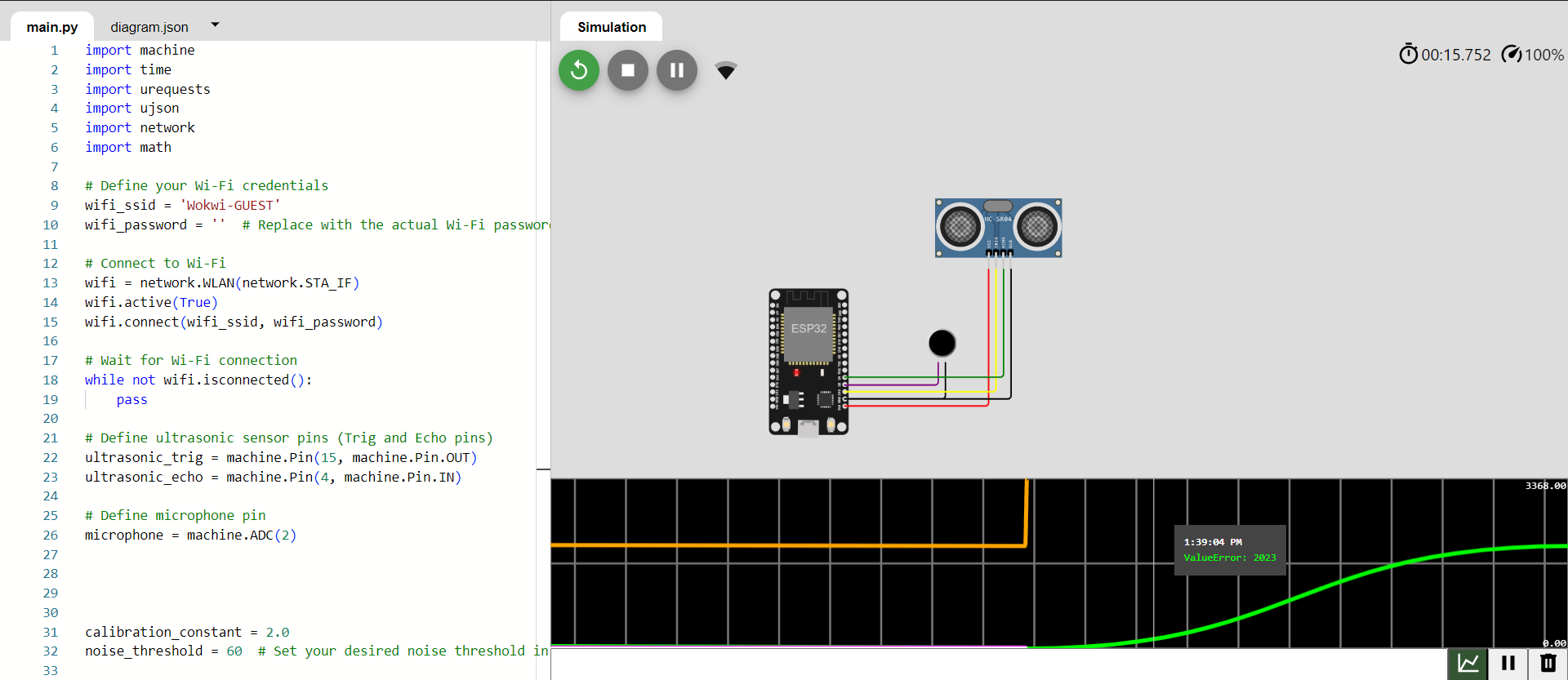
        send\_data\_to\_firebase(distance, noise\_level\_db)

        time.sleep(1)  # Adjust the sleep duration as needed

except KeyboardInterrupt:

    print("Monitoring stopped")

**out put:**



**Real time noise level monitoring system promotes public awareness and contributes to noise pollution mitigation:**

1. **Immediate Feedback:**

RTNLMS provides real-time data on noise levels in specific areas. When people can see the noise levels in their surroundings, they become immediately aware of the noise pollution they are exposed to.

2. **Educating the Public:**

By displaying real-time data in public spaces or online platforms, people can learn about noise levels and their potential health impacts. This knowledge empowers individuals and communities to make informed decisions regarding their activities, like choosing quieter routes or times to travel.

3. **Encouraging Behavior Change:**

Awareness of high noise levels can encourage people to change their behavior. For instance, individuals might reduce the volume of their music, businesses might implement noise reduction measures, and construction activities might be scheduled during less sensitive hours.

4. **Enabling Policy Advocacy:**

Citizens armed with real-time data can advocate for stricter noise pollution regulations. They can use the information to demonstrate the extent of the problem, leading to more stringent policies and regulations.

5. **Targeted Enforcement:**

Authorities can use real-time data to target enforcement efforts. For example, if a certain area consistently shows high noise levels during specific hours, law enforcement can focus their efforts on that area during those times.

6. **Urban Planning and Zoning:**

City planners can use data from RTNLMS to make informed decisions about zoning and urban development. Avoiding residential areas near noisy industrial zones or busy highways can help prevent future noise pollution problems.

7. **Noise Mapping:**

RTNLMS data can be used to create noise maps, illustrating noise levels across different parts of a city. These maps are invaluable for urban planners, helping them identify high-risk areas and plan accordingly.

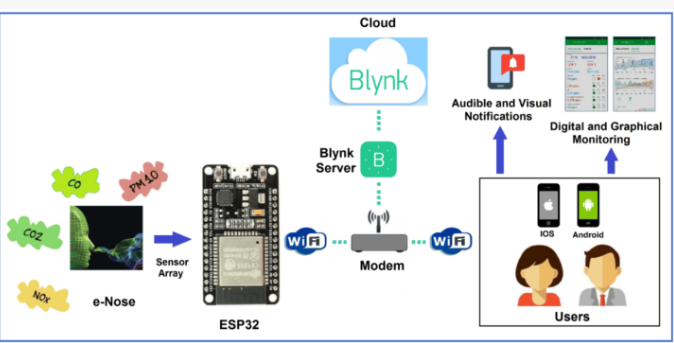
8. **Community Engagement:**

Real-time noise data can be integrated into community engagement programs. Workshops, seminars, or online forums can be organized to discuss the data, educate citizens, and brainstorm solutions collaboratively.

9. **Technological Innovation:**

Public awareness can drive innovation in noise reduction technologies. Increased demand for quieter machinery, vehicles, and appliances can lead to technological advancements that contribute to overall noise pollution reduction.

**THE CONCEPT:**

****