JSTanks - Test Report

Jiahao Li	Pavithran Pathmarajah	Viren Patel
LI577	PATHMAP	PATELVH3
Student #	001410729	Student #

October 31, 2016

Contents

1 Revision History

1.1 Revision 0

Date	Developer	Change
October 31	Jiahao Li	Initial Draft
October 31	Pavithran Pathmarajah	Initial Draft
October 31	Viren Patel	Initial Draft

2 General Information

2.1 Purpose

2.2 Scope

2.3 Acronyms, Abbreviations, and Symbols

2.4 Overview of Document

3 Plan

3.1 Software Description

JSTanks is a game which allows users enjoy it on a website without downloading it. This game let the user control a tank to fire and move on the map in order to protect its home base and itself from the damage of other tanks'.

3.2 Test Team

The test team will consist of Jiahao Li, Pavithran Pathmarajah, Viren Patel and some random players as testers. Jiahao Li, Pavithran Pathmarajah, Viren Patel will split the entire testing which cover all different types of tests. The random player will test the performance of the game and give feedbacks to the development team.

3.3 Automated Testing Approach

3.4 Testing Tools

3.5 Testing Schedule

4 System Test Description

4.1 Tests for Functional Requirements

4.1.1 HTML file test

Name: Creating new browser by html file
Description: Test if the executable HTML file creates a new browser window.
Type: Unit test (dynamic, manual, black box)
Initial State: A html file in the operating system
Input: Click the html file
Output: A new browser
Pass: A new browser pops up.

4.1.2 Standby state test

Name: Standby state
Description: Test if the game has a standby state in which it waits for user input.
Type: Unit test (static, automated, black box)
Initial State: A new browser
Input: –
Output: The standby state of the game
Pass:The new browser with the game menu in it.

4.1.3 Menu test

Name: Menu representation
Description: est if the menu with four sections which are game, pause/continue, level and introduction shows up in the standby state.
Type: Unit test (dynamic, manual, black box)
Initial State: A new browser
Input: –
Output: The standby state of the game
Pass: The menu with four sections is represented in the standby state.

4.1.4 The game section of the menu test

Name: Menu of game section
Description: Test if the sub menu of game section which has choices of starting a new game and quit shows up when the game section is clicked.
Type: Unit test (dynamic, manual, black box)
Initial State: Menu in the standby state
Input: Click the game section of the menu
Output: The sub menu
Pass:The sub menu with choice of starting a new game and quit.

4.1.5 The pause section of the menu test

Name: Menu of pause and continue
Description: Test if the sub menu of pause/continue section which has choices of pause and continue shows up when the pause/continue section is clicked.
Type: Unit test (dynamic, manual, black box)
Initial State: Menu in the standby state
Input: Click the pause/continue section of the menu
Output: The sub menu
Pass: The sub menu with choice of pause and continue.

4.1.6 The level section of the menu test

Name: Menu of level
Description: Test if the sub menu of level section which has choices of level 1, level 2, and level 3 shows up when the level section is clicked.
Type: Unit test (dynamic, manual, black box)
Initial State: Menu in the standby state
Input: Clickthe level section of the menu
Output: The sub menu
Pass: The sub menu with choice of level1, level 2 and level 3.

4.1.7 Game start test

Name: Start the game
Description: Test if the game shall be reset and start when [starting a new game] is clicked.
Type: Unit test (dynamic, manual, black box)
Initial State: The sub menu of the game section
Input: Click the choice of starting a new game
Output: The standby state of the game
Pass: The standby state of the game with all objects on their initial position on

the map.

4.1.8 Game quit test

Name: Quit the game
Description: Test if the game comes to the end state that the GUI turns into black with only the menu on it when the choice of quit is clicked.
Type: Unit test (dynamic, manual, black box)
Initial State: The running state of the game
Input: Click the choice of quit
Output: the end state of the game
Pass: The game comes to the end state that the GUI turns into black with only the menu on it.

4.1.9 Game pause test

Name: Pause the game
Description: Test if the game comes to the pause state when the choice of pause is clicked.
Type: Unit test (dynamic, manual, black box)
Initial State: The running state of the game
Input: Click the choice of pause
Output: The pause state of the game
Pass: The game comes to the pause state that all stuff in the game freeze and stay in the temporal positions.

4.1.10 Game continue test 0

Name: Continue the game
Description: Test if the game comes back to the running state from the pause state when the choice of continue is clicked in the pause state.
Type: Unit test (dynamic, manual, black box)
Initial State: The pause state of the game
Input: Click the choice of continue
Output: The running state of the game
Pass: All stuff frozen in the pause state are activated and back into the routine.

4.1.11 Game continue test 1

Name: Continue the game in the running state
Description: Test if there is any effect on the game when the choice of continue is clicked in the running state.
Type: Unit test (dynamic, manual, black box)

Initial State: The running state of the game
Input: Click the choice of continue
Output: The running state of the game
Pass: No effect on the running state.

4.1.12 AI test

Name: The routine of AI e
Description: Test if the AI controls tanks to move and fire randomly.
Type: Unit test (dynamic, automated, white box)
Initial State: The running state of the game
Input: –
Output: The routine of tanks controlled by the AI
Pass: The tanks controlled by the AI move and fire randomly.

4.1.13 Level test

Name: Levels of the game
Description: Test if the moving speed of tanks controlled by the AI change when level 1, level 2 or level 3 is clicked.
Type: Unit test (dynamic, automated, black box)
Initial State: The running state of the game
Input: Click the choice of different level
Output: The different speed of movements of tanks controlled by the AI
Pass: The higher the level is, the higher the speed of movements of tanks controlled by the AI is.

4.1.14 Default level test

Name: The default level of the game
Description: Test if the level 1 is chosen as the default speed of tanks controlled by the AI when the game starts.
Type: Unit test (static, automated, white box)
Initial State: Thethe standby state of the game
Input: Click the choice of starting a new game
Output: The game starts with tanks controlled by the AI moving in the lowest speed
Pass: The game starts with tanks controlled by the AI moving in the lowest speed.

4.1.15 Information test

Name: The information of the game
Description: Test if The window with the information of the game in it pops up when the section of introduction is clicked.
Type: Unit test (dynamic, manual, black box)
Initial State: The menu with four sections
Input: Click the choice of different level
Output: The window with information in it
Pass: The window with information in it pops up.

4.1.16 Movement test

Name: The movement of the tank controlled by the user
Description: Test if the tank controlled by the user moves left, right, up or down when the left, right, up or down key on the keyboard is pressed.
Type: Unit test (dynamic, manual, black box)
Initial State: The running state of the game
Input: Press the left, right, up or down key on the keyboard
Output: The movement of the tank controlled by the user
Pass: The tank moves to the correct direction one step according to the key pressed by the user every time.

4.1.17 Continuous movement test

Name: The continuous movement of the tank controlled by the user
Description: Test if the tank controlled by the user keeps moving in the direction of left, right, up or down when the left, right, up or down key on the keyboard is held.
Type: Unit test (dynamic, manual, black box)
Initial State: The running state of the game
Input: Hold the left, right, up or down key on the keyboard
Output: The continuous movement of the tank controlled by the user
Pass: The tank keeps moving in the correct direction according to the key held by the user until the user release the key.

4.1.18 Bullet launch test

Name: Launch the bullet
Description: Test if the tank controlled by the user launches a bullet when the user press the space on the keyboard.
Type: Unit test (dynamic, manual, black box)
Initial State: The running state of the game
Input: Press the space on the keyboard

Output: The tank controlled by the user launches a bullet in the direction it faces to
Pass: The tank controlled by the user launches a bullet in the direction it faces to.

4.1.19 Bullet movement test

Name: The movement the bullet
Description: Test if the bullet keep moving in the same direction after being launched.
Type: Unit test (dynamic, automated, black box)
Initial State: The bullet is launched
Input: –
Output: The bullet keep moving in the same direction which it launched to
Pass: The bullet keep moving in the same direction which it launched to.

4.1.20 Bullet disappearance test

Name: The bullet disappearance
Description: Test if the bullet disappears when it hits the tank, wall, steel, home base or the boundary of the map.
Type: Unit test (dynamic, automated, black box)
Initial State: the bullet hits a wall. a steel, the home base of the boundary of the map
Input: –
Output: The bullet disappear
Pass: The bullet disappears when it hits the tank, wall, steel, home base or the boundary of the map.

4.1.21 Wall hit test

Name: The wall hit by the bullet
Description: Test if the wall disappears when it is hit by the bullet.
Type: Unit test (dynamic, automated, black box)
Initial State: The wall is hit by the bullet
Input: –
Output: The wall disappears
Pass: The wall disappears immediately when it is hit by the bullet.

4.1.22 Steel hit test 0

Name: The steel hit by the bullet twice
Description: Test if the steel stays the same when it is hit by the bullet twice.

Type: Unit test (dynamic, automated, black box)
Initial State: The steel is hit by the bullet
Input: –
Output: The steel stays the same
Pass: The steel stays the same when it is hit by the bullet twice.

4.1.23 Steel hit test 1

Name: the steel hit by the bullet at the third time
Description: Test if the wall disappears when it is hit by the bullet at the third time.
Type: Unit test (dynamic, automated, black box)
Initial State: The steel is hit by the bullet
Input: –
Output: The steel disappears
Pass: The steel disappears when it is hit by the bullet at the third time.

4.1.24 Enemy tanks hit test

Name: The tank controlled by the AI hit by the bullet
Description: Test if the tank controlled by the AI disappears when it is hit by the bullet.
Type: Unit test (dynamic, automated, black box)
Initial State: The tank controlled by the AI is hit by the bullet
Input: –
Output: The tank controlled by the AI disappears
Pass: The tank controlled by the AI disappears when it is hit by the bullet.

4.1.25 Home base hit test 0

Name: The home base hit by the bullet for first four times
Description: Test if the home base stays the same when it is hit by the bullet for first four times.
Type: Unit test (dynamic, automated, black box)
Initial State: The home base is hit by the bullet
Input: –
Output: The home base stays the same
Pass: The home base stays the same when it is hit by the bullet for first four times.

4.1.26 Home base hit test 1

Name: The home base hit by the bullet at the fifth time
Description: Test if the home base disappears when it is hit by the bullet at the fifth time.
Type: Unit test (dynamic, automated, black box)
Initial State: The home base is hit by the bullet
Input: –
Output: The home base disappears
Pass: The home base disappears when it is hit by the bullet at the fifth time.

4.1.27 User tank hit test 0

Name: The tank controlled by the user hit by the bullet for first four times
Description: Test if the tank controlled by the user stays the same when it is hit by the bullet for first four times,.
Type: Unit test (dynamic, automated, black box)
Initial State: The tank controlled by the user is hit by the bullet
Input: –
Output: The tank controlled by the user stays the same
Pass: The tank controlled by the user stays the same when it is hit by the bullet for first four times.

4.1.28 User tank hit test 1

Name: The tank controlled by the user hit by the bullet at the fifth time
Description: Test if the tank controlled by the user disappears when it is hit by the bullet at the fifth time.
Type: Unit test (dynamic, automated, black box)
Initial State: The tank controlled by the user is hit by the bullet
Input: –
Output: The tank controlled by the user disappears
Pass: The tank controlled by the user disappears when it is hit by the bullet at the fifth time.

4.1.29 Game over test 0

Name: the tank is destroyed
Description: Test if the game comes to the end state when the tank controlled by the user disappears.
Type: Unit test (dynamic, automated, black box)
Initial State: The tank controlled by the user disappears
Input: –
Output: The end state of the game

Pass: The game comes to the end state that the GUI turns into black with only the menu on it.

4.1.30 Game over test 1

Name: The home base is destroyed
Description: Test if the game comes to the end state when the home base disappears.
Type: Unit test (dynamic, automated, black box)
Initial State: The home base disappears
Input: –
Output: The end state of the game
Pass: The game comes to the end state that the GUI turns into black with only the menu on it.

4.2 Tests for Nonfunctional Requirements

4.2.1 Area of Testing1

4.2.2 Area of Testing2

5 Tests for Proof of Concept

5.1 Area of Testing1

5.2 Area of Testing2

6 Comparison to Existing Implementation

7 Unit Testing Plan

7.1 Unit testing for internal functions

7.1.1 Wall type test

Name: Ask for the type of the wall
Description: Test if the program return the type of the wall when you ask for it.
Type: Unit test (dynamic, automated, white box)
Initial State: The object of wall is created
Input: wall.type()
Output: BARRIER
Pass: The program return the type BARRIER when we call the wall.type() function.

7.1.2 Steel type test

Name: Ask for the type of the steel
Description: Test if the program return the type of the steel when you ask for it.
Type: Unit test (dynamic, automated, white box)
Initial State: The object of steel is created
Input: steel.type()
Output: BARRIER
Pass: The program return the type BARRIER when we call the wall.type() function.

7.1.3 Home base type test

Name: Ask for the type of the home base
Description: Test if the program return the type of the hoem base when you ask for it.
Type: Unit test (dynamic, automated, white box)
Initial State: The object of home base is created
Input: homebase.type()
Output: BARRIER
Pass: The program return the type BARRIER when we call the wall.type() function.

7.1.4 Wall draw test

Name: Draw the wall on the game board
Description: Test if the image of the wall shows up on the position we set on the game board in the right size when we call this function.
Type: Unit test (dynamic, automated, white box)
Initial State: The game board with no image on the position (startX,startY)
Input: wall.draw(canvas,startX,startY,tileSize,t)
Output: The image of the wall shows up on the position (startX,startY) of the game board
Pass: The image of the wall shows up on the position (startX,startY) of the game board in the tileSize.

7.1.5 Steel draw test

Name: Draw the steel on the game board
Description: Test if the image of the steel shows up on the position we set on the game board in the right size when we call this function.
Type: Unit test (dynamic, automated, white box)
Initial State: The game board with no image on the position (startX,startY)

Input: steel.draw(canvas,startX,startY,tileSize,t)
Output: The image of the steel shows up on the position (startX,startY) of the game board
Pass: The image of the steel shows up on the position (startX,startY) of the game board in the tileSize.

7.1.6 Home base draw test

Name: Draw the home base on the game board
Description: Test if the image of the home base shows up on the position we set on the game board in the right size when we call this function.
Type: Unit test (dynamic, automated, white box)
Initial State: The game board with no image on the position (startX,startY)
Input: homebase.draw(canvas,startX,startY,tileSize,t)
Output: The image of the home base shows up on the position (startX,startY) of the game board
Pass: The image of the home base shows up on the position (startX,startY) of the game board in the tileSize.

7.1.7 Wall hit test

Name: Hit the wall
Description: Test if the program decreases the points of strength of the wall after the wall is hit.
Type: Unit test (dynamic, automated, white box)
Initial State: The remaining points of strength of the wall
Input: wall.hit()
Output: –
Pass: One point of strength of the wall is decreased after the wall is hit.

7.1.8 Steel hit test

Name: Hit the steel
Description: Test if the program decreases the points of strength of the steel after the steel is hit.
Type: Unit test (dynamic, automated, white box)
Initial State: The remaining points of strength of the steel
Input: steel.hit()
Output: –
Pass: One point of strength of the wall is decreased after the steel is hit.

7.1.9 Home base hit test

Name: Hit the home base
Description: Test if the program decreases the points of strength of the home base after the home base is hit.
Type: Unit test (dynamic, automated, white box)
Initial State: The remaining points of strength of the home base
Input: homebase.hit()
Output: –
Pass: One point of strength of the wall is decreased after the home base is hit.

7.1.10 Wall health get test

Name: Get health of the wall
Description: Test if the program return the remaining points of strength of the wall after calling this function.
Type: Unit test (dynamic, automated, white box)
Initial State: the object of wall
Input: wall.getHealth()
Output: The remaining points of strength of the wall
Pass: Return the remaining points of strength of the wall after calling this function.

7.1.11 Steel health get test

Name: Get health of the steel
Description: Test if the program return the remaining points of strength of the steel after calling this function.
Type: Unit test (dynamic, automated, white box)
Initial State: the object of steel
Input: steel.getHealth()
Output: The remaining points of strength of the steel
Pass: Return the remaining points of strength of the steel after calling this function.

7.1.12 Home base health get test

Name: Get health of the home base
Description: Test if the program return the remaining points of strength of the home base after calling this function.
Type: Unit test (dynamic, automated, white box)
Initial State: the object of home base
Input: homebase.getHealth()
Output: The remaining points of strength of the home base

Pass: Return the remaining points of strength of the home base after calling this function.

7.1.13 Wall position get test

Name: Get the position of the wall
Description: Test if the program return the remaining points of strength of the wall after calling this function.
Type: Unit test (dynamic, automated, white box)
Initial State: the object of wall has already be drawn
Input: wall.getPosition()
Output: The position of the wall
Pass: Return the position (x,y) of the wall after calling this function.

7.1.14 Steel position get test

Name: Get the position of the steel
Description: Test if the program return the remaining points of strength of the steel after calling this function.
Type: Unit test (dynamic, automated, white box)
Initial State: the object of steel has already be drawn
Input: steel.getPosition()
Output: The position of the steel
Pass: Return the position (x,y) of the steel after calling this function.

7.1.15 Home base position get test

Name: Get the position of the home base
Description: Test if the program return the remaining points of strength of the home base after calling this function.
Type: Unit test (dynamic, automated, white box)
Initial State: the object of home base has already be drawn
Input: homebase.getPosition()
Output: The position of the home base
Pass: Return the position (x,y) of the home base after calling this function.

7.2 Unit testing for output files

8 Appendix

8.1 Symbolic Parameters

8.2 Usability Survey Questions