

King Me

Software Test Case Document

| | |
|----------------------------|---|
| 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 | 8/17/2018 | First Draft – Sections Outlined | 0.9 |
| Chris Deslongchamp | 8/20/2018 | Second Draft – Content added to all sections | 1.0 |
| William Brown, Vincent Finn, Matthew Quaschnick | 8/21/2018 | Updated with comments and small tweaks. | 1.1 |

Table of Contents

| | |
|--|----|
| Revision History | 2 |
| 1. Introduction | 4 |
| 1.1 Definitions, Acronyms and Abbreviations..... | 4 |
| 1.2 References | 4 |
| 2. Testing Enviroments | 4 |
| 2.1 Environment 1: Windows 10..... | 4 |
| 3. Setup Information and Prerequisites..... | 4 |
| 4. Test Cases..... | 4 |
| 4.1 Test Cases 1: Initialize Game..... | 4 |
| 4.1.1 Description | 4 |
| 4.1.2 Additional Prerequisites..... | 5 |
| 4.1.3 Scenario..... | 5 |
| 4.2 Test Cases 2: Player's Turn..... | 6 |
| 4.2.1 Description | 6 |
| 4.2.2 Additional Prerequisites..... | 6 |
| 4.2.3 Scenario..... | 6 |
| 4.3 Test Cases 3: Opponent's Turn | 9 |
| 4.3.1 Description | 9 |
| 4.3.2 Additional Prerequisites..... | 9 |
| 4.3.3 Scenario..... | 10 |
| 4.4 Test Cases 1: Paused Game..... | 10 |
| 4.4.1 Description | 10 |
| 4.4.2 Additional Prerequisites..... | 11 |
| 4.4.3 Scenario..... | 11 |
| 4.5 Test Cases 5: Game Over | 11 |
| 4.5.1 Description | 11 |
| 4.5.2 Additional Prerequisites..... | 12 |
| 4.5.3 Scenario..... | 12 |
| 5. Appendix | 14 |

| | |
|-------------------|----|
| 5.1 Glossary..... | 14 |
|-------------------|----|

1. Introduction

The purpose of this document is to describe the testing approaches used while evaluation the functionality and performance of King Me program as to meet the requirements outlined in the requirement document. King Me is a C# application that allows two users to play a game of checkers from remote locations via connecting to a server.

1.1 Definitions, Acronyms and Abbreviations

Please refer to the Appendix and Glossary sections for any definitions and abbreviations.

1.2 References

The document may feature terms and references which can be found on The American Checker Federation website.

2. Testing Environments

The program and associated test cases have been run within the following test environments.

2.1 Environment 1: Windows 10

| | | | |
|----------------|---|----------------------|--------------|
| Machine Name | Windows PC | DB Directory | N/A |
| OS and Version | Windows 10.0.17134, 8 GB RAM, 256 GB SSD | Interpreter Platform | C# .exe file |
| Tester Name | Chris Deslongchamp | Test Date | TBA |
| New Log | | State | N/A |

3. Setup Information and Prerequisites

Prior to running the program, the following prerequisites must be met.

- The program and associated test cases can be run by launching the program. The program features a GUI that will handle all interactions. No interactions will be done through a console or terminal.
- Two instances of the application must be running to perform all test cases successfully. Both instances must be running by the same person so that they may see the results on both players' screens.
- An internet connection is required for optimum performance of the program.

4. Test Cases

4.1 Test Cases 1: Initialize Game

4.1.1 Description

The case consists of covering the steps required to host or join a game on a computer.

4.1.2 Additional Prerequisites

None.

4.1.3 Scenario

| ID | Req | Priority | Description | Execution Steps | Expected Results | Actual Results | Comments |
|----|-------------------------------------|----------|-------------------------|---|---|----------------|----------|
| A1 | R1.1.1 | 1 | Host a game | 1. Click the Host Game Button | GUI window changes to show the hosting game menu. The users IP address is shown. Message states waiting for opponent to join. | | |
| A2 | R1.1.2 | 1 | Join a game | 1. Click the Join Game button 2. Enter Host IP Address in IP Address Text box and press enter. | GUI window changes to allow user to enter to enter the Host IP Address. Once submitted the GUI window changes to show the user has connected. | | |
| A3 | R1.1.3 | 3 | How to play menu shows | 1. Click the how to play button | GUI Window changes to the How to Play menu, showing the user how to make moves and use the program. | | |
| A4 | R1.1.4 | 3 | How to play menu closes | 1. Click the back button on the how to play menu. | GUI Window changes back to the main menu screen. | | |
| A5 | R1.1.1, 1.1.2, 2.1.1, 2.1.2, 2.1.3, | 1 | Game Starts | 1. Once both players have joined the game starts. | GUI Window changes to a game board with timer in top right corner | | |

| | | | | | | | |
|--|-------|--|--|--|--|--|--|
| | 2.1.4 | | | | and tokens (red in every other square in the first three rows and white in every other square in the last three rows). Players are assigned turns (Player 1 and Player 2). Players are alerted who's turn it is first. | | |
|--|-------|--|--|--|--|--|--|

4.2 Test Cases 2: Player's Turn

4.2.1 Description

The case consists of covering the steps of actions you may perform on your turn.

4.2.2 Additional Prerequisites

It must be your turn to run any of these test cases.

4.2.3 Scenario

| ID | Req | Priority | Description | Execution Steps | Expected Results | Actual Results | Comments |
|----|------------------------|----------|---------------------------------|--|---|----------------|----------|
| B1 | R2.1.5 | 2 | Player's ID is shown. | 1. Locate the Player's ID in the bottom center of the screen. | The player's ID should reflect if they got the first move or second by saying Player 1 or Player 2. | | |
| B2 | R2.1.6 | 2 | Current Player's Turn is shown. | 1. Locate the Current Player's Turn above the timer in the top right corner. | Your player id should be shown above the timer in the top right corner. | | |
| B3 | R2.2.1, R2.2.2, R2.2.3 | 1 | Player makes a legal move. | 1. Click on one of your tokens. 2. Click on an empty square (below your | Token will move to the empty square diagonal to its original position and | | |

| | | | | | | | |
|----|---|---|--|--|---|--|--|
| | | | | token on the left or right if you are red or above your token on the left or right if you are white.) | your turn will end. | | |
| B4 | R2.2.1, R2.2.2, R2.2.3 | 1 | Player makes an illegal move | <ol style="list-style-type: none"> 1. Click on one of your tokens. 2. Click on an opponent token. | Token will not move, and you are prompted to make another move selection. | | |
| B5 | R2.2.1, R2.2.2, R2.2.3, R2.2.4 | 1 | Player makes an illegal move | <ol style="list-style-type: none"> 1. Click on one of your tokens. 2. Click on an empty square not diagonal to your token. | Token will not move, and you are prompted to make another move selection. | | |
| B6 | R2.2.1, R2.2.2, R2.2.3, R2.2.4 | 1 | Player makes an illegal move | <ol style="list-style-type: none"> 1. Click on one of your tokens. 2. Click on another one of your tokens. | Token will not move, and you are prompted to make another move selection. | | |
| B7 | R2.2.1, R2.2.2, R2.2.3 | 1 | Player makes a legal move with a king. | <ol style="list-style-type: none"> 1. Click on one of your king tokens. 2. Click on an empty diagonal square either above or below the king. | King will move to empty square you selected, and your turn ends. | | |
| B8 | R2.2.1, R2.2.2, R2.2.3 | 1 | Player gets a king token. | <ol style="list-style-type: none"> 1. Click on a token one row away from the opponents back row. | Token is transformed into a king and your turn ends | | |

| | | | | | | | |
|-----|------------------------------|---|----------------------------|--|---|--|--|
| | | | | 2. Move the token into the opponents back row. | | | |
| B9 | R2.2.1, R2.2.2, R2.2.3 | 1 | Player makes a legal jump. | <ol style="list-style-type: none"> 1. Select a token that is next to an opponent's token. 2. Select the empty square on the other side of the token (must be diagonal to the player and opponent tokens.) | Token will jump over opponent token, opponent token that was jumped is removed from the board, and player's turn ends. | | |
| B10 | R2.2.1, R2.2.2, R2.2.3 | 1 | Player makes a multi jump | <ol style="list-style-type: none"> 1. Select a token that is next to an opponent's token. 2. Select the last square in the multi jump. (The square must be diagonal to an opponent's token and have an empty square diagonal to that opponent that the player could land in form another jumps.) | The token is moved to the last square in the jump, all opponent tokens that were jumped are removed from the board, and the player's turn ends. | | |

| | | | | | | | |
|-----|----------------|---|--|---|--|--|--|
| B11 | R2.2.5 | 2 | Player request a pause. | 1. Click the pause game button. | A pause request is sent to the opponent, the game stops the timer, and the GUI updates saying a pause has been requested. | | |
| B12 | R2.2.5, R2.4.1 | 2 | Player request is accepted | 1. Wait for opponent to accept the pause. | The game is paused, and the paused game menu appears. | | |
| B13 | R2.2.5 | 2 | Player request is rejected | 1. Wait for opponent to reject the pause request. | Your turn resumes, and the pause request menu disappears. | | |
| B14 | R2.2.6, R2.2.8 | 2 | Player is alerted their turn has begun | 1. Wait for your opponent to finish their turn. | A window will appear saying your turn will start. A count down will start. When the countdown ends, the window disappears, the move timer resumes counting down, and the player can click the board and buttons. | | |

4.3 Test Cases 3: Opponent's Turn

4.3.1 Description

The case consists of covering the steps of actions you can perform on your opponents' turn.

4.3.2 Additional Prerequisites

It must not be your turn to run these test cases.

4.3.3 Scenario

| ID | Req | Priority | Description | Execution Steps | Expected Results | Actual Results | Comments |
|----|----------------|----------|---|---|--|----------------|----------|
| C1 | R2.3.1 | 1 | Opponent makes a legal move. | 1. Wait for your opponent to make a legal move. | Your opponent's move should be reflected on your screen. | | |
| C2 | R2.3.2, R2.4.1 | 2 | Opponent requests a pause and it is accepted. | 1. Wait for your opponent to request a pause. 2. When the pause request menu appears, click the accept button. | The game is paused, and the paused game menu appears. | | |
| C3 | R2.3.2 | 2 | Opponent requests a pause and it is rejected | 1. Wait for your opponent to request a pause. 2. When the pause request menu appears, click the reject button. | The game resumes and the pause request menu disappear. | | |
| C4 | R2.2.1 | 1 | Cannot make a move or select a button. | 1. Try to move one of your tokens. 2. Try to select the pause or surrender buttons. | You are unable to move any tokens or click any buttons. | | |

4.4 Test Cases 1: Paused Game

4.4.1 Description

The case consists of covering the steps of action you may take during a pause.

4.4.2 Additional Prerequisites

The game must be paused to run these test cases.

4.4.3 Scenario

| ID | Req | Priority | Description | Execution Steps | Expected Results | Actual Results | Comments |
|----|--------|----------|---------------------------------|---|---|----------------|----------|
| D1 | R2.4.2 | 2 | Resume the game. | 1. Click the resume game button. | The pause menu updates stating that a resume game request has been sent. | | |
| D2 | R2.4.2 | 2 | Accept the resume game request | 1. Wait for opponent to request a resume game. 2. Click accept once the Pause game menu updates to show a resume has been requested. | The pause menu disappears and the game resumes for the player who's turn it was. | | |
| D3 | R2.4.2 | 2 | Reject the resume game request. | 1. Wait for opponent to request a resume game. 2. Click accept once the Pause game menu updates to show a resume has been requested. | The resume game request disappears from the pause menu and the game stays paused. | | |

4.5 Test Cases 5: Game Over

4.5.1 Description

The case consists of covering the steps to end a game and what actions you can take once a game has ended.

4.5.2 Additional Prerequisites

None.

4.5.3 Scenario

| ID | Req | Priority | Description | Execution Steps | Expected Results | Actual Results | Comments |
|----|-------------------|----------|---|---|--|----------------|----------|
| E1 | R2.2.7, R2.5.3 | 3 | Game over due to surrender. | <ol style="list-style-type: none">Click the surrender button.Click the yes on the are you sure you want to surrender menu. | The game ends. Both players are taken to the game over screen and it states which your opponent won, and you surrendered. | | |
| E2 | R2.2.8 | 3 | Game over due to failure to make a move. | <ol style="list-style-type: none">Let your move timer reach 0. | The game ends. Both players are taken to the game over screen and it states which the opponent won, and you failed to make a move within the time limit. | | |
| E3 | R2.5.1 | 1 | Game over due to capturing all opponent tokens. | <ol style="list-style-type: none">Jump over the last token your opponent has. | The game ends. Both players are taken to the game over screen and it states you won by capturing all your opponent's tokens. | | |
| E4 | R2.5.1 | 1 | Game over due to same move being made 3 times. | <ol style="list-style-type: none">Move the same piece back and forth between the same two squares three times. | The game ends. Both players are taken to the game over screen and it states your opponent won by forcing you | | |

| | | | | | | | |
|-----|--------|---|--|---|--|--|--|
| | | | | 2. Have your opponent make different moves. | to make the same move. | | |
| E5 | R2.5.2 | 3 | Game over due to a tie from both players making the same move three times. | 1. Both you and your opponent must move the same piece back and forth between the same two squares three times. | The game ends. Both players are taken to the game over screen and it states the game ended in a tie. | | |
| E6 | R2.5.4 | 1 | Exit the game | 1. Click the exit game button on the game over screen. | The application closes. | | |
| E7 | R2.5.5 | 1 | Request a rematch. | 1. Click the rematch button. | The game over screen updates stating that you have requested a rematch. | | |
| E8 | R2.5.5 | 2 | Rematch accepted. | 1. Click the accept button once your opponent requests a rematch. | The game starts over, and a new game board is shown. (See test case A5) | | |
| E9 | R2.5.5 | 2 | Rematch rejected. | 1. Click the reject button once your opponent requests a rematch. | The rematch request disappears from the game over screen. | | |
| E10 | R2.5.6 | 2 | Find a new game. | 1. Click the new game button. | The game over screen is replaced by the start menu. Player can | | |

| | | | | | | | |
|--|--|--|--|--|--------------------------------|--|--|
| | | | | | either host or join a game. | | |
|--|--|--|--|--|--------------------------------|--|--|

5. Appendix

5.1 Glossary

Token – Another name for a piece or checker.

Opponent – Another name for the opposing player.

Host – The player hosting the game. The player who the other player connects to.

Player 1 – The player who moves first and controls the red tokens.

Player 2 – The player who moves second and controls the white tokens.

King – A token who has reach the opposing sides back row and can now move in any direction.