Project Report: Interactive Chess Game (Web-based)

Hemant Godve

Objective

The objective of this project was to design and implement a fully interactive, web-based chess game using HTML, CSS, and JavaScript. The application provides an engaging chess experience directly in the browser, featuring complete chess logic, user-friendly visual feedback, and turn-based play.

Features

- **Chess Logic:** Implements standard chess rules for piece movement, turn alternation, capturing, pawn promotion, castling, and checkmate detection.
- **Interactive Board:** 8x8 grid rendered with dynamic HTML elements, supporting piece selection, move highlighting, and captures.
- **Animated Visuals:** Each move triggers visual effects for feedback, including greens for valid moves and shakes for highlights; neon-styled text for dynamic feedback.
- **Turn Indicator:** Displays whose turn it is, with styles and highlights to differentiate white and black turns.
- **Responsive UI:** Adaptable layout for different screen sizes using CSS; fluid styling for game cells and board.
- **Theme Support:** Modular CSS structure allows easy addition of multiple themes (e.g., dark and light modes).
- **Performance Optimizations:** Efficient DOM updates and minimized effect animations for seamless gameplay.

Core Technologies Used

- **HTML:** Markup for chess grid, UI elements, and dynamic turn/status panels.
- **CSS:** Custom styles for the board, pieces, and visual effects; neon animations and interactive transitions.
- **JavaScript:** Main game logic, state management, move generation, board updates, event handling, and animations.
- **jQuery:** Used for simplified DOM manipulation and event attachment.

Main Implementation Details

Board Structure: 8x8 HTML grid with unique cell IDs for position tracking and style updates.

Piece Representation: Each piece tracked as a JS object with position, type, captured/moved flags, and associated image/symbol.

Move Logic: Functions for valid move generation per piece type, taking into account piece-specific behavior and interactions, including complex moves (castling, promotion).

Move Highlight: Green highlights and neon-text effects for valid moves; shake animations for selected cells.

User Feedback: Turn highlight, check notifications, and endgame pop-ups.

Themes: Easily switchable `dark.css` and `light.css` styles for improved user experience.

Summary

This project demonstrates the ability to engineer a fully functional and visually engaging chess game in the web environment. It leverages modern web technologies, clean code organization, and interactive UX principles:

- Robust handling of all chess logic and special moves.
- Clear, animated feedback for user actions.
- Highly maintainable, theme-ready front-end codebase.
- Professional presentation and seamless browser experience.

Conclusion

The web-based chess game serves as a testament to best practices in browser-based application development. By integrating core gameplay mechanics, lively animations, and responsive interface design, this project showcases comprehensive proficiency in HTML, CSS, and JavaScript engineering.