# King Me

Requirements Specification

---

| Group Members | Vincent Finn, Chris Deslongchamp, Matthew Quaschnick, William Brown |
|---|---|
| **Faculty Advisor** | Dr. Filippos Vokolos, Ph. D. |
| **Project Stakeholder** | Dr. Filippos Vokolos, Ph. D. |

## Revision History

| Name | Date | Reason for Change | Revision |
|------|------|-------------------|----------|
| Chris Deslongchamp | 7/17/2018 | First Draft – Sections Outlined | 0.9 |
| Chris Deslongchamp | 7/24/2018 | Second Draft – Content added to all sections | 1.0 |
| Will Brown, Vincent Finn, Chris Deslongchamp, Matthew Quaschnick | 7/30/2018 | Revisions – Updating info based on comments and adding images | 1.1 |

# Table of Contents

# Introduction

## Purpose of Document

This document will provide all of the requirements for the game King Me. It will serve as a reference for the developers and the customer for developing the final version of the game.

## Scope of Document

This document will contain enough information such that a developer will be able to easily translate the requirements to code without ambiguity.

## Overview of Document

This document will contain functional and nonfunctional requirements for the game King Me, as well as use cases and diagrams. The game contains two components that allow two players to play on their respective Windows machine that connect through Peer to Peer network sharing. The PC components will be referred to as **Host** and **Client** for the player that hosts the game and the player that joins the game. This document will also contain mock-ups for the graphical user interface and use cases.

# Description

## Product Perspective

King Me is an online two-player checkers game designed to be played on two computers across the internet. One player will host the game while the second player joins the host's game. Once both players have joined, one player will be randomly assigned as Player 1 and the other will be assigned Player 2. Afterwards, the checkers game will launch, and players will take turns making their moves within a certain time frame with Player 1 making the first move.
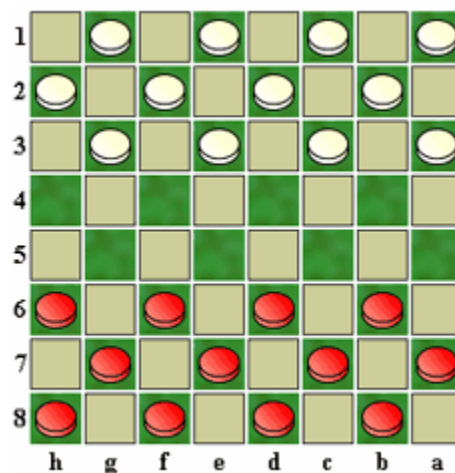


*Figure 1: Checkers Board*

The game is intended to run on two PCs, Player 1 and Player 2. Players will make their move on their own PC and that move will be translated to the other's PC through the server connection. Once the

move has been applied to both Player's screens, and that player will hand off the turn to his opponent. This will continue until a player has won or both players end up in a draw.

### Player PC Interface

The PC Interface will contain a start menu with options to host or join a game. Once connected the PC interface will display the game board, pieces, which player will go first, whose turn it is and a pause option.

## Product Functions

### Host Functionality

The Host will have the following functionality:

- Ability to host a game (through Peer to Peer networking)
- Ability to make a move that will be checked by a move validation function
- Ability to receive opponents move
- Ability to communicate move to opponent
- Ability to pause a game
- Ability to surrender
- Ability to win, tie, or lose a game

### Client Functionality

The Server will have the following functionality:

- Ability to join a game (through Peer to Peer networking)
- Ability to make a move that will be checked by a move validation function
- Ability to receive opponents move
- Ability to communicate move to opponent
- Ability to pause a game
- Ability to surrender
- Ability to win, tie, or lose a game

## User Description

The ideal users for King Me would be two people located across from each other in the same room. However, players do not have to be in the same room if they have an internet connection.

## Assumptions and Dependencies

### C#

C# is an object-oriented language that enables developers to build a variety of secure and robust applications that run on the .NET Framework. You can use C# to create Windows client applications,

XML Web services, distributed components, client-server applications, database applications and more. This program will be developed using C# version 7.3 using the .Net Core Framework. The project fully depends on this Language/Framework. If Microsoft stops development of C# or .Net the project will be halted and will have to be migrated over to a new Language such as Java.

### .NET GUI

The GUI will be developed using Visual Studio's Windows Form editor found when creating a Visual C# Windows Form Application. The GUI will utilize the art assets created for this project which are discussed later under the "Art Assets" section. The GUI is required as it is vital to the other systems working properly.

### C# Networking

C# contains multiple built in libraries to help with Peer to Peer connections. Specifically the System.Net namespace contains classes that provide support for creating applications that send and receive data. Online play is critical to the user experience, thus getting this system functioning is required.

### Art Assets

The various members of the team will create the art assets. These assets while not required to make the game functional, will make the game significantly easier and more enjoyable to play, thus are required to be completed. They will be created using Adobe Photoshop CC. These assets must be scheduled and completed on time as to not impede development of other systems.

## Requirements Apportioning

| Priority Level | Description |
|---|---|
| 1 | Priority 1 requirements are essential to the product and must be in the final build. These requirements must be tested and verified to ensure proper functionality. |
| 2 | Priority 2 requirements are not required for the final build, but will be provided if there is sufficient time. The system will be designed such that it is extendable to easily incorporate these requirements at a later time. |
| 3 | Priority 3 requirements are not required, and will not be considered in the design of the system. If sufficient time remains the requirement will be incorporated. |

# Functional Requirements

## Host and Client
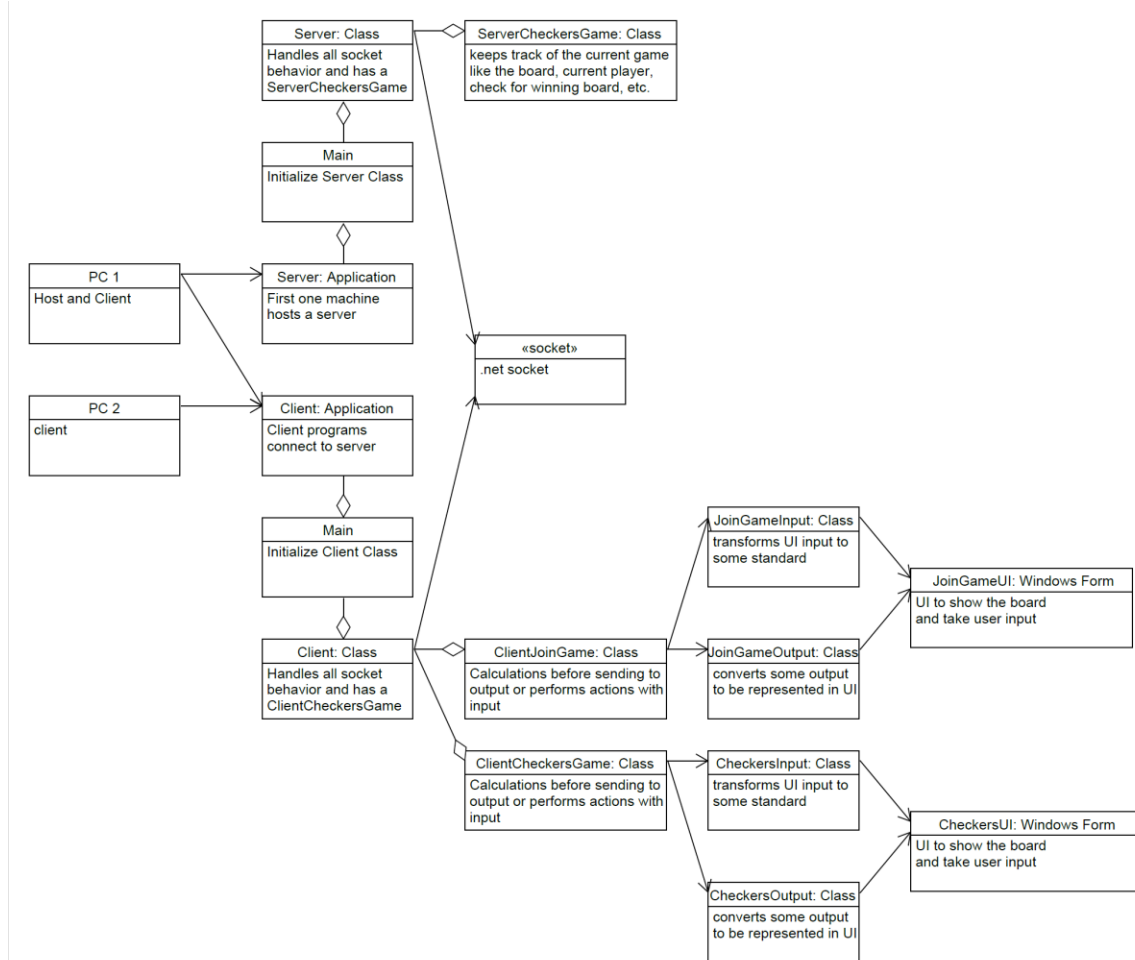


*Figure 2:KingMe Architecture*

### R1.1 Starting a Game

| Requirement | Title | Description | Priority |
|---|---|---|---|
| R1.1.1 | Host Game | Player will be able to choose to host a game. | 1 |
| R1.1.2 | Join Game | Player will be able to join a game. | 1 |
| R1.1.3 | How To Play | Button that when clicked, shows how to use the program to make a move and take their turn. | 2 |

## Running a Game

### R2.1 Loading a Game

| Requirement | Title | Description | Priority |
|---|---|---|---|
| R2.1.1 | Player Selection | Players are randomly assigned either Player 1 or Player 2. | 1 |

| R2.1.2 | Load Game Board | A green and white checkered board is displayed on both player's screens. The board will contain 64 squares alternating in color and be arranged in 8 rows and 8 columns. The square in the top left corner of the board will be white followed by a green square and continue the alteration for the rest of the board. | 1 |
|---|---|---|---|
| R2.1.3 | Load Player Tokens | 12 Red circle tokens and 12 White circle tokens will be placed on the board. The tokens will only be placed on green board squares. White will be placed in the last three rows while Red will be placed in the top 3 rows. Player 1 will be Red and Player 2 will be White. | 1 |
| R2.1.4 | Load Move Timer | A timer will be placed in the top right corner of the screen and will count down at the start of each players turn and will reset once a player has made their move. | 2 |
| R2.1.5 | Show Player's Id | On the bottom center of the screen Players will be alerted of which Player they have been assigned, either Player 1 or Player 2. | 1 |
| R2.1.6 | Show Current Move Player | Above the Move Timer, a message will show which Player's turn it is. | 1 |

## R2.2 On Player's Turn

| Requirement | Title | Description | Priority |
|---|---|---|---|
| R2.2.1 | Move Selection | On the Player's turn, they will be able to select a token they want to move and where they want to move it. | 1 |
| R2.2.2 | Move is checked if it is a legal move | Once a Player has submitted a move, the move is checked to make sure it is allowed. | 1 |
| R2.2.3 | Move is legal | If the move is legal, the move will be made on the player's screen. The timer will stop and reset. The other player will be shown the move and their turn will begin. | 1 |
| R2.2.4 | Move is illegal | If the move is illegal, the move will not be made. The current turn's player will be alerted to choose a different move. The timer will continue to count down. | 1 |
| R2.2.5 | Pause the game | The player can request a pause. | 3 |
| R2.2.6 | Alert of Start of Turn | The player will be alerted that their turn has begun. | 2 |
| R2.2.7 | Surrender | Player can choose to surrender. | 3 |
| R2.2.8 | Timer | Timer counts down on player's turn. If the timer reaches zero, the player loses the game or turn. | 2 |

## R2.3 Off-Turn Player

| Requirement | Title | Description | Priority |
|---|---|---|---|
| R2.3.1 | Other Player has made a legal move | The board will update with the other player's move. | 1 |

| R2.3.2 | Accept or Decline Pause Request | The player will be alerted that the other player would like to pause the game. They will be allowed to accept or decline the pause. | 3 |

## R2.4 Paused Game

| Requirement | Title | Description | Priority |
|---|---|---|---|
| R2.4.1 | Pause Message Displayed | A message will display stating that the game is paused and by which player. | 3 |
| R2.4.2 | Request Game Resume | A button will be displayed to allow a player to request a game to resume. Once a request is made the other player can choose to accept the resume. | 3 |

## R2.5 Game Over

| Requirement | Title | Description | Priority |
|---|---|---|---|
| R2.5.1 | Player has won | Once a player has met the win condition, both players will be alerted, and the winning player will be displayed. | 1 |
| R2.5.2 | Game is a tie | Once a tie has occurred, both players will be alerted of the tie. | 1 |
| R2.5.3 | Player has surrendered | Once a player surrenders, both players will be alerted, and the non-surrendering player will be declared the winner. | 3 |
| R2.5.4 | Exit Game | Once one of the end game messages has been displayed, players can choose to exit the game. | 1 |
| R2.5.5 | Rematch | Once one of the end game messages has been displayed, players can choose to start a new game with each other. They can choose to keep the same turn order or switch turn order. Player 1 would become Player 2 and vice versa. | 2 |
| R2.5.6 | New Game | Once one of the end game messages has been displayed, a player can choose to start a new game. Doing so will send them to the start menu again where they can host or join a game. | 1 |

# Non-Functional Requirements

## Network Connection

### R3.1.1 Player Connection

Players should remain connected to each other the entirety of the game. There should be no delay from when a player submits a move, the move is confirmed to be legal, and the legal move is made on both player's screen. This will be a constant priority for the team. **Priority 1**

### R3.1.2 Player Disconnection

If a player disconnects during a game, a timer will start for the player to reconnect. If the player reconnects during the time limit, the game will resume. If the player fails to rejoin during the time limit, the other player will be declared the winner. **Priority 2**

## Host Operating System Requirements

### R4.1 OS Support

The game expected to support Windows 7,8, and 10. The game may support Mac OS 10.X and above. Other Operating systems will not be tested, and Mac OS is not guaranteed to be supported. **Priority 1**

## Accessibility

### R5.1 Game Download

The game will be available for download from a GitHub link that is public. **Priority 1**

### R5.2 Small File Size

The file size for the game will be kept as small as possible to decrease download times, preferable under 100 megabytes. **Priority 2**

## Playtesting

### R6.1 Playtesting

During alpha and beta iterations, testing will be done internally to make sure functions run correctly. Once the program has reached version 1 and higher, playtesting will be done by the team as well as some outside players (6-10 additional people). **Priority 1**

# User Interface

## Start Screen

| Requirement | Title | Description | Priority |
|---|---|---|---|
| R7.1.1 | Host Game Button | Button to host a game. | 1 |
| R7.1.2 | Join Game Button | Button to join a game being hosted. | 1 |
| R7.1.3 | How-To Play Button | Button to open the how to play menu. | 2 |
| R7.1.4 | How-To Play Menu | Menu pop-up that explains how players make turns and move pieces. | 2 |

## Game Screen

| Requirement | Title | Description | Priority |
|---|---|---|---|
| R7.2.1 | Game Board | A green and white checkered board is displayed on both player's screens. The board will contain 64 squares alternating in color and be arranged in 8 rows and 8 columns. The square in the top left corner of the board will be white | 1 |

| Requirement | Title | Description | Priority |
|---|---|---|---|
| | | followed by a green square and continue the alteration for the rest of the board. The board will be placed in the left center of the game screen. | |
| R7.2.2 | Player Tokens | 12 Red circle tokens and 12 White circle tokens will be placed on the board. The tokens will only be placed on green board squares. White will be placed in the last three rows while Red will be placed in the top 3 rows. Player 1 will be Red and Player 2 will be White. | 1 |
| R7.2.3 | Move Timer | A timer will be placed in the top right corner of the program. | 2 |
| R7.2.4 | Current Player's Turn | Above the timer it will say, "Player # Turn" and the font color will be the player's token color. | 1 |
| R7.2.5 | Player's Id | The player's token color will be displayed in the bottom right corner. | 1 |
| R7.2.6 | Player Color Notification | Open joining and loading a game, a menu will display alerting the player of their token color and whether they go first or not. | 1 |
| R7.2.7 | Pause Button | Above the player's id color and surrender button, a pause button will placed. | 2 |
| R7.2.8 | Surrender Button | Below the pause button but above the player's Id will be a surrender button. | 3 |
| R7.2.9 | Pause Request Menu | A window will appear whenever a player requests a pause. If the player requested the pause the window will say awaiting pause approval. If the player didn't request the pause, the window will give them an option to accept or decline the pause. If they accept the window will change to say game is paused, with a resume play button. If the pause is rejected, the window will disappear. | 2 |

## Game Over Screen

| Requirement | Title | Description | Priority |
|---|---|---|---|
| R7.3.1 | Game Over Screen | A window will appear alerting the players of the results of the game. (winner, tie, surrender, etc.) | 1 |
| R7.3.2 | Exit button | On the bottom, right, center of the window will be an exit option. | 1 |
| R7.3.3 | Rematch | On the bottom, left, center of the window will be a rematch button. | 2 |
| R7.3.4 | New Game | On the bottom center of the window will be a new game button. | 1 |

## How-To Screen

| Requirement | Title | Description | Priority |
|---|---|---|---|
| R7.4.1 | How-To Play instructions | Detailed instructions will inform the player of how to make a move, how to host, or join a game, and any other tips for how to use the program. | 2 |

| | | | | |
|---|---|---|---|---|
| R7.4.2 | Exit window | An exit button will be placed at the bottom of the window. | 2 | |

## Use Cases

| Use Case | Title | Precondition | Action | PostCondition |
|---|---|---|---|---|
| R8.1.1 | Host a Game | A player wants to host a checkers game. | Player clicks the host button. | Player becomes a game host and another player may join their game. |
| R8.1.2 | Join a Game | A player wants to join a checkers game. | Player clicks the join game button. Player is shown games to join. Player selects a game to join. | Player joins the game and the game loads. |
| R8.1.3 | Make a legal move | A player wants to move a piece from tile 3,1 to empty tile 4,2. | Player selects the token in tile 3,1 and then selects the empty tile 4,2. | Move is checked if it is legal. If it is legal the move is made and displayed on both players' screens. |
| R8.1.4 | Make an illegal move | A player wants to move a piece from tile 3,1 to enemy player tile 4,2. | Player selects their token in tile 3,1 and then selects the enemy player tile 4,2. | Move is checked if it is legal. Since it is illegal, the player is alerted that they must choose a different move. |
| R8.1.5 | Wait out the timer | A player hasn't made a move yet | A player waits for the clock to hit 0 | Player forfeits his turn, and the turn is passed to the opponent. |
| R8.1.6 | Pause a game | A player wants to pause a game. | Player selects the pause game button. | A window opens alerting the player's that a pause has been requested. |
| R8.1.7 | Pause rejected | A player wants to reject a request for a pause. | Player selects the reject pause button. | The pause window closes and the game resumes. |
| R8.1.8 | Pause accepted | A player wants to accept a request for a pause. | Player selects the accept button. | The pause window updates to say the game is in a pause. |
| R8.1.9 | Resume Play | A player wants to resume play. | Player selects the resume play button. | The pause window updates saying a player wants to resume the game. Once both players select resume, a count down will display. Once the countdown reaches 0, the window will close and the game will resume. |

| R8.1.10 | Surrender | A player wants to forfeit or quit the game. | Player selects the surrender button. | The game ends and the surrendering player loses the game. The game over screen appears and displays the results. |
|---|---|---|---|---|
| R8.1.11 | Player wins the game | Player captures all enemy tokens. | Player jumps the enemy's final token. | The game ends and the game over screen displays the results. The player wins. |
| R8.1.12 | Player wins the game | Player forces opponent to make the same move 3 times. | Opponent makes the same move for a third time. | The game ends and the game over screen displays the results. The player wins. |
| R8.1.13 | Player ties the game | Both players make the same move 3 times. | Opponent or player makes the same move for a third time. | The game ends and the game over screen displays the results. The players tie. |
| R8.1.14 | Player makes a jump | Player wants to jump an enemy token. | Player selects their token and then selects the tile diagonal to them and the enemy token they want to jump. | The move is checked that it is legal. The jump is made and the enemy token is captured and disappears from the board. |
| R8.1.15 | Player wants to exit game over screen. | Player is done playing after the game ends. | Player clicks the exit button on the game over screen. | The game closes. The opponents is alerted the player left and cannot rematch. |
| R.8.1.16 | Player wants to rematch | Player wants to play the same player again | Player clicks the rematch button. | Other player is alerted of a rematch request. Once the other player accepts the request a new game will start. If the other player exits, the rematch player is sent to the start menu. |
| R8.1.17 | Player wants to play a new game | Player wants to play a different opponent. | Player clicks the new game button. | Player is taken to the start menu where they can choose to host or join a new game. |