Chess Game Applications

Participants name:

Jeet Bannerjee

Abhisekh Biswas

Nandini Barman

Anoushka Debnath

Swarnavo Mondal

Abhisikta Saha

Project Type: Webpage applications with HTML, CSS and js

Academic Year: 2024

Software Requirements Specification for Chess Game Applications

1. Introduction

This document outlines the software requirements for a web-based Chess Game application. The application aims to provide an interactive platform for users to play chess online.

2. System Overview

The Chess Game is a web-based application that simulates a chessboard and allows users to play chess. It includes features such as turn indication, game reset functionality, piece movement, and win condition detection.

3. Functional Requirements

3.1 User Interface

- The application shall display a chessboard with a traditional 8x8 grid layout.
- The interface shall include a title "MASTER CHESS" at the top of the page.
- The application shall display a turn indicator showing which player's turn it is.
- The interface shall include a reset button to restart the game.
- The application shall display game status messages (e.g., check, checkmate).

3.2 Chessboard

- The chessboard shall be represented as an HTML unordered list (<ul>).
- Each square on the board shall be represented as a list item (<li>) with a unique ID.
- The board shall be divided into 8 rows, each containing 8 squares.
- Each square shall have a unique identifier in the format "bXYZ" where X is the row number and YZ is the column number.
- The chessboard shall be coloured in a checkered pattern.

3.3 Chess Pieces

- The initial layout of chess pieces shall be correctly set up on the board.

- Black pieces shall be placed on rows 7 and 8.

- White pieces shall be placed on rows 1 and 2.

- The correct initial placement of each piece type (pawn, rook, knight, bishop, queen, king) shall be implemented.

- Chess pieces shall be represented by both text and images on the board.

3.4 Game Play

- The application shall allow players to make moves by clicking on pieces and valid destination squares.

- The application shall track and display whose turn it is (White or Black).

- The application shall prevent players from moving pieces when it's not their turn.

- The application shall handle piece movement and capture logic.

- The application shall prevent invalid moves (e.g., moving into check, moving through other pieces).

3.5 Game State Management

- The application shall keep track of the current game state, including piece positions and the current player's turn.

- The application shall detect and handle special chess situations such as check and checkmate.

3.6 Game Reset

- The application shall provide a reset functionality to restart the game, accessible via a "Reset" button.

3.7 Win Conditions

- The application shall detect when a player is in check.

- The application shall detect when a player is in checkmate, ending the game.

- The application shall display appropriate messages for check and checkmate situations.

4. Non-Functional Requirements

4.1 Performance

- The web application shall load quickly and respond to user interactions without significant delay.

- The game logic shall execute efficiently to ensure smooth gameplay.

4.2 Usability

- The user interface shall be intuitive and easy to understand for chess players.

- The chessboard layout shall be clear and visually distinguishable.

- Chess pieces shall be easily identifiable through both text and images.

4.3 Compatibility

- The application shall be compatible with modern web browsers.

- The application shall be responsive and accessible on both desktop and mobile devices.

4.4 Maintainability

- The HTML structure shall be well-organized and easy to maintain.

- The JavaScript code shall be modular and well-commented for easy maintenance and future enhancements.

- The application shall use external CSS (style.css) for easy styling modifications.

5. Future Enhancements

- Implement a more sophisticated move validation system for each piece type.

- Add support for special chess moves (e.g., castling, en passant, pawn promotion).

- Implement a chess clock to time each player's moves.

- Add support for online multiplayer functionality.

- Implement a system to record and replay games.

- Add an AI opponent for single-player mode.

- Improve the visual design of the chessboard and pieces.

6. Constraints

- The current implementation uses client-side technologies (HTML, CSS, JavaScript) and may lack server-side processing for advanced features.

- The win condition detection (check and checkmate) is currently a placeholder and needs to be fully implemented.

- The application does not currently support all special chess rules and moves.

This comprehensive SRS provides a foundation for the Chess Game website, covering both the HTML structure and JavaScript functionality. It can be expanded upon as the project develops and new features are added.