Task Management Tool

Overview

Develop a task management tool using the MERN stack within 2 days. The tool should allow users to create accounts, manage tasks, and collaborate with others in real-time.

Requirements

Frontend (React)

1. **User Registration and Login**:
   - Create a registration form that accepts username, email, and password.
   - Implement login functionality using JWT authentication.
2. **Task Creation and Management**:
   - Design a form for users to create tasks with due dates and priorities.
   - Implement a task list that displays all tasks with their due dates and priorities.
   - Enable users to edit and delete tasks.
3. **Real-time Collaboration**:
   - Implement a real-time collaboration feature using WebSockets.
   - Allow multiple users to collaborate on tasks in real-time.
4. **Search Bar**:
   - Implement a search bar that allows users to search for tasks by title, description, or due date.

Backend (Node.js/Express)

1. **User Authentication and Authorization**:
   - Implement JWT authentication for user registration and login.
   - Use middleware to authenticate and authorize users for protected routes.
2. **Task API**:
   - Create API endpoints for creating, reading, updating, and deleting tasks.
   - Implement validation and error handling for API requests.
3. **Real-time Collaboration API**:
   - Create API endpoints for real-time collaboration using WebSockets.
   - Implement validation and error handling for API requests.

4. **Database Integration**:
    - Integrate the task management tool with a MongoDB database.
    - Implement data modeling and schema design for tasks and users.

Database (MongoDB)

1. **User Collection**:
    - Store user information (e.g., username, email, password).
2. **Task Collection**:
    - Store task information (e.g., title, description, due date, priority).
3. **Collaboration Collection**:
    - Store real-time collaboration data (e.g., user IDs, task IDs, collaboration history).

Evaluation Criteria

1. **User Experience and Interface** (30%):
    - Is the user registration and login process smooth and intuitive?
    - Is the task creation and management process user-friendly and feature-rich?
    - Is the real-time collaboration feature easy to use and visually appealing?
2. **Backend API and Database Integration** (25%):
    - Are the API endpoints well-organized and properly validated?
    - Is the database schema well-designed and normalized?
    - Are the API requests and responses properly handled and error-checked?
3. **Real-time Collaboration Feature** (20%):
    - Is the real-time collaboration feature functional and easy to use?
    - Are multiple users able to collaborate on tasks in real-time?
4. **Security and Authentication** (25%):
    - Is user authentication and authorization properly implemented using JWT?
    - Are user passwords properly hashed and stored?
    - Are API requests and responses properly secured using HTTPS?

Timeframe

- Hackathon duration: 2 days
- Submission deadline: End of day 2

Resources

- MERN stack documentation and tutorials
- MongoDB Atlas for database hosting
- WebSocket library (e.g., (link unavailable)) for real-time collaboration