TEST CASES DOCUMENT

for

Checkers

Version 1.0.0

Prepared by:
John Zlotek
Matt Horger
Jake Carfagno
Preet Patel

Team: Big Chungus

August 14, 2019

Contents

Revision History

Name	Date	Reason For Changes	Version
1.0.0	12–08–19	Initial Draft	pp534

1 Introduction

1.1 Purpose of Document

The purpose of this document to demonstrate that the Checkers game design and functionality aligns with the requirements outlined in the requirement document. This document includes testing of all possible scenarios by defining acceptation and noting the actual outcome of the case.

1.2 Definitions, Acronyms and Abbreviations

If any then it would be mention in Appendix and Glossary sections

1.3 Refrences

All though required references are can be found in the preceding document but any new reference will be noted under reference section.

2 Testing Environment

This section has brief information about the environment where the test was performed and the information of the tester.

2.1 Environment 1

Machine Name		Database Directory			
OS/ Version		Interpreter Platform		Client Server	
Tester Name			Test Date		
Comment				State	

3 Testing Environment Setup and Prerequisites

Prerequisit

- This game has GUI for interaction and for that at least Java 12.0 is needed on the system

- Just for testing purpose it can be test on single machine but for crossplatform Internet is needed

- Active Internet connection so it player can connet to server and play over Tux

4 Test Cases

This section is divided into four sections where it tests the game into four phases, the first phase include the launching phase of the game, second phase test the validation of gameplay, third small phase test the functionality of multiple session of game and the last phase of testing includes the end of the game.

4.1 Test Phase 1: Starting Game

4.1.1 Description

This section covers the testing of the initial phase of launching a game, creating a session and starting a new game.

4.1.2 Prerequisites for this test case

An internet connection regardless of testing between multiple entities, or single in order to connect to server.

4.1.3 Scenario

Number	Case	Expectation	Actual Outcome	Steps	Requrienment Refrence
1	Launch the game	GUI of Lobby menu should be launched			
2	Create a session	Create a session ID			R4.1
3	Join new game	Once lobby selected from loby menu, player should be able to join that session			R4.2
4	Create multiple session Max 5	No more then 10 session should be allowed			L2.2
5	Start 5 game succeffuly with 10 player	There should be 10 gameplay going on a same time on a server			L2.3
6	Start 5 game succeffuly with 11 player	21th player should not able to join any session.			L2.3
7	After starting game with two player	GUI should have 8X8 board with randomly assigned color			G1

4.2 Test Phase 2: Gameplay validation

4.2.1 Description

This section covers the testing of the gameplay validation, it covers the test of moves which are valid, moves are invalid, valid jump, invalid jump and such.

4.2.2 Prerequisites for this test case

A valid game session running - single session at least.

4.2.3 Scenario

Number	Case	Expectation	Actual Outcome	Steps	Requrienment Refrence
1	Black piece first move	Only player with black piece should make move			G2
2	White piece invalid first move	If Player with white piece makes the move then it should be discarded and the piece should come back to the original position			G2
3	Valid Move - Non crowned Piece	Can move diagonally in either direction and peice stays in new square			G5
4	Valid single jump to capture - Non crowned Piece	Only jump and capture the opponent piece if there is an empty square above that targeted piece.			G5

Number	Case	Expectation	Actual Outcome	Steps	Requrienment Refrence
5	Valid Multiple jump to capture - Non crowned Piece	If jump made over multiple piece of opponent considering test case 4 then it should be valid.			G5
6	Invalid Move - Non crowned Piece	If piece move in horizontal or vertical direction then it should come back to original position			G5
7	Invalid Move - Non crowned Piece	If tried move backward diagonally then it should be discarded and it should come back to original position			G5
8	Invalid single jump to capture - Non crowned Piece	if there is not space to place the piece after the jump to capture then it should come back to original position			G5
9	Invalid single jump to capture - Non crowned Piece	If jump made over players own piece then			G5

Number	Case	Expectation	Actual Outcome	Steps	Requrienment Refrence
10	Turning into crown piece	If either player piece reaches to the 10th row from player direction then, piece should turn into a crowned piece			G6
11	Valid Move - Crowned Piece	It can move in any four diagonal direction			G6
12	Valid single jump to capture - Crowned Piece	It can jump in any direction considering the test case 4 is valid with each jump			G6
13	Valid Multiple jump to capture - Crowned Piece	it can jump in any direction considering the test case 12 is valid with each jump			G6
14	Invalid Move - Crowned Piece	If moved in horizontal or vertical then piece should return in original direction			G6
15	Invalid single jump to capture - Crowned Piece	If jumped is made over own piece then it discards it and peice should return in original direction			G6
16	If crowned piece placed in the 10th row again	The crown piece state should not be affected in any way	12		G6

4.3 Test Phase 3: Multiple

4.3.1 Description

This section covers the testing of the proper game performance of multiple session running the same time.

4.3.2 Prerequisites for this test case

More than one game session running to test this.

4.3.3 Scenario

Number	Case	Expectation	Actual Outcome	Steps	Requrienment Refrence
1	Session activity only between two connected player	Only changes should be done and seen between two connected player, it should not interfier other session game play			1.1

4.4 Test Phase 4: Ending the game

4.4.1 Description

This section covers the testing of the scenario when the game ends and possibilities of further action.

4.4.2 Prerequisites for this test case

Atleast one game running

4.4.3 Scenario

Number	Case	Expectation	Actual Outcome	Steps	Requrienment Refrence
1	Winning Jump	If the valid jump is made over the last piece of the opponet then player should be decalred winner			6.2.1.1
2	Winning Player Screen	Player should be informed with the message			6.2.1.1
3	Lossing Player Screen	Player should be informed with the message			6.2.1.1
4	Player 1 wants to quit	Once player 1 click on quit it should send him back to lobby page			6.2.1.2
5	Player 1 left the game	Player 2 will be informed that he/she won the game and send back to lobby page			6.2.1.2
6	After game is fineshed if player 1 choose to rematch and Player 2 also do the same	If Player 2 also click on the rematch then new game should get going on same session			6.2.1.2
7	After game is fineshed if player 1 choose to rematch and Player 2 quit	if player 2 click on the quit then both player should be returned to lobby menu			6.2.1.2

15

5 Appendix

5.1 Glossary

Tux - Place where server is hosted, Tux is offered by Drexel University.

Bibliography

[1] The American Checker Foundation, *USA Checkers*, https://www.usacheckers.com/, 2019.

[2] W.J. Rayment, *History of Checkers or Draughts*, http://www.indepthinfo.com/checkers/history.shtml, 2004.