Chess Editor

January 22, 2021

Devin M. O'Brien
Preston Williamson
Brandon Kyle
Dakota Simpkins
Tyler Wallschleger

1 Project Definition

1.1 Purpose

2 Project Requirements

2.1 Functional Requirments

2.1.1 Primary Requirements

The primary requirements of the project must be functional at 'all times'. These requirements include the following:

1. User can play a game of chess:
   - Locally (vs Player)
   - vs Computer
2. User can create, load, and play chess scenarios and variants like:
   - Chess960
   - 3-Check
   - King of The Hill [1]
   - Horde
3. The user is able to add a chess engine[1].

2.1.2 Secondary Requirements

The secondary requirements of the project may not always be function and are dependent on specific situation. These requirements include the following:

1. User can play a game of chess remotely.
2. User can use program to analyze moves.

2.2 Usability

2.2.1 User Interface

The user interface must not be obtrusive and intuitive. Visuals need to be properly contrasted for differentiation.

2.2.2 Performance

The program must be able to perform without noticable performance drop in all situations with the exceptional of analysis and secondary features.

[1] Whether the chess engine works is dependent on the engine. However at least two public chess engines must work with the application.

2.3 System

2.3.1 Hardware

The required hardware is a standard personal computer with a monitor display.

2.3.2 Software

Required Software The required software includes:

- Graphviz: Reason

Operating System This program must be able to on both Windows 10 and Linux (specifically OpenSuse 15.2).

2.3.3 Database

(POSTPONED)

2.3.4 Networking

Certain secondary functionality requires a network connection in order to work properly.

2.4 Security

There are currently no security requirements with the exception of optional password protection for saved games and hosted games.

3 Project Specification

3.1 Scope

3.2 Libraries/ Frameworks/ Development Environments/ Dependencies

3.3 Platform

3.4 Genre

4 System - Design Perspective

4.1 Subsystems

4.1.1 Models

5 System - Analysis Perspective

5.1 Subsystems

6 Project Scrum Report

6.1 Product Backlog

6.2 Sprint Backlog

6.3 Burndown Chart

7 Subsystems

7.1 Subsystem 1

7.1.1 Initial Design and Models

7.1.2 Data dictionary

7.1.3 Revisions

7.1.4 Scrum Backlog

7.1.5 Coding

Approach

Language

7.1.6 User Training

7.1.7 Testing

8 Complete System

9 References

References

[1] Chess.com, “Chess variants - the 6 most important ones.”