

Project Overflow

Stack To use:

Frontend: React.js, Redux.js, Tailwind CSS

Backend: Node.js, Express.js, Prisma

Database: PostgreSQL

Real-time: Socket.io

Auth: JWT-based authentication

AI Integration: TensorFlow.js for task prediction

Notifications: Real-time alerts via Web Sockets

Develop backend and database:

1. Develop all models
2. Develop all controllers
3. Develop all routes
4. Develop real-time basic chatting
5. Develop AI integration
6. Develop real-time notification

Develop Frontend:

1. User Authentication (JWT-based login/signup with role management: Admin, Project Manager, Team Member)
2. Dashboard (Overview of projects, tasks, and deadlines)
3. Project Management (Create, edit, delete projects, assign team members, and set deadlines)
4. Task Management (Assign tasks, update statuses, and set priorities)
5. Team Collaboration (Basic messaging system & task comments)
6. Basic AI Integration (Suggest optimized task assignments & predict delays using TensorFlow.js)
7. Notifications (On-screen alerts for task updates and deadlines)
8. Reports & Analytics (Project progress, task completion rate, and team workload stats)