

Team Name/Project Name	
Component	Requirements
Document	Requirements
Responsible	QA Leader
Version:1.0	Date:

# Software Requirements Document

## 1. System Description

A program that will allow users to play checkers either locally or remotely.

## 2. User Profile

User: any person at the local machine

Connected User: The person and his remote machine.

Either of the actors can be referred to as a player

## 3. Nonfunctional requirements<sup>1</sup>

### a) Portability

Since the program is written in Java it should be platform independent and should be able to run on any kinds of systems with the JVM.

### b) Maintainability

The program will be well documented so that changes will be made easily. Also the program will use abstraction to allow for easy modifications and expansion.

### c) Robustness

The program will include extensive error checking in the conventions provided by Java.

### d) Performance

The program will limit how much information needs to be sent over network. The code will be written cleanly so that overhead is minimized.

### e) Ease of installation

It will be a self-extracting file. JVM needs to be installed. Other than that this does not require any additional installations.

### f) Data Integrity

There should not be any data that would be saved from session to session. No data should be lost.

### g) Scalability

Abstraction allows the expansion in the new areas of the checkers world.

---

<sup>1</sup> Taken from: Use Cases. Requirementst in Context. D. Kulak, E. Guiney. Addison Wesley 2000

#### 4. User Interface requirements

- a) Visualization constraints
- b) Interaction constraints

#### 5. Design Constraints

- a) Layered architectural style
- b) Operation Platform
- c) Tools
- d) Object oriented technology

#### 6. Implementation Constraints

- a) Language
- b) Libraries
- c) Implementation platform

#### 7. Functional Requirements

<b>Identifier:</b> R 1	<b>Needed/Wanted:</b> Needed	<b>Priority:</b> High
<b>Use case Name<sup>2</sup>:</b> Move King		
<b>Summary:</b> Moving a king.		
<b>Basic course of events:</b> <ul style="list-style-type: none"><li>• User: selects piece</li><li>• User: selects where he wants the piece to go</li><li>• System: verifies that it is a legal move</li><li>• System: moves piece if it is legitimate.</li></ul>		
<b>Alternative paths:</b> The king falls back to the original placement if the move was not valid		
<b>Extension points:</b> Capturing the piece with the king.		
<b>Pre - Conditions:</b> <ul style="list-style-type: none"><li>• The piece selected is a king</li><li>• The piece selected is the correct player's king</li></ul>		
<b>Post- Conditions:</b> <ul style="list-style-type: none"><li>• The piece has been moved to the correct location</li></ul>		

<b>Author:</b>
<b>Date:</b>

<b>Identifier:</b> R 2	<b>Needed/Wanted:</b> Needed	<b>Priority</b> High
<b>Use case Name<sup>3</sup>:</b> King Capture		
<b>Summary:</b> A king captures one of the other players pieces		
<b>Basic course of events:</b> <ul style="list-style-type: none"> <li>• User: selects the piece</li> <li>• User: selects the piece he wants to capture</li> <li>• System: verifies that it is a legal move</li> <li>• System: If legal the piece is captured and the king is moved to the new spot</li> </ul>		
<b>Alternative paths:</b> If illegal the game is returned to the state it was in before the move was attempted		
<b>Extension points:</b> Extension of moving a king		
<b>Pre - Conditions:</b> <ul style="list-style-type: none"> <li>• The selected piece is a king</li> <li>• The piece selected is correct player's king</li> <li>• Capture has to be made</li> </ul>		
<b>Post- Conditions:</b> <ul style="list-style-type: none"> <li>• The piece is captured</li> <li>• The king is moved</li> </ul>		
<b>Author:</b>		
<b>Date:</b>		

<b>Identifier:</b> R3	<b>Needed/Wanted:</b> Needed for smooth exit from checkers match	<b>Priority</b> Medium - Not needed immediately but will improve user experience.
-----------------------	--	---

<b>Use case Name<sup>4</sup>:</b> Network Resign.	
<b>Summary:</b>	This use case covers all the ways for the local user to be informed of the connected user being removed from the game. The connected user can resign from the game voluntarily, by quitting the game, or closing the client. The connected user can also be resigned by an interruption in the network connection, or invalid commands being sent to the host.
<b>Basic course of events:</b>	<ul style="list-style-type: none"> <li>• Connected User: Connected user resigns from the game.</li> <li>• Connected User: Message is sent to local program</li> <li>• System: Informs local user that other player has left</li> <li>• System: Ends the local game</li> </ul>
<b>Alternative paths:</b>	<ul style="list-style-type: none"> <li>• Something interferes with the connection</li> <li>• System: The connection times out</li> <li>• System: Inform local user that connection was lost</li> <li>• System: Ends the local game</li> </ul>
<b>Extension points:</b>	No Extension points
<b>Pre - Conditions:</b>	Network Connection has been established and the game has started
<b>Post- Conditions:</b>	The program has quit and connected user is disconnected, network connection is terminated, the game has ended.
<b>Author:</b>	
<b>Date:</b>	

<b>Identifier:</b> R 4	<b>Needed/Wanted:</b> needed	<b>Priority</b> High – integral part of network play
<b>Use case Name<sup>5</sup>:</b> Connect to Host		
<b>Summary:</b>	Create a connection to a user specified Checkers host. Setup the connection and any preliminary information needed. Once all setting have been agreed on start a new game.	

<b>Basic course of events:</b>	<ul style="list-style-type: none"> <li>• User: prompt system to enter a remote host location</li> <li>• System: Contact remote host</li> <li>• Connected User: Send a player name over</li> <li>• Connected User: start new game</li> </ul>
<b>Alternative paths:</b>	If any of the steps fail the User will get a message informing him/her the connection failed.
<b>Extension points:</b>	None.
<b>Pre - Conditions:</b>	No previous Network Connections, any previous game has ended.
<b>Post- Conditions:</b>	A network game is running, if the Basic Course of events is followed, otherwise the User needs to establish a new connection.
<b>Author:</b>	
<b>Date:</b>	

<b>Identifier:</b> R5	<b>Needed/Wanted:</b> This functional requirement is wanted by the user.	<b>Priority</b> The priority for this functional requirement is very low because draw is not offered a lot through out the game and it does not change the course of the game drastically.
<b>Use case Name<sup>6</sup>:</b> Draw.		
<b>Summary:</b>	This use case allows any user playing the current game to offer a draw at any time during the game. The other user can accept or decline that offer. If the offer is accepted the game ends with a draw. If the offer is declined the game continues.	
<b>Basic course of events:</b>	<ul style="list-style-type: none"> <li>• Player 1: Presses a draw button.</li> <li>• System: Sends the draw offer to other player (Player 2)</li> <li>• Player 2: Accepts the offer to declare a draw</li> <li>• System: Sends the answer back to Player 1</li> <li>• System: Player 1 is notified</li> <li>• System: Game is ended</li> </ul>	
<b>Alternative paths:</b>	<ul style="list-style-type: none"> <li>• Player 2: declines the offer for a draw.</li> <li>• System: Sends the answer back to Player 1</li> <li>• System: Player 1 is notified</li> <li>• System: Game continues</li> </ul>	

<b>Extension points:</b>	This use case is connected to the ending of the game.
<b>Pre - Conditions:</b>	A game is in progress.
<b>Post- Conditions:</b>	Game has ended in a draw.
<b>Author:</b>	
<b>Date:</b>	

<b>Identifier:</b> R6	<b>Needed/Wanted:</b> This functional requirement is needed by the user.	<b>Priority</b> The functional requirement for this use case has a high priority because user should be able to quit the game any time he wants to.
<b>Use case Name</b> <sup>7</sup> : User/Host exits the game.		
<b>Summary:</b> This use case allows the user to exit the game at any time. The user should be able to exit the game when he is playing it on the single system or when he is hosting the network game, which in turn should disconnect the user on the other side.		
<b>Basic course of events:</b> <ul style="list-style-type: none"> <li>• User: closes program or resigns the game</li> <li>• System: Inform both players the game has ended due to resignation, and the winner is declared.</li> <li>• System: Disconnects the network user if it is a network game</li> <li>• System: Game ends.</li> </ul>		
<b>Alternative paths:</b> None		
<b>Extension points:</b> The loss of network connection scenario is covered under the Network Resign use case.		
<b>Pre - Conditions:</b> User must be in the game in order to quit it.		
<b>Post- Conditions:</b> Game has ended.		
<b>Author:</b>		
<b>Date:</b>		

<b>Identifier:</b> R7	<b>Needed/Wanted:</b> This function is wanted by the user for the proper execution of the program.	<b>Priority:</b> High
<b>Use case Name</b> <sup>8</sup> : Move Piece		

<b>Summary:</b>	This use case deals with the actions involved when the user moves one piece from one square on the board to another.
<b>Basic course of events:</b>	<ul style="list-style-type: none"> <li>• User selects single piece it would like to move by clicking on that square.</li> <li>• User selects the square to move the piece to and clicks on that square.</li> <li>• The system validates it and checks to see if it's legal.</li> <li>• If it is legal, the system allows the piece to be moved there and updates the state of the program.</li> </ul>
<b>Alternative paths:</b>	If illegal, system returns to its previous state. (Before move was attempted)
<b>Extension points:</b>	There may be a possible extension between this and capturing pieces, since that case involves making a move.
<b>Pre - Conditions:</b>	It is the users turn to move.
<b>Post- Conditions:</b>	The user will have made a move validated by the controller. The state of the system will have been updated.
<b>Author:</b>	
<b>Date:</b>	

<b>Identifier:</b> R8	<b>Needed/Wanted:</b> Needed by the user.	<b>Priority:</b> High
<b>Use case Name<sup>9</sup>:</b> Capture Piece		
<b>Summary:</b>	This use case deals with the actions involved when a player tries to jump a piece belonging to the opposing player.	
<b>Basic course of events:</b>	<ul style="list-style-type: none"> <li>• User selects single piece it would like to move.</li> <li>• User selects the square to move the piece to.</li> <li>• The system validates it and checks to see if it's legal.</li> <li>• If it is legal, the system allows the piece to be moved there and updates the state of the program.</li> </ul>	
<b>Alternative paths:</b>	If illegal, system returns to its previous state. (Before move was attempted)	
<b>Extension points:</b>	This is an extension of move a single piece.	
<b>Pre - Conditions:</b>	It is the user's turn to move and the game is in the correct current state. The correct pieces have been removed from the board.	
<b>Post- Conditions:</b>	The user will have made a move validated by the controller. The state of the system will have been updated.	
<b>Author:</b>		
<b>Date:</b>		

<b>Identifier:</b> R9	<b>Needed/Wanted:</b> Needed	<b>Priority:</b> High
<b>Use case Name</b> <sup>10</sup> : Setup Options		
<b>Summary:</b>	The user must use the setup to properly establish preferences and conditions for the game.	
<b>Basic course of events:</b>	<ul style="list-style-type: none"> <li>• User: Selects desired settings.</li> <li>• User: Confirms settings.</li> <li>• System: Puts in place user selected preferences.</li> </ul>	
<b>Alternative paths:</b>	User exits the program before setting all the options	
<b>Extension points:</b>	None	
<b>Pre - Conditions:</b>	Checkers game is loaded but the playable game has not yet begun.	
<b>Post- Conditions:</b>	Settings are in place. Actual game ready to begin.	
<b>Author:</b>		
<b>Date:</b>		

<b>Identifier:</b> R10	<b>Needed/Wanted:</b> Needed	<b>Priority:</b> High
<b>Use case Name</b> <sup>11</sup> : Start Game		
<b>Summary:</b>	Allow the users to start playing the game after they have set up the options.	
<b>Basic course of events:</b>	<ul style="list-style-type: none"> <li>• System: Display GUI objects.</li> <li>• System: Game begins.</li> </ul>	
<b>Alternative paths:</b>	None.	
<b>Extension points:</b>	None.	
<b>Pre - Conditions:</b>	Setup was completed.	
<b>Post- Conditions:</b>	The game board is displayed. The red player is free to make the first move	
<b>Author:</b>		
<b>Date:</b>		

<b>Identifier:</b> R11	<b>Needed/Wanted:</b> Needed	<b>Priority:</b> Medium
<b>Use case Name</b> <sup>12</sup> : Host Game		
<b>Summary:</b>	Waits for connection from remote user then begins the game.	
<b>Basic course of events:</b>	<ul style="list-style-type: none"> <li>• System: Wait for connection.</li> <li>• System: Exchange information with remote user.</li> </ul>	



• System: Start game.	
Alternative paths:	Host quits game.
Extension points:	Possible extension of setup.
Pre - Conditions:	Setup completed and host networking selected.
Post- Conditions:	Network game has begun.
Author:	
Date:	