SP-14 Build Chess Game Using AI

SOFTWARE REQUIREMENTS SPECIFICATION (SRS)

4850 SECTION 2

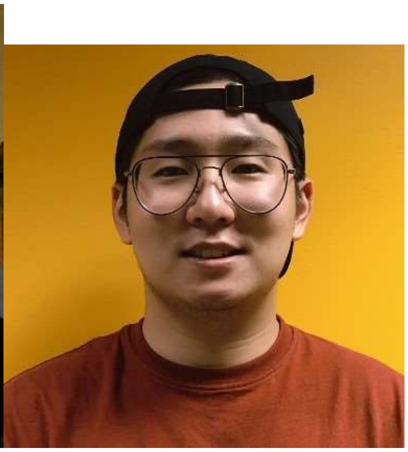
Spring Semester

Professor Perry

2023

Aaron Dailey

Nicholas Kennel



Haige Zhu

Contents

1.0 Introduction

The following project is a chess game that allows the user to play against a Ai on a Windows computer made from C# , Ai algorithms, and runs on the Unity engine.

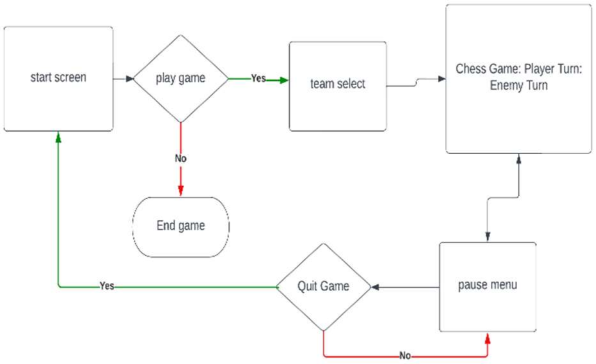
1.1 Overview

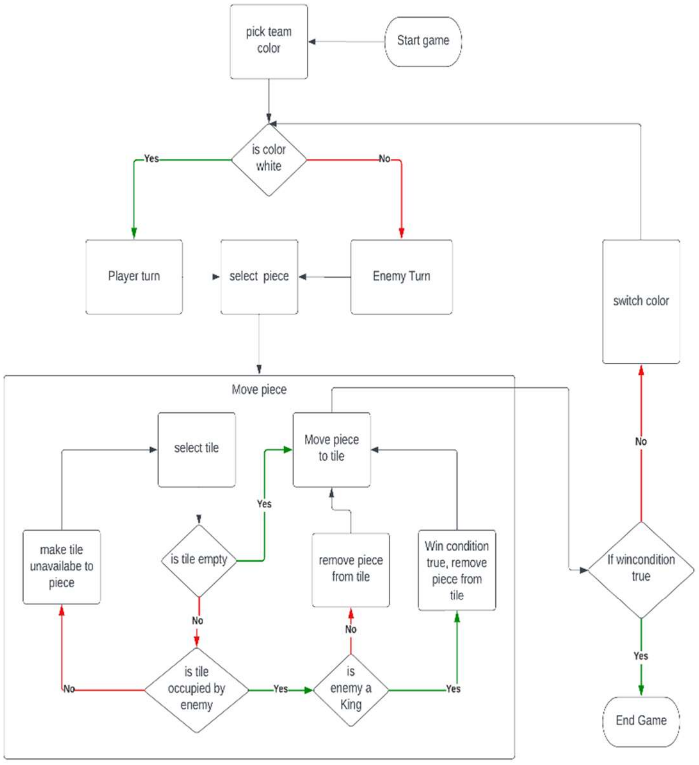
The project is an AI chess game where the user will play against a CPU chess player.

1.2 Project Goals

Goal of project is for the user to play a chess game with Ai.

2.0 Design Constraints





3.0 Requirements

3.1 User Requirements

The game begins with a start screen where it would allows you to do 2 options, one start game which will bring up a canvas, or a different scene which in that case lets you pick which starting color you want. The starting color is going to be the player while the opposite color is going to be the AI. From there on, it's going to go to the main chess game scene where the player can play the AI. The chess game scene is going to consist of some canvases for the UI, chess pieces, and the chess board. It's going to be a basic UI that lets you be able to just click on a chess piece, then it's going to enable you to see where that piece can move as well as where it can move and be vulnerable to enemy pieces.

After that you move a chess piece, it'll automatically switch from the player's turn to the AI's turn, and then the AI's turn will then select a chess piece and then move their chess piece. It will switch back and forth until it gets to a point where one of the chess pieces will take another chess piece. When a piece takes another chess piece it would replace that piece, or just plain delete the enemy's chess piece. Then that will continue until you take the enemy(AI) king or until the enemy either take your king and at that point, it will change a win condition through either you or the AI to true. Once the king is taken, instead of going to the enemy's turn or your turn it will, then just go to the victory screen or defeat screen.

Finally, you could choose to replay the game again.

3.2 AI Requirements

We use the training data as chess games and rules. Our goal for the AI part is at least for the AI to be able to move the piece by the logic and the rules, while AI plays with the user. For example, if the player the user uses white and he makes a move with the pawn to E4, then the AI will use the minimax algorithm to evaluate the optimal solution based on the user's moves, and then the Alpha-beta algorithm will optimize the calculation time, trying to optimize the time to within a few seconds

Deep learning and data sets to train AI to achieve AI to play against people. It can set the difficulty. The AI part is to calculate the possibility of winning each step through the Alpha -Beta pruning algorithm and reduce the calculation time through algorithm optimization. The Minmax algorithm is used to evaluate the optimal solution based on the user's moves.

4.0 External Interface Requirements

4.1 User Interface Requirements

An interactive chess piece as well as available chess movement, pause screen, start screen, and a victory/lose menu.

5.2 Hardware Interface Requirements

A computer that has x64 architecture for its cpu, and gpu that has DX10, DX11, and DX12-capable GPUs.

5.3 Software Interface Requirements

Required software need is Windows 7 (SP1+) or Windows 10, 64-bit versions or Windows 11.