Chess Game Development Challenge

March 13, 2025

Context

Chess remains one of the most popular and intellectually stimulating board games globally, renowned for its strategic depth and complexity. With an increasing interest in digital gaming and remote interactions, creating an engaging, intuitive, and robust digital chess platform has become highly relevant.

Imagine you're part of a gaming startup aiming to deliver an exceptional online chess experience for both casual players and competitive chess enthusiasts. Your goal is to create a desktop-friendly digital chess game offering smooth gameplay, real-time interactions, intuitive user interface, and accurate game logic validation. Your platform aims to attract chess players who value both competitive play and casual gaming sessions, ensuring they can easily access games with friends or other online players.

Your challenge, as a team of four developers, is to develop a fully functional, interactive desktop chess game within a timeframe of **3 weeks**.

Your Task

Requirements:

- Develop a responsive desktop chess game with standard chess rules implementation.
- Include player versus player (local and online) gameplay functionality.
- Ensure valid move highlighting and invalid move prevention.
- Implement basic game state management (check, checkmate, stalemate).
- Include a visually appealing, clean, and intuitive user interface.
- Ensure smooth animations and interactions for piece movements.

Documentation and Resources

To assist in completing this challenge, you may utilize the following resources:

- **Chess Logic Libraries:** Chess.js (Chess.js Documentation), Chessboard.js (Chessboard.js Documentation)
- **Frontend Frameworks:** ReactJS (React Documentation), Angular (Angular Documentation), VueJS (Vue Documentation)
- **CSS and UI Frameworks:** Tailwind CSS (Tailwind Documentation), Bootstrap (Bootstrap Documentation)
- **Real-time Communication:** Socket.IO (Socket.IO Documentation)
- **Deployment Tools:** Netlify (Netlify Documentation), Vercel (Vercel Documentation)

Project Rules

- **Collaboration:** All team members must collaborate using GitHub. Regular commits are mandatory at each step.
- **Mentors Access:** you'll be assigned mentors. You must add them as collaborators to the GitHub repository of your project.

Milestones and Deadlines

To help structure your project, the following milestones must be respected:

1. **Milestone 1 (End of Week 1):** UI/UX design, chessboard setup, and local two-player gameplay functionality implemented.
2. **Milestone 2 (Mid of Week 2):** Move validation logic and basic game state management (checks, checkmates) completed.
3. **Milestone 3 (End of Week 2):** Real-time online player versus player integration completed.
4. **Milestone 4 (Mid of Week 3):** Game animations and interface responsiveness fully implemented.
5. **Milestone 5 (End of Week 3):** Comprehensive testing, bug fixes, deployment, and final presentation.

Tips and Advice

- Define clear roles within your team (Frontend Developer, Backend Developer, Game Logic Developer, Project Manager).
- Begin by thoroughly understanding the chess logic and rules before coding.
- Maintain clear and concise documentation and commenting within your codebase.
- Use version control (GitHub) effectively for collaborative work and resolving merge conflicts early.
- Perform regular testing to ensure game logic accuracy and UI responsiveness.
- Focus on delivering a polished and smooth user experience.

Good luck and enjoy the process of building your chess game!