

Programming Task for Students: Room Management Application

Overview

You are tasked with building a **Room Management Application** where users can add, view, and select rooms. The application will consist of a frontend (HTML, CSS, JavaScript) and a backend (using json-server to simulate a REST API). This is an **individual project**, and your understanding of the code will be thoroughly evaluated by the teacher through detailed questions.

Requirements

1. **Individual Work:**
 - This project must be completed individually. Any collaboration, code sharing, or plagiarism will result in an automatic failure.
 - You must write all code yourself, though you may refer to official documentation (e.g., MDN, json-server docs) for guidance.
2. **Teacher Evaluation:**
 - After submission, the teacher will ask detailed questions about your code (e.g., “Why did you use this approach?”, “What does this function do?”, “How does this API call work?”).
 - If you cannot provide correct and detailed answers, you will fail the assignment, even if the code works.
3. **Backend with json-server:**
 - Use json-server to create a mock REST API for managing rooms.
 - Install json-
4. **Nice looking UI:**
 - Use vanilla css to create balanced UI with your own styling
 -
 -
5. **How the app should work:**
 - You can see all existing rooms and add new rooms.
 - Rooms should be presented in a grid and it must be responsive to screen size. 1 column in vertical mobile screen, more columns when screen is wider.
 - When selecting one room, you can see what sensors are there and add/modify/delete sensors
 - When selecting on sensor, you can see the measurements (id, timestamp, + 2 measurement of you own choice) of that sensor. You can add/modify/delete measurements.
 - You can filter measurements of some sensor

Tasks:

- **Create:** Users can create a room map by adding rooms, and within each room, add sensors to monitor environmental data. Measurements for each sensor are manually entered through a form.
- **Read:** Fetch and display sensor readings in a table format, showing relevant details for each measurement.

- **Update:** Edit the details of rooms, sensors, and their associated readings (e.g., sensor name, location, and measurement values).
- **Delete:** Remove sensors (and associated readings) or individual readings from the system.
- **Filter:** View sensor data based on specific criteria, such as time range and sensor type.