Naan Mudhalvan Phase 4 Assessment

Course Name : Internet Of Things

Project Title : Environmental monitoring

Team Name is : Tronixx

Team Members :

|  |  |  |
| --- | --- | --- |
| Name | Register number | Naan Mudhalvan ID |
| Thanush T | 721221106112 | au721221106112 |
| PradeepKumar M | 721221106068 | au721221106068 |
| Raja k | 721221106077 | au721221106077 |
| Ragupathi M | 721221106076 | au721221106076 |

**Introduction:**

Real-Time Environmental Monitoring Platform

In today's era of IoT (Internet of Things), monitoring and visualizing real-time environmental data have become critical in various applications, from smart homes to industrial automation. This web page serves as a simplified example of an Environmental Monitoring Platform that provides real-time updates on temperature and humidity.

**Key Features:**

Real-Time Data:

The platform displays up-to-the-second temperature and humidity measurements, allowing users to monitor environmental conditions as they change.

User-Friendly Interface:

The user interface is designed for clarity and ease of use. Real-time data is presented in a visually appealing format for easy comprehension.

Simulated Data:

In this demonstration, data is simulated for educational purposes. In a real-world application, this platform would be connected to IoT devices for actual data retrieval.

**Technologies Used:**

**HTML:** The structure of the web page is created using HTML. It defines the layout, headings, and content.

**CSS:** Cascading Style Sheets (CSS) are employed for styling and layout adjustments. They enhance the visual appeal of the platform.

**JavaScript:** JavaScript is used to simulate real-time data updates. It periodically fetches data and dynamically updates the web page.

Usage:

For educational purposes, this platform simulates real-time environmental data. The "updateData" function in the JavaScript file generates random temperature and humidity values every 5 seconds.

In a real-world scenario, data would be collected from IoT sensors and transmitted to the platform, ensuring accurate monitoring.

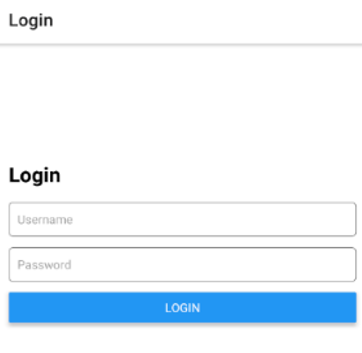
**Customization:**

This platform serves as a starting point. You can extend it by connecting real IoT devices, adding additional data visualization elements, and enhancing the user experience.

Incorporate this platform into your IoT projects, smart homes, or environmental monitoring applications to gain insights into the changing world around you.

This introduction provides an overview of the web page, its purpose, the technologies used, and how it can be customized for practical applications.

**LOGIN PAGE**



**Login Page:**

**HTML:** The login page consists of an HTML form with fields for the username and password. The user is required to enter valid credentials to access the environmental data.

CSS: CSS styling is applied to format and style the login form.

Main Page:

**Navigation Menu**: This section contains navigation links that are initially hidden. It becomes visible after successful login.

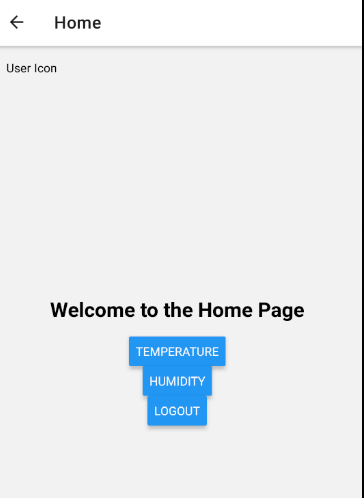
**HOME PAGE:**

**"Temperature":** This link allows the user to view real-time temperature data.

**"Humidity":** This link allows the user to view real-time humidity data.

**"Logout":** This link allows the user to log out and return to the login page.

**Data Display:** Initially, this section is hidden, and it becomes visible when the user selects an option from the navigation menu. It displays real-time environmental data and the location "Madukkarai."



The displayed data is updated based on the user's selection.

JavaScript (script.js):

Form Submission Handling: The JavaScript code handles form submission. It checks if the entered username and password are correct. If they are, it hides the login form and displays the navigation menu.

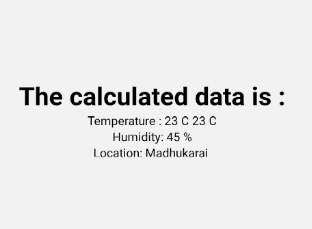
Navigation Menu Event Handlers:

**"Temperature**": When this option is clicked, it triggers a JavaScript function to display simulated **temperature and humidity data in the data display section.**

**"Humidity":** Clicking this option triggers a function to display simulated humidity data.

**"Logout":** Clicking the logout option clears the login form, shows the login form again, and hides the navigation menu and data display section. It effectively logs the user out.

**FINAL DATA:**



**Data Display:**

This section displays real-time environmental data. Initially, it is hidden, and it is updated with the relevant data when the user selects "Temperature" or "Humidity" from the navigation menu.

Simulated Data:

Simulated data is provided in the JavaScript code to mimic real-time environmental monitoring. In a real-world scenario, this data would be fetched from actual IoT devices.

**Conclusion:**

In this project, we have created a simplified web-based Environmental Monitoring Platform using HTML, CSS, and JavaScript. The platform serves as an educational demonstration of how to monitor and visualize real-time environmental data from IoT devices

Javascript:

// Function to handle form submission

function handleLogin(event) {

event.preventDefault();

const username = document.getElementById("username").value;

const password = document.getElementById("password").value;

// Check if the username and password are correct (you can replace this with your authentication logic)

if (username === "yourUsername" && password === "yourPassword") {

// Authentication successful

document.getElementById("loginForm").style.display = "none"; // Hide the login form

document.getElementById("menu").style.display = "block"; // Show the menu

} else {

alert("Invalid credentials. Please try again.");

}

}

// Attach the form submission handler

document.getElementById("loginForm").addEventListener("submit", handleLogin);

// Navigation Menu Click Handlers

document.getElementById("temperature-link").addEventListener("click", displayTemperature);

document.getElementById("humidity-link").addEventListener("click", displayHumidity);

document.getElementById("logout-link").addEventListener("click", logout);

// Function to display temperature data

function displayTemperature() {

const temperature = (Math.random() \* 10 + 20).toFixed(2); // Simulated temperature

const humidity = (Math.random() \* 20 + 40).toFixed(2); // Simulated humidity

const dataText = `Temperature: ${temperature} °C, Humidity: ${humidity}%`;

document.getElementById("data-text").textContent = dataText;

}

// Function to display humidity data

function displayHumidity() {

const humidity = (Math.random() \* 20 + 40).toFixed(2); // Simulated humidity

const dataText = `Humidity: ${humidity}%`;

document.getElementById("data-text").textContent = dataText;

}

// Function to log out

function logout() {

document.getElementById("loginForm").reset(); // Clear the login form

document.getElementById("loginForm").style.display = "block"; // Show the login form

document.getElementById("menu").style.display = "none"; // Hide the menu

document.getElementById("data-display").style.display = "none"; // Hide the data display

}

// Simulate data update (same as before)

setInterval(displayTemperature, 5000);

Html:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Environmental Monitoring Platform</title>

<link rel="stylesheet" href="styles.css">

</head>

<body>

<header>

<h1>Environmental Monitoring Platform</h1>

</header>

<main>

<!-- Login Form -->

<div class="login-form">

<h2>Login</h2>

<form id="loginForm">

<label for="username">Username:</label>

<input type="text" id="username" name="username" required><br>

<label for="password">Password:</label>

<input type="password" id="password" name="password" required><br>

<button type="submit">Log In</button>

</form>

</div>

<!-- Navigation Menu (Hidden Initially) -->

<div class="menu" id="menu" style="display:none;">

<h2>Menu</h2>

<ul>

<li><a href="#" id="temperature-link">Temperature</a></li>

<li><a href="#" id="humidity-link">Humidity</a></li>

<li><a href="#" id="logout-link">Logout</a></li>

</ul>

</div>

<!-- Data Display -->

<div class="sensor-data" id="data-display" style="display:none;">

<h2>Real-time Data</h2>

<p id="data-text"></p>

<p>Location: Madukkarai</p>

</div>

</main>

<script src="script.js"></script>

</body>

</html>