Hackathon Task: Intelligent Task Management System

Objective:

Develop an intelligent task management system where users can create, manage, and collaborate on tasks in real-time. The application should use Clerk for authentication and Socket.io for real-time updates, leveraging the full MERN stack.

Task Breakdown:

Day 1: Setup and Core Features

Project Setup:

- Set up the MERN project structure for both frontend and backend.
- Integrate Clerk for user authentication and authorization.
- Create user models and authentication routes.

Excellent UI/UX:

- Design and implement the UI using React; frontend CSS development frameworks can be used.
- Set up React Router for navigation between different views (e.g., login, dashboard, task details).

Task Management:

- Implement backend routes for creating, updating, deleting, and fetching tasks using Express.js and MongoDB.
- Create a form in React to create and edit tasks with fields for task name, description, due date, priority, and status.

Day 2: Real-Time Collaboration and Advanced Features

Real-Time Updates:

- Integrate Socket.io for real-time updates.

- Implement real-time notifications for task updates (e.g., new tasks, updated tasks).
- Ensure that all connected users see updates instantly when a task is created, updated, or deleted.

Collaboration Features:

- Allow users to share tasks and collaborate on them.
- Implement role-based permissions (e.g., task creator, collaborator, viewer).

Day 3: Polishing and Final Touches

UI Enhancements:

- Improve the overall design and responsiveness of the UI.
- Add loading spinners, error messages, and success notifications.

Search and Filters:

- Implement search functionality to find tasks by name, description, or tags.
- Provide filters to view tasks by status, priority, due date, or collaborators.

Validation:

- Implement proper validation and sanitization of user inputs (using Yup or Zod to validate form inputs).

Deployment:

- Deploy the application to a cloud service (e.g., Netlify, Render) and share the link.
- Ensure the application is accessible and provide a live demo using Loom describing your web-app and its features and share the link.

Submission Guidelines:

The task deadline is 3 days. Earlier and neat submission will be scored higher by the jury,

GitHub Repository: Provide a link to the GitHub repository with clear instructions on how to run with the credentials.

Hosted Web-app URL: Provide the link that you have hosted your web – app with credentials for testing.

Live Demo: Deploy the application and provide a live demo link by explaining about the web app and detailed explanation about the project structure, features using a Loom video.

Evaluation Criteria:

Functionality: How well the application meets the requirements and additional features.

Code Quality: Cleanliness, organization, and readability of the code.

User Experience: Intuitive and responsive user interface, UI/UX would be main deciding factor for ranking between 2 similar submissions.

Innovation: Creative and unique features beyond the basic requirements.

Prizes and Opportunities:

- **Top 15:** Cash prizes worth 60,000 INR.

The distribution ensures that the top prizes are significantly higher while still rewarding all top 15 finalists:

1. **1st Place:** 12,000 INR
2. **2nd Place:** 9,000 INR
3. **3rd Place:** 7,000 INR
4. **4th Place:** 5,000 INR
5. **5th Place:** 4,000 INR

6. **6th Place:** 3,500 INR
7. **7th Place:** 3,000 INR
8. **8th Place:** 2,500 INR
9. **9th Place:** 2,000 INR
10. **10th Place:** 1,800 INR
11. **11th Place:** 1,700 INR
12. **12th Place:** 1,500 INR
13. **13th Place:** 1,500 INR
14. **14th Place:** 1,500 INR
15. **15th Place:** 1,500 INR

- **Top 20:** Direct interview opportunities for a remote Full Stack Developer role at Lucid Growth.

This task provides a clear and engaging project for developers, we offer you a chance to win cash prizes and secure an interview for a Full Stack Developer role at Lucid Growth!

Happy coding:)