**9238-Mangayarkarasi College Of Engineering**

(Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai) MANGAYARKARASI NAGAR, PARAVAI, MADURAI – 625 402 Website: <http://mce-madurai.ac.in> E- Mail: : [mangai.enggcoll@gmail.com](mailto:mangai.enggcoll@gmail.com)

**Phase-4**

**Noise Pollution Monitoring**

**Project Team Members**

**1.Aarthi Murugan(923821106001)**

**2.N.Aruna Shree(923821106007)**

**3.K.Saranya(923821106045)**

**4.A.Sujitha Jasmine(923821106051)**

**5.S.Shabira(923821106049)**

**6.R.Priyadharshini(923821106037)**

**Project Guide,**

**Mr.R.M.Senthilkumar,AP/ECE**

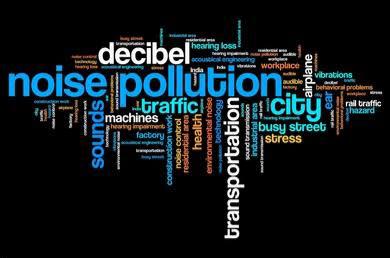
**Introduction:**

**Noise pollution monitoring has taken a Significant step forward with the Integration of IoT technology and web Solutions. This approach allows us to Collect real-time data on Residential areas .Through simulations, we can see how these technologies create virtual data to helping us understand and protect our Natural spaces more effectively.IoT sensors placed within the residential area to collect data on various Environmental factors, which is then sent to central systems using the Internet. Web technologies process this data and create mirroring the real world. This simulation helps us predict The impact of changes and conservation efforts. The combination of IoT and web technologies makes environmental Data accessible to scientists, and the public, enabling Informed decision-making and promoting sustainable practices. This Marks a significant advancement in the world of highlights the power of technology in conservation Efforts.**

****

**Noise Pollution Monitoring**

**Noise or sound level monitoring or measurement is a process to measure the magnitude of Noise in industries and residential area. Data collected from Noise level monitoring & Testing helps us to understand trends and action can be taken to reduce noise pollutionTherefore, careful monitoring is required to avoid excessive levels that could lead to exceedances and even shutdowns. Noise and vibration monitoring is increasingly required to reduce the risk of structural damage, minimize complaints, and comply with regulatory requirements.**

****

**SIMULATION PROCESS**

**WOKWI:**

**Wokwi is a platform that allows you to simulate And test your code for microcontroller-based projects, including Those written in Python for microcontrollers like the ESP32. To Implement an noise pollution monitoring system in a residential areas using Wokwi, you’ll need to follow these steps:**

**1.Create a Wokwi Account:**

**Start by creating an account on the Wokwi platform if you Don’t have one already.**

**2.Select the Microcontroller:**

**In Wokwi, choose the microcontroller you want to work With. For noise Pollution monitoring projects, the ESP32 is a Popular choice due to its built-in Wi-Fi capabilities.**

**3.Design Your Circuit:**

**Using Wokwi’s intuitive drag-and-drop interface, design the Circuit for your noise pollution monitoring system. This may Include adding sensors (e.g.,IR receiver,noise sensors), LEDs, and any other components you need.**

**4.Write the Python Code:**

**In the Wokwi interface, you can write Python code to Interact with the sensors and control the microcontroller. For example, you can use the machine module to Configure pins and sensors and the urequests module to Send data to a server or ThingSpeak.**

**5.Simulate Your Project:**

**Click the “Run” button to simulate your project. You can Observe how your Python code interacts with the virtual Environment, sensors, and microcontroller.**

**6.Test and Debug:**

**Use the simulation environment to test your code for noise pollution monitoring. You can check if the sound level readings are correct and if your data sending Function works as expected. If any issues arise, use the Debug tools provided by Wokwi to identify and resolve Problems in your code.**

**7.Save Your Project:**

**Save your project on Wokwi so you can access it later or Share it with others.**

**8.Explore More Sensors and Components:**

**You can expand your project by adding more sensors and Components to simulate a comprehensive noise pollution Monitoring system. For example, you can add sensors for noise level, light, or GPS to gather more data.**

**PYTHON CODE FOR WOKWI:**

**The provided python program is designed for an ESP8266 based to Monitor noise level using a IR sensor and send that Data to ThinkSpeak, a cloud – based IoT platform .**

**Import machine**

**Import time**

**Import urequests**

**Import network**

**# WiFi and ThingSpeak configuration**

**WIFI\_SSID = “Wokwi-GUEST”**

**WIFI\_PASSWORD = “”**

**THINGSPEAK\_API\_KEY = “14UWOU7T2MK6BB7N”**

**THINGSPEAK\_CHANNEL\_ID = “2321849”**

**# Define IR sensor pin**

**Ir\_sensor\_pin = 22**

**# Function to read IR sensor data**

**Def read\_ir\_sensor():**

**Ir\_sensor = machine.Pin(ir\_sensor\_pin, machine.Pin.IN)**

**Return ir\_sensor.value()**

**# Function to send data to ThingSpeak**

**Def send\_to\_thingspeak(data):**

**url =** [**https://api.thingspeak.com/update**](https://api.thingspeak.com/update)

**headers = {“Content-Type”: “application/x-www-form-urlencoded”}**

**payload = “api\_key={}&field1={}”.format(THINGSPEAK\_API\_KEY, data)**

**try:**

**response = urequests.post(url, headers=headers, data=payload)**

**if response.status\_code == 200:**

**print(“Data sent to ThingSpeak successfully”)**

**else:**

**print(“Failed to send data to ThingSpeak. Status code:”, response.status\_code)**

**response.close()**

**except Exception as e:**

**print(“Failed to send data to ThingSpeak:”, e)**

**# Function to check WiFi connection status**

**Def check\_wifi\_status():**

**Sta = network.WLAN(network.STA\_IF)**

**Sta.active(True)**

**Sta.connect(WIFI\_SSID, WIFI\_PASSWORD)**

**While not sta.isconnected():**

**Pass**

**Print(“Connected to WiFi”)**

**# Connect to WiFi**

**Check\_wifi\_status()**

**# Main loop**

**While True:**

**Try:**

**Noise\_level = read\_ir\_sensor()**

**Print(“Noise Level:”, noise\_level)**

**# Send data to ThingSpeak**

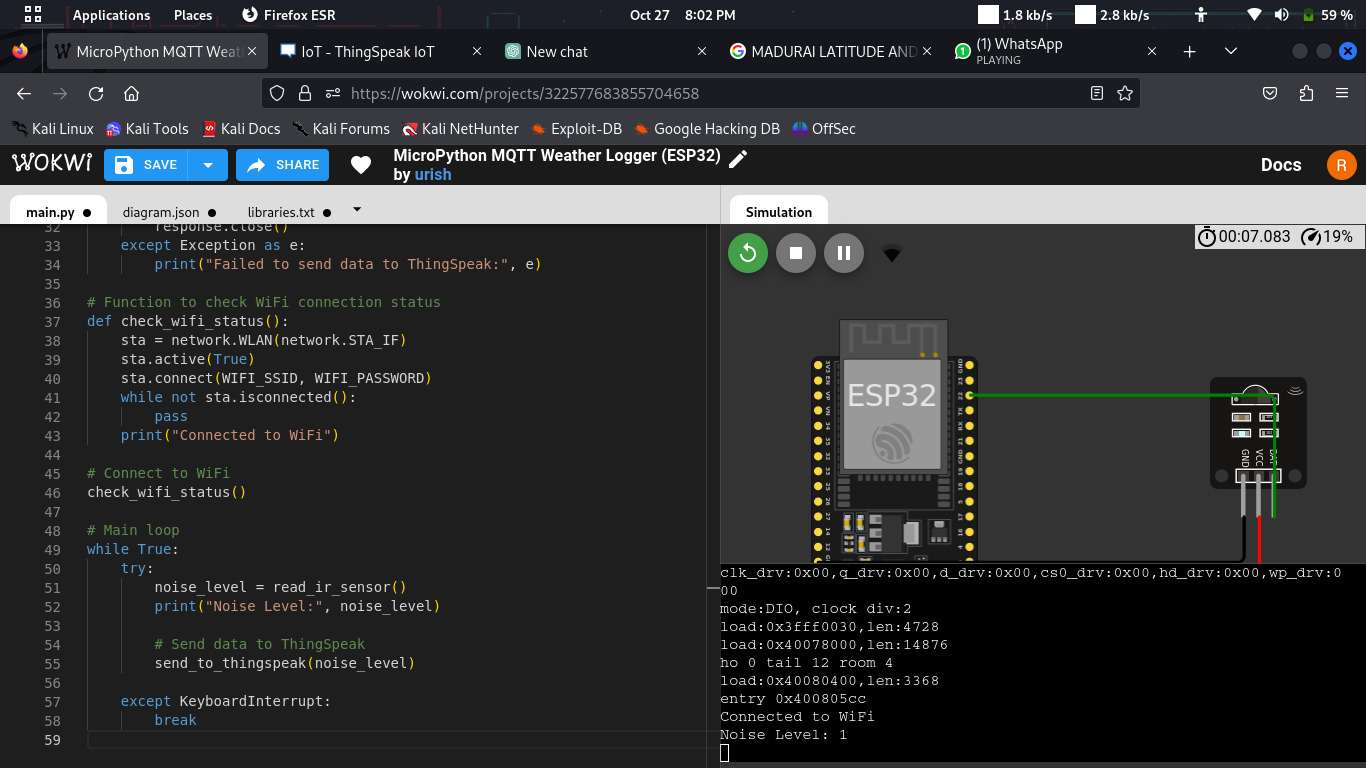
**Send\_to\_thingspeak(noise\_level)**

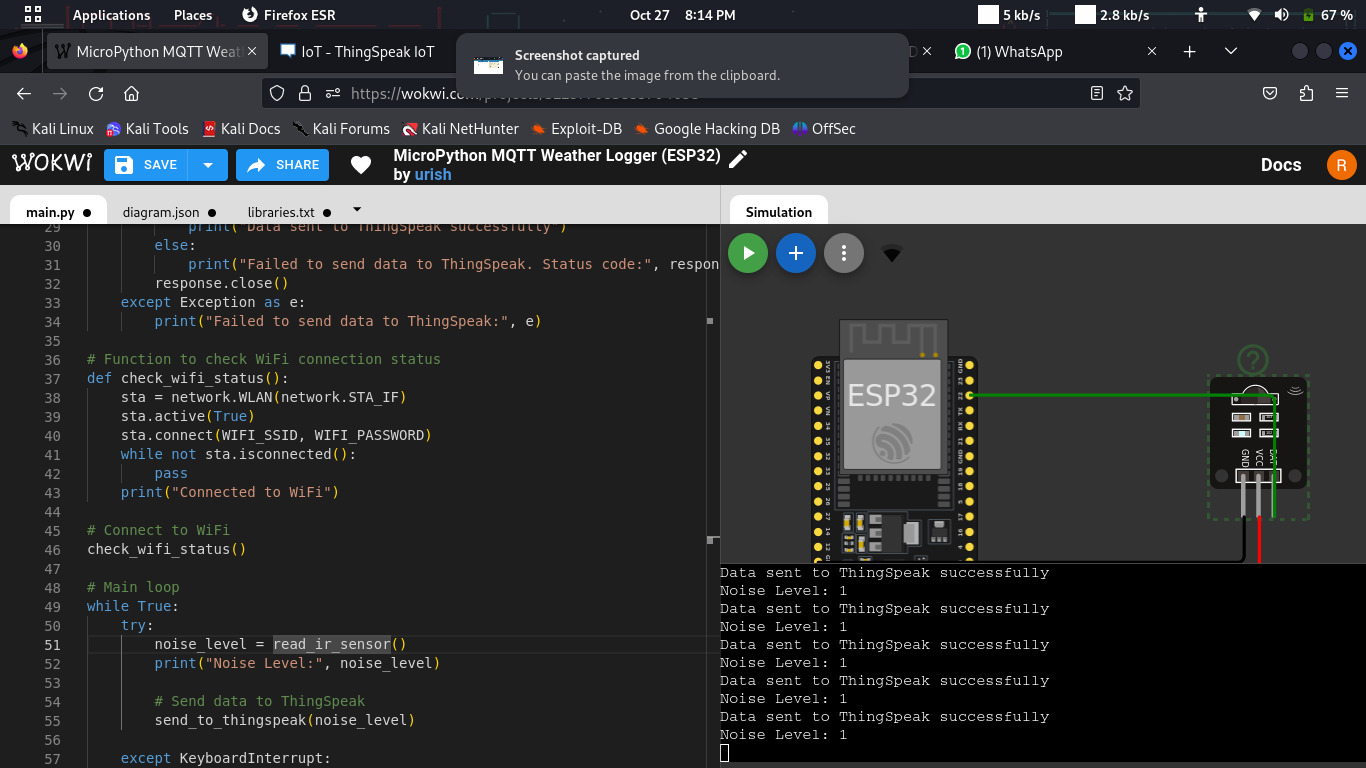
**Except KeyboardInterrupt:**

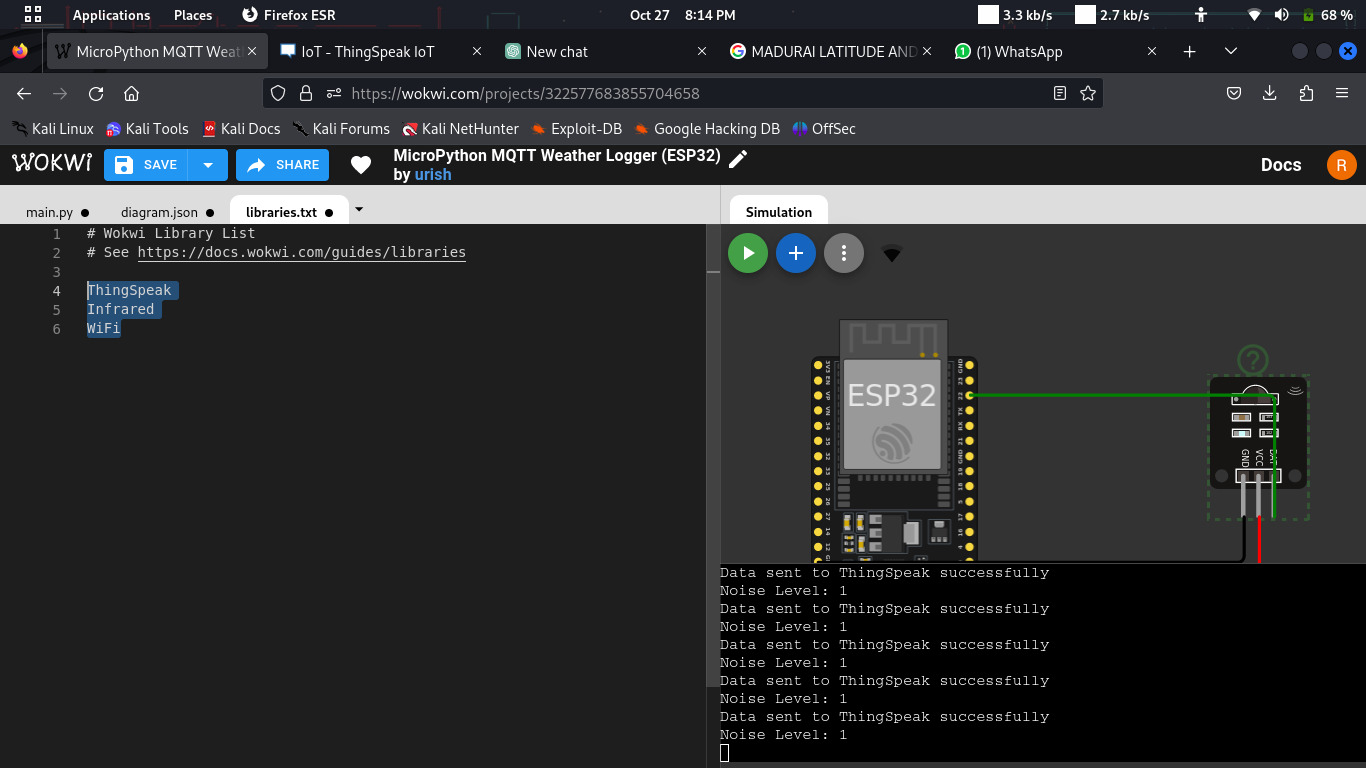
**Break**

**The Above Program Send the data to the Thing speak server. The Data Get From the Microcontroller ESP32.Noise level Can be Get using the IR Sensor. The Data will Send for Every 5 Minutes. The Program Access The ThingSpeak server Via the API key.**

**Output in wokwi:**

****

****

****

**We can adjust the sound level Of the sensor because this is the simulation process Not having a physical components.**

**The Follwing Library Files are used in WOKWI**

**We Must include The Library Files**

**IR sensor**

**WiFi**

**ThingSpeak**

**THINGSPEAK**

**ThingSpeak is our chosen server for your noise pollution Monitoring project.**

**1.Set Up ThingSpeak Account:**

**If you haven’t already, create an account on**

**ThingSpeak (**[**https://thingspeak.com/**](https://thingspeak.com/)**).**

**2.Create a ThingSpeak Channel:**

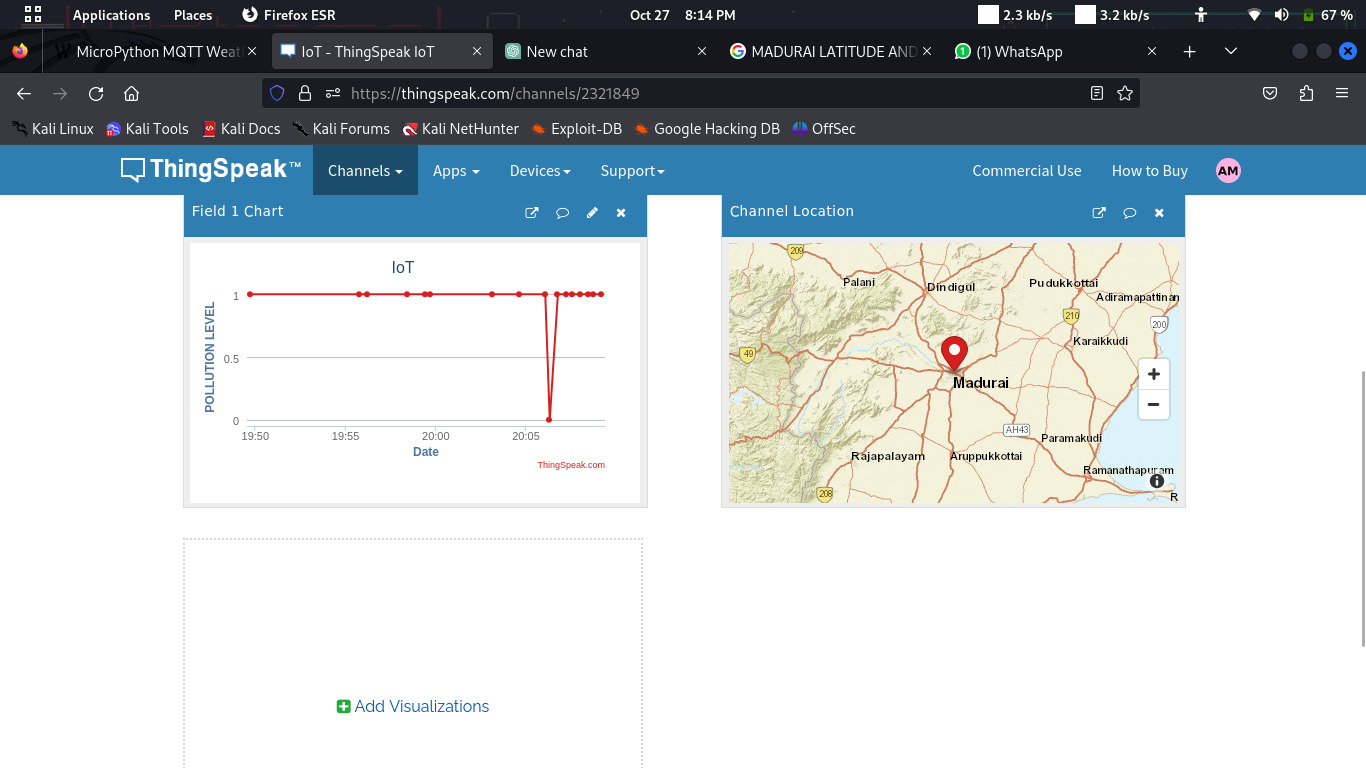
**In your ThingSpeak account, create a new Channel. This channel will be used to store the data From your noise pollution monitoring system.**

**3.Note Your API Key:**

**In the channel settings, you’ll find an API Key. You will need This key to send data to your ThingSpeak channel.**

**4.Integrate ThingSpeak in Your Python Code:**

**In your Python code running on Wokwi, use the Urequests library to send data to ThingSpeak. You can Construct a URL with your API Key and the data you want To send.**

****

**WEB PLATFORM**

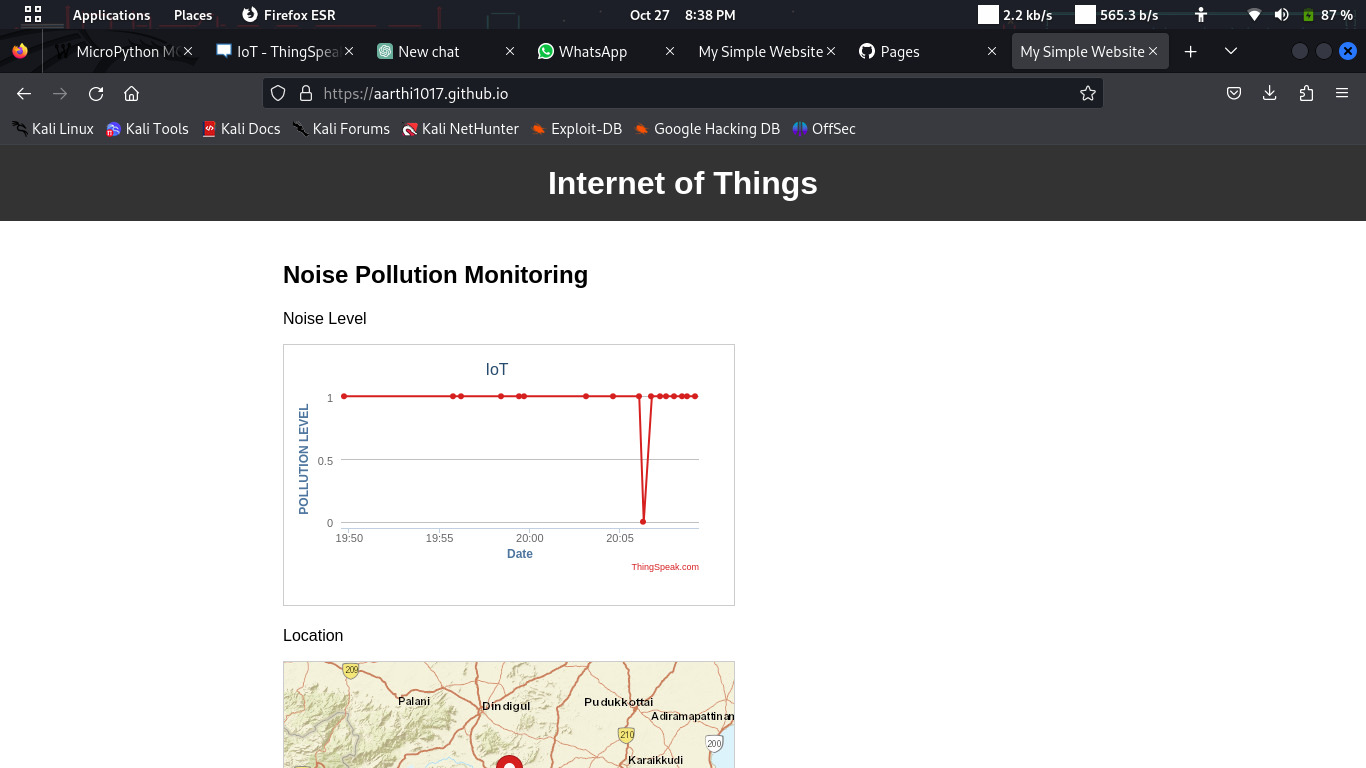
**Our website is a user-friendly platform designed For environmental enthusiasts and data aficionados. It provides Real-time access to essential environmental information, such As noise. The site offers an intuitive and Visually appealing interface, displaying data in an easily Digestible format.**

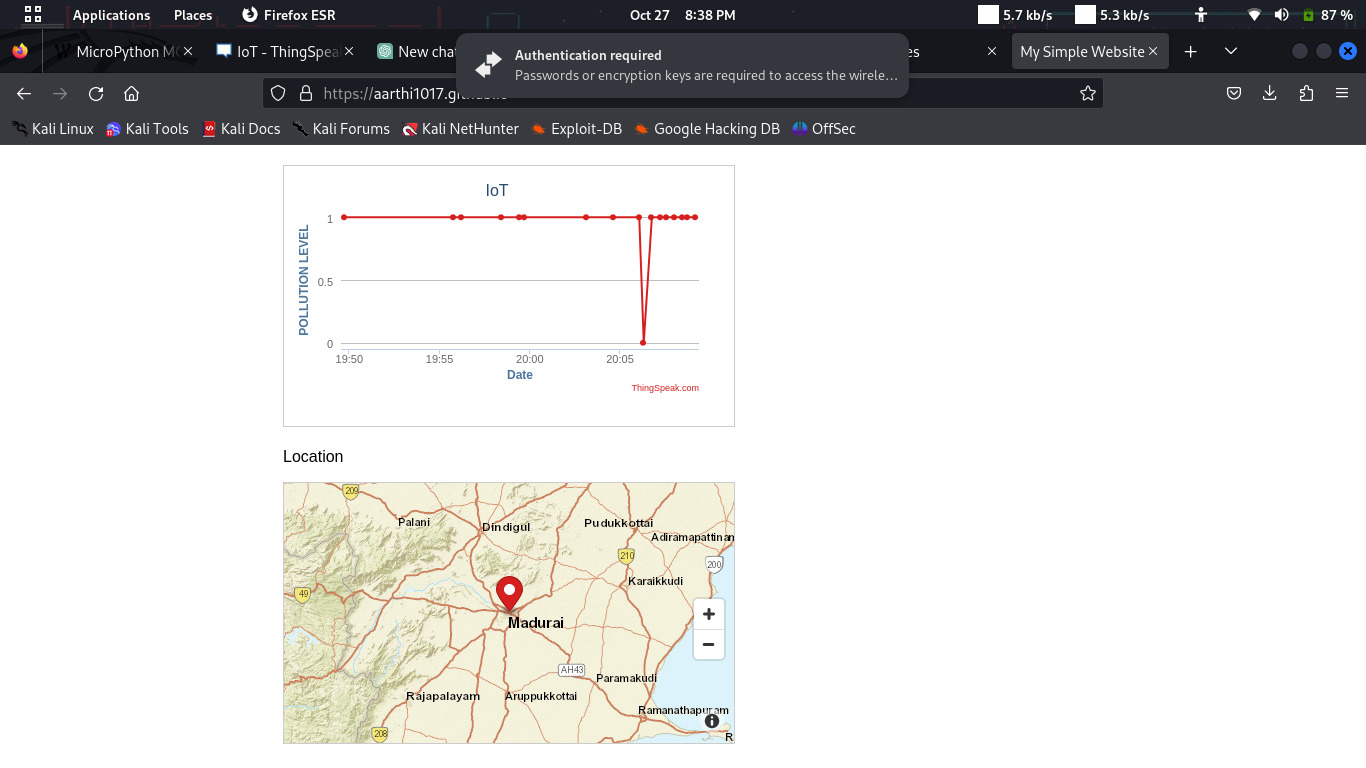
**Users can stay informed about environmental conditions in Their preferred locations, be it parks, gardens, or other outdoor Settings. The website extracts data from ThingSpeak, ensuring Accuracy and reliability. This data is presented with clean, Organized design elements, allowing users to track Environmental trends and make informed decisions.**

**Our website is a valuable resource for both casual observers and Serious environmentalists, offering a seamless and enjoyable Experience for exploring and understanding the world around us.**

**YOU CAN ACCESS OUR WEBSITE USING THE BELOW URL**

[**https://aarthi1017.github.io/**](https://aarthi1017.github.io/)

****

****

**CONCLUSION**

**Noise Pollution Monitoring is a crucial practice that Harnesses the power of IoT and web technologies to ensure the health And sustainability of our natural environments. By simulating this Process with platforms like Wokwi, we can develop and fine-tune our Monitoring systems in a safe and controlled digital environment before Implementing them in real-world settings. The integration of ThingSpeak further enhances our ability to collect, analyze, and display Vital environmental data. As technology continues to advance, the System energy between IoT, web technologies, and environmental Conservation holds the promise of a more informed and interconnected Future for both the protection of our increasing noise level and the enjoyment of Outdoor enthusiasts.**