

Real-time Audio Chat Rooms - Technical Challenge

Deadline: 2-3 Days

Objective:

Develop a web application that facilitates real-time audio conversations in dedicated chat rooms, complemented by a side text chat.

Core Requirements:

1. User Accounts:

- Implement user registration and login functionality.
- Users should have a basic profile to manage their display name or other essential information.

2. Room Creation:

- Users can create new audio chat rooms.
- Rooms must be designated as either **Public** or **Private** during creation.

3. Room Status & Lobby System:

- A newly created room is initially in an **Inactive** state.
- The room creator must explicitly initiate an action to make the room **Live** before anyone can join.
- If a user attempts to join an **Inactive** room (especially private ones), they will enter a "lobby" state, waiting for the creator to make the room **Live**.

4. Private Room Access:

- Joining a **Private** room requires a specific access code provided by the room creator.

5. Real-time Communication (within a Live Room):

- **Audio Chat:** Participants must be able to engage in a live, multi-party audio conversation.

- **Side Text Chat:** A separate, concurrent text-based chat functionality must be available within the room. This text chat **must be implemented using WebSockets**.
-

Your Task:

Build this application using the **MERN stack** (MongoDB, Express.js, React, Node.js) and appropriate real-time communication technologies. Focus on creating a robust, intuitive, and user-friendly experience.

Bonus Considerations (Optional but Recommended):

- Implement caching strategies (e.g., for user data, room lists) to demonstrate improved performance.
 - Showcase good practices in state management, error handling, code structure, and security.
-

Deliverables:

1. A functional web application demonstrating all core requirements.
2. A brief written overview covering your key architectural decisions.