Project Overview: Realtime Chatroom Platform

Project Title:

Realtime Chatroom Platform

Timeline:

Started: March 2025

Completed: May 2025

Technologies Used:

- Frontend: HTML, CSS, JavaScript
- Backend: Node.js, Express.js
- Database: MongoDB Atlas
- Real-time Communication: Socket.io (WebSockets)
- Authentication: JWT (JSON Web Tokens)
- Deployment: Vercel (frontend), Render (backend)

Objective:

To design and develop a secure, full-stack, real-time chat application that allows users to communicate instantly via public or private rooms with user authentication and room-level access control.

Platform Highlights:

1. User Authentication

- Secure signup/login system using JWT tokens
- Authenticated sessions for protected access
- Local storage of tokens on client-side for persistence

2. Room-Based Messaging

- Users can:
  - Join public rooms (e.g., Study, Chill, Games)
  - Create or join private rooms with:

  - A room name
  - A 4-digit password
  - Auto-assigned room ID
- Room details are stored in MongoDB for easy access

3. Real-Time Messaging
- Implemented using Socket.io to ensure:
  - Low-latency two-way communication
  - Broadcast of messages only within the current room
  - Realtime UI updates without refreshing

4. Chatroom Features
- Welcome message displaying user and room details
- Message timestamps
- Auto-scroll on new messages
- Notifications when users join or leave the room
- "Enter" key to send messages for ease

5. Responsive UI with Mode Toggle
- Fully responsive design for mobile, tablet, and desktop
- Light/Dark mode toggle with persistence across sessions
- Clean, user-friendly interface for smooth experience

6. Cloud Hosting
- Frontend hosted on Vercel
- Backend hosted on Render
- Environment variables managed securely
- Always accessible via browser, no installation required

Security Measures:
- JWT used for securing APIs and Socket connections
- Password-protected rooms with secure validation
- Environment-based MongoDB connection strings

- Input validations on frontend and backend

Folder Structure Overview:

realtime-chatroom/

??? client/ # Frontend files (HTML/CSS/JS)

??? server/ # Backend server with routes & sockets

? ??? auth/ # Login & register routes

? ??? rooms/ # Room creation & join logic

? ??? index.js # Main server setup with Socket.io

Learning Outcomes:

- Gained full-stack development experience
- Applied real-time concepts using WebSockets
- Understood deployment of scalable web apps
- Practiced user authentication and security
- Improved UI/UX development and responsive design

Future Enhancements:

- Chat history saving for all rooms
- File/image sharing within chat
- Admin/moderator tools (kick/mute users)
- Emojis and message reactions
- Mobile app version using React Native or Flutter

Conclusion:

The Realtime Chatroom Platform is a secure, real-time, and user-friendly messaging app built using modern full-stack technologies. With its clean design, robust backend, and real-time efficiency, it serves as a scalable base for future social or collaboration tools. It demonstrates clear understanding of web development, socket programming, UI design, and deployment best practices.