

Roadmap for Collaborative Web Platform

Tech Stack Recommendation:

- Frontend: React.js, TailwindCSS/Bootstrap, Redux (optional), Socket.io
- Backend: Node.js, Express.js, WebSocket (Socket.io), REST APIs
- Database: MongoDB
- AI Integration: Gemini API / OpenAI API
- Video/Audio Chat: WebRTC, Socket.io or 100ms/Agora
- Authentication: JWT + OAuth

Modules Roadmap:

A. Authentication & Team Management:

- Register/Login (email, Google, GitHub)
- Create/join team, invite via link/code
- Assign team leader

B. Team vs Team Coding Battles:

- Monaco Editor + Socket.io sync
- Countdown timer sync
- Code submit -> run via Judge0/Docker
- Evaluate and score

C. Video/Audio Chat:

- WebRTC + Socket.io or 3rd party APIs (Agora/100ms)
- Battle room integration

D. Leaderboard:

- Track wins, accuracy, time
- Global and room-wise stats

E. AI Assistant for Prompt Battles:

- Prompt challenges, MCQs using AI
- Code output, Fix Bug, Theory questions
- Time + Accuracy-based scoring
- AI feedback on answers

F. AI Learning Platform:

- AI tutor based on skill level
- Topic-wise learning (loops, recursion, etc.)
- Interactive Q&A + progress tracking

Folder Structure (MERN):

/client/src/components, /pages, /services, App.jsx

/server/routes, /controllers, /models, index.js

Testing & Deployment:

- Testing: Jest, Cypress
- Deployment: Vercel/Netlify (Frontend), Render/Heroku (Backend), MongoDB Atlas (DB), Railway (Socket server)

Milestone Plan:

- Week 1-2: Auth, teams
- Week 3-4: Editor + battles

- Week 5: Video/audio
- Week 6: AI quizzes
- Week 7: Leaderboard
- Week 8+: AI tutor

Tips:

- Start modular
- Use Socket.io rooms for team logic
- Use Gemini/OpenAI for Q&A generation and feedback