**Noise Level Monitoring**

**Objective:**

This project aims to provide a website for noise level monitoring. Along with a button to start and stop monitoring, the website offers real-time input on the degree of noise currently present. It also presents noise level historical data in a table manner. High noise level rows are indicated with an alert class, and the historical data is updated dynamically. This project might be used for many different things, like keeping an eye on the noise levels in a certain area or environment.

**Source Code:**

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>Noise Level Monitoring</title>

<style>

body {

font-family: Arial, sans-serif;

background-color: #f5f5f5;

margin: 0;

padding: 0;

}

header {

background-color: #333;

color: #fff;

text-align: center;

padding: 1rem 0;

}

h1 {

margin: 0;

}

main {

max-width: 800px; /\* Increased max-width \*/

margin: 2rem auto;

padding: 2rem;

background-color: #fff;

border-radius: 5px;

box-shadow: 0 0 10px rgba(0, 0, 0, 0.2);

}

#noise-level-display {

text-align: center;

}

h2 {

font-size: 1.5rem;

margin: 0;

}

#noise-level-value {

font-size: 2rem;

color: #007bff;

margin: 1rem 0;

}

footer {

text-align: center;

background-color: #333;

color: #fff;

padding: 1rem 0;

}

/\* Additional CSS \*/

button {

background-color: #007bff;

color: #fff;

padding: 10px 20px;

border: none;

cursor: pointer;

border-radius: 5px;

margin: 10px;

}

#historical-data {

text-align: center;

margin-top: 2rem;

}

table {

width: 100%;

border-collapse: collapse;

}

table, th, td {

border: 1px solid #ddd;

}

th, td {

padding: 8px;

text-align: center;

}

.alert {

background-color: #ff6b6b;

}

</style>

</head>

<body>

<header>

<h1>Noise Level Monitoring</h1>

</header>

<main>

<section id="noise-level-display">

<h2>Current Noise Level:</h2>

<p id="noise-level-value">Loading...</p>

<button id="startStopButton">Start Monitoring</button>

</section>

<!-- Historical Data Section -->

<section id="historical-data">

<h2>Historical Noise Data</h2>

<table>

<tr>

<th>Time</th>

<th>Noise Level (dB)</th>

</tr>

<!-- Add historical data rows dynamically using JavaScript -->

</table>

</section>

</main>

<footer>

<p>&copy; 2023 NaanMudhalvan.com</p>

</footer>

<script>

const noiseLevelValue = document.getElementById("noise-level-value");

const startStopButton = document.getElementById("startStopButton");

const historicalData = document.querySelector("table");

let monitoringStarted = false;

let historicalDataRows = [];

startStopButton.addEventListener("click", () => {

monitoringStarted = !monitoringStarted;

startStopButton.textContent = monitoringStarted ? "Stop Monitoring" : "Start Monitoring";

if (monitoringStarted) {

// Start monitoring and update the button text

startMonitoring();

}

});

function startMonitoring() {

const interval = 1000; // Update interval in milliseconds

const updateNoiseLevel = () => {

const noiseLevel = Math.random() \* 130;

noiseLevelValue.textContent = noiseLevel.toFixed(2) + " dB";

if (monitoringStarted) {

// If monitoring is ongoing, update historical data

const currentTime = new Date().toLocaleTimeString();

historicalDataRows.push([currentTime, noiseLevel.toFixed(2)]);

updateHistoricalData();

}

};

const updateHistoricalData = () => {

// Limit historical data to the last 10 entries

if (historicalDataRows.length > 10) {

historicalDataRows.shift(); // Remove the oldest entry

}

historicalData.innerHTML = "";

historicalDataRows.forEach((row) => {

const newRow = historicalData.insertRow(-1);

row.forEach((cellData) => {

const cell = newRow.insertCell();

cell.textContent = cellData;

});

if (parseFloat(row[1]) > 120) {

// Example: Add an alert class to rows with high noise levels

newRow.classList.add("alert");

}

});

};

// Update the noise level and historical data

updateNoiseLevel();

updateHistoricalData();

// Start a recurring update interval

setInterval(updateNoiseLevel, interval);

}

</script>

</body>

</html>

**Conclusion:**

This project displays an easy-to-use website for tracking noise levels. It provides historical data tracking, alerts for high noise levels, and real-time feedback. Even though the example uses simulated data, it can be used as a basis for developing real-world noise monitoring systems that use sensors. This experiment shows the possibility of developing efficient instruments to control and react to noise levels in different settings.