Checkers Requirements Document

Tim Adams, Chris Lucas, Tyler Sancibrian, Tony Burke

Version 1.0

Table of Contents

Introduction

Checkers is a classic board game that has been played for thousands of years. More recently, it was added to Vermont Technical College's Knight's Party Table, although without consistent rules. The following document details the required rules and regulations necessary for Checkers to function properly on the Knight's Party Table.

Key Concepts

The basic concepts of Checkers is simple - a board game comprised of twenty-four total pieces, twelve for each player, on a checkerboard. A player's objective is to remove their opponents pieces from the board by jumping the other pieces. While Checkers is already mostly programmed on the Knight's Party Table, the rules are not. The following must be programmed:

Enforcements for the following rules

- Players cannot move their pieces on adjacent squares, only diagonal.
- Players can only move forward.
- When a player jumps another piece, that piece is removed from the board.
- When a player moves his piece to the opposite side of the board, that piece is "kinged"
    - Kinged pieces can move backwards, but not adjacent.

Use Cases

Starting a Game

Goal

The user will successfully start a game of Checkers.

Other Resources Needed

No other resources are needed to start a game of Checkers.

User Action

After starting the Knight’s Party Table, the user will select “Checkers” from the list of games shown to them. The second user will then select

Product Action

The product will display the list of available games to the user. After the user selects “Checkers” from the list, the product will display a confirmation window to the other player.

Playing a Game

Goal

The users will play a game of Checkers, jumping each others pieces in an attempt to claim victory.

Other Resources Needed

No other resources are needed to play a game of Checkers.

User Action

The user will select a piece they wish to move, and then select a square that they wish to move that piece to. The second user will then do the same.

Product Action

The product will move the selected piece to the given location. If the given location violates the rules of Checkers, the product will not allow the user to move it to that location.

Finishing a Game

Goal

The player will either win or lose a game.

Other Resources Needed

No other resources are needed for a player to finish a game.

User Action

The user will either take the last of their opponent’s pieces, or they will lose their last piece to their opponent.

Product Action

When one of the sides has lost all of their pieces, the product will display three windows - one to the winner, one to the loser, and one to the public screen - with the appropriate message congratulating and consoling, as well as showing the victor to the crowd. The product will then ask the users whether or not they wish to play another game of Checkers, or go back to the game menu.

Functional Requirements

For further clarification, the following are requirements necessary for the Checkers game to function as designed:

- Two Players

Non-Functional Requirements

The following are non-functional requirements for the Checkers game:

Platform

Checkers will run on the Knight's Party Table. The Knight's Party Table will run the following:

Windows 10 Home Edition

Performance

Checkers will run within a reasonable amount of time, reasonable being dictated by the definition found in the project scope document.

Reliability

Any form of a system crash is unacceptable. Users are using the product to play Checkers, not constantly reboot the Knight's Party Table.