

SE 3XA3: Test Plan TankWar

Team #212, Genius
Di Wu, 400117248, wud43
Jiahao Zhou, 400082351, zhouj56
Xinyu Huang, 400120376, huangx65

April 6, 2020

Contents

1	General Information	3
1.1	Purpose	3
1.2	Scope	3
1.3	Acronyms, Abbreviations, and Symbols	3
1.4	Overview of Document	4
2	Plan	4
2.1	Software Description	4
2.2	Test Team	4
2.3	Automated Testing Approach	5
2.4	Testing Tools	5
2.5	Testing Schedule	5
3	System Test Description	5
3.1	Tests for Functional Requirements	5
3.1.1	Area of Testing1 for Tank Movements	5
3.1.2	Area of Testing2 for Tank Attacking	19
3.1.3	Area of Testing3 for Tank Properties	22
3.1.4	Area of Testing4 for Gaming Process	24
3.1.5	Area of Testing5 for Map Editing	28
3.1.6	Area of Testing6 for Game Rules	37
3.2	Tests for Nonfunctional Requirements	53
3.2.1	Area of Testing1 for Look and Feel	53
3.2.2	Area of Testing2 for Usability and Humanity	57
3.2.3	Area of Testing3 for Performance	60
3.2.4	Area of Testing4 for Operational and Environmental	65
3.2.5	Area of Testing5 for Maintainability and Support	66
3.2.6	Area of Testing6 for Security	67
3.2.7	Area of Testing7 for Cultural	68
3.3	Traceability Between Test Cases and Requirements	69
4	Tests for Proof of Concept	72
4.1	Area of Testing1 for Map Saving	72
4.2	Area of Testing2 for the Test coverage	72
4.3	Area of Testing3 for the User Acceptance	73

5	Comparison to Existing Implementation	73
6	Unit Testing Plan	73
6.1	Unit testing of internal functions	73
6.2	Unit testing of output files	78
7	Appendix	79
7.1	Symbolic Parameters	79
7.2	Usability Survey Questions?	79

List of Tables

1	Revision History	2
2	Table of Abbreviations	3
3	Table of Definitions	4
4	Traceability Matrix for Functional Requirement	70
5	Traceability Matrix for Non-functional Requirement	71

List of Figures

This document describes the test plan for TankWar. The template for the test plan is from SE 3XA3 course.

Table 1: **Revision History**

Date	Version	Notes
23 February 2020	1.0	Create the first version of the Test Plan
28 February 2020	1.1	Finish all the parts of the Test Plan
3 April 2020	2.0	According to the comments and rubrics from TA to the vision 0 Test plan, the new test plan is updated. The functional and nonfunctional testing are changed more reasonable. Moreover, more unit tests are added into the test plan. The questions on the usability survey are edited to be clearer and new questions are added.

1 General Information

1.1 Purpose

The purpose of testing is to make sure the new game work properly on Windows and Mac OS X system.

1.2 Scope

The test plan states the testing rules and specific test cases to evaluate the functionality of the TankWar game. The objective of this document is to check whether the game meets the functional and non-functional requirements, and proof of concept in the SRS. The test plan can also be referred as the outline of the testing procedure to perform the tests.

1.3 Acronyms, Abbreviations, and Symbols

Table 2: **Table of Abbreviations**

Abbreviation	Definition
GUI	Graphical User Interface
PVP	Player1 Versus Player2
PVE	Player1 and Player2 Versus Environment
SRS	Software Requirements Specification

Table 3: **Table of Definitions**

Term	Definition
Developers	The team members.
Project	TankWar project. Recreate and upgrade the original game BattleCity.
Functional Test- ing	Testing derived from a description of how the program functions.
Dynamic Test- ing	Testing which includes test cases run during execution.
Static Testing	Testing that does not involve program execution.
Manual Testing	Testing conducted by people.
Automated Testing	Testing that run automatically by software.

1.4 Overview of Document

The document states the plan for testing the functional requirements, non-functional requirements, and proof of concept designed for the TankWar project in specific methods and metrics.

2 Plan

2.1 Software Description

The TankWar project will recreate and upgrade the open source project for BattleCity, the new game will keep all the original functionality which allows two players to fight against enemy tanks. Also, two new modes and three new types of tank will be added. The two new modes include PVP and map editing, which allow two players fight against each other and design their own maps, respectively. The new tanks added are DoubleLife tank, Fast tank, and FastBullet tank.

2.2 Test Team

The test team include all the group members, who are Xinyu Huang, Di Wu, and Jiahao Zhou.

2.3 Automated Testing Approach

The python unittest will be used to do the automated tests.

2.4 Testing Tools

Python unittest will be used for automated unit tests, and the python coverage tool coverage.py will be used to calculate the path, branch, statement coverage.

2.5 Testing Schedule

See Gantt Chart at the following url

3 System Test Description

3.1 Tests for Functional Requirements

3.1.1 Area of Testing1 for Tank Movements

Test for Going Up

1. MT-GU-1

Type: Manual, Dynamic, Functional

Initial State: After the game starts.

Input: "W" key held by player 1.

Output: Player 1's tank goes up, which is shown on the screen.

How test will be performed: After the game starts, the corresponding tank should go up if the player 1 holds "W" key from the keyboard.

2. MT-GU-2

Type: Manual, Dynamic, Functional

Initial State: After the game starts.

Input: "W" key pressed by player 1.

Output: Player 1's tank goes up by 1 unit of length, which is shown on the screen.

How test will be performed: After the game starts, the corresponding tank should go up for 1 unit of length if the player 1 presses "W" key once from the keyboard.

3. MT-GU-3

Type: Manual, Dynamic, Functional

Initial State: After the double-life tank is shot by a bullet and the ultimate skill is activated.

Input: "W" key held by player 1.

Output: Player 1's double-life tank still goes up, which is shown on the screen.

How test will be performed: After the double-life tank is shot and the ultimate skill is activated, the tank can ignore any damages for 1 second and player 1 can still let the tank move up by holding the "W" key from the keyboard.

4. MT-GU-4

Type: Manual, Dynamic, Functional

Initial State: After the double-life tank is shot by a bullet and the ultimate skill is activated.

Input: "W" key pressed by player 1.

Output: Player 1's double-life tank still goes up by 1 unit of length, which is shown on the screen.

How test will be performed: After the double-life tank is shot and the ultimate skill is activated, the tank can ignore any damages for 1 second and player 1 can still let the tank move up for 1 unit of length by pressing the "W" key from the keyboard.

5. MT-GU-5

Type: Manual, Dynamic, Functional

Initial State: After the game starts.

Input: Upward Arrow key held by player 2.

Output: Player 2's tank goes up, which is shown on the screen.

How test will be performed: After the game starts, the corresponding tank should go up if the player 2 holds upward arrow key from the keyboard.

6. MT-GU-6

Type: Manual, Dynamic, Functional

Initial State: After the game starts.

Input: Upward arrow key pressed by player 2.

Output: Player 2's tank goes up by 1 unit of length, which is shown on the screen.

How test will be performed: After the game starts, the corresponding tank should go up for 1 unit of length if the player 2 presses upward arrow key once from the keyboard.

7. MT-GU-7

Type: Manual, Dynamic, Functional

Initial State: After the double-life tank is shot by a bullet and the ultimate skill is activated.

Input: Upward arrow key held by player 2.

Output: Player 2's double-life tank still goes up, which is shown on the screen.

How test will be performed: After the double-life tank is shot and the ultimate skill is activated, the tank can ignore any damages for 1 second and player 2 can still let the tank move up by holding the upward arrow key from the keyboard.

8. MT-GU-8

Type: Manual, Dynamic, Functional

Initial State: After the double-life tank is shot by a bullet and the ultimate skill is activated.

Input: Upward arrow key pressed by player 2.

Output: Player 2's double-life tank still goes up by 1 unit of length, which is shown on the screen.

How test will be performed: After the double-life tank is shot and the ultimate skill is activated, the tank can ignore any damages for 1 second and player 2 can still let the tank move up for 1 unit of length by pressing the upward arrow key from the keyboard.

Test for Going Down

1. MT-GD-1

Type: Manual, Dynamic, Functional

Initial State: After the game starts.

Input: "S" key held by player 1.

Output: Player 1's tank goes down, which is shown on the screen.

How test will be performed: After the game starts, the corresponding tank should go down if the player 1 holds "S" key from the keyboard.

2. MT-GD-2

Type: Manual, Dynamic, Functional

Initial State: After the game starts.

Input: "S" key pressed by player 1.

Output: Player 1's tank goes down by 1 unit of length, which is shown on the screen.

How test will be performed: After the game starts, the corresponding tank should go down for 1 unit of length if the player 1 presses "S" key once from the keyboard.

3. MT-GD-3

Type: Manual, Dynamic, Functional

Initial State: After the double-life tank is shot by a bullet and the ultimate skill is activated.

Input: "S" key held by player 1.

Output: Player 1's double-life tank still goes down, which is shown on the screen.

How test will be performed: After the double-life tank is shot and the ultimate skill is activated, the tank can ignore any damages for 1 second and player 1 can still let the tank move down by holding the "S" key from the keyboard.

4. MT-GD-4

Type: Manual, Dynamic, Functional

Initial State: After the double-life tank is shot by a bullet and the ultimate skill is activated.

Input: "S" key pressed by player 1.

Output: Player 1's double-life tank still goes down by 1 unit of length, which is shown on the screen.

How test will be performed: After the double-life tank is shot and the ultimate skill is activated, the tank can ignore any damages for 1 second and player 1 can still let the tank move down for 1 unit of length by pressing the "S" key from the keyboard.

5. MT-GD-5

Type: Manual, Dynamic, Functional

Initial State: After the game starts.

Input: Downward Arrow key held by player 2.

Output: Player 2's tank goes down, which is shown on the screen.

How test will be performed: After the game starts, the corresponding tank should go down if the player 2 holds downward arrow key from the keyboard.

6. MT-GD-6

Type: Manual, Dynamic, Functional

Initial State: After the game starts.

Input: Downward arrow key pressed by player 2.

Output: Player 2's tank goes down by 1 unit of length, which is shown on the screen.

How test will be performed: After the game starts, the corresponding tank should go down for 1 unit of length if the player 2 presses downward arrow key once from the keyboard.

7. MT-GD-7

Type: Manual, Dynamic, Functional

Initial State: After the double-life tank is shot by a bullet and the ultimate skill is activated.

Input: Downward arrow key held by player 2.

Output: Player 2's double-life tank still goes down, which is shown on the screen.

How test will be performed: After the double-life tank is shot and the ultimate skill is activated, the tank can ignore any damages for 1 second and player 2 can still let the tank move down by holding the downward arrow key from the keyboard.

8. MT-GD-8

Type: Manual, Dynamic, Functional

Initial State: After the double-life tank is shot by a bullet and the ultimate skill is activated.

Input: Downward arrow key pressed by player 2.

Output: Player 2's double-life tank still goes down by 1 unit of length, which is shown on the screen.

How test will be performed: After the double-life tank is shot and the ultimate skill is activated, the tank can ignore any damages for 1 second

and player 2 can still let the tank move down for 1 unit of length by pressing the downward arrow key from the keyboard.

Test for Going Left

1. MT-GL-1

Type: Manual, Dynamic, Functional

Initial State: After the game starts.

Input: "A" key held by player 1.

Output: Player 1's tank goes left, which is shown on the screen.

How test will be performed: After the game starts, the corresponding tank should go left if the player 1 holds "A" key from the keyboard.

2. MT-GL-2

Type: Manual, Dynamic, Functional

Initial State: After the game starts.

Input: "A" key pressed by player 1.

Output: Player 1's tank goes left by 1 unit of length, which is shown on the screen.

How test will be performed: After the game starts, the corresponding tank should go left for 1 unit of length if the player 1 presses "A" key once from the keyboard.

3. MT-GL-3

Type: Manual, Dynamic, Functional

Initial State: After the double-life tank is shot by a bullet and the ultimate skill is activated.

Input: "A" key held by player 1.

Output: Player 1's double-life tank still goes left, which is shown on the screen.

How test will be performed: After the double-life tank is shot and the ultimate skill is activated, the tank can ignore any damages for 1 second and player 1 can still let the tank move left by holding the "A" key from the keyboard.

4. MT-GL-4

Type: Manual, Dynamic, Functional

Initial State: After the double-life tank is shot by a bullet and the ultimate skill is activated.

Input: "A" key pressed by player 1.

Output: Player 1's double-life tank still goes left by 1 unit of length, which is shown on the screen.

How test will be performed: After the double-life tank is shot and the ultimate skill is activated, the tank can ignore any damages for 1 second and player 1 can still let the tank move left for 1 unit of length by pressing the "A" key from the keyboard.

5. MT-GL-5

Type: Manual, Dynamic, Functional

Initial State: After the game starts.

Input: Leftwards Arrow key held by player 2.

Output: Player 2's tank goes left, which is shown on the screen.

How test will be performed: After the game starts, the corresponding tank should go left if the player 2 holds leftwards arrow key from the keyboard.

6. MT-GL-6

Type: Manual, Dynamic, Functional

Initial State: After the game starts.

Input: leftwards arrow key pressed by player 2.

Output: Player 2's tank goes left by 1 unit of length, which is shown on the screen.

How test will be performed: After the game starts, the corresponding tank should go left for 1 unit of length if the player 2 presses leftwards arrow key once from the keyboard.

7. MT-GL-7

Type: Manual, Dynamic, Functional

Initial State: After the double-life tank is shot by a bullet and the ultimate skill is activated.

Input: Leftwards arrow key held by player 2.

Output: Player 2's double-life tank still goes left, which is shown on the screen.

How test will be performed: After the double-life tank is shot and the ultimate skill is activated, the tank can ignore any damages for 1 second and player 2 can still let the tank move left by holding the leftwards arrow key from the keyboard.

8. MT-GL-8

Type: Manual, Dynamic, Functional

Initial State: After the double-life tank is shot by a bullet and the ultimate skill is activated.

Input: Leftwards arrow key pressed by player 2.

Output: Player 2's double-life tank still goes left by 1 unit of length, which is shown on the screen.

How test will be performed: After the double-life tank is shot and the ultimate skill is activated, the tank can ignore any damages for 1 second and player 2 can still let the tank move left for 1 unit of length by pressing the leftwards arrow key from the keyboard.

Test for Going Right

1. MT-GR-1

Type: Manual, Dynamic, Functional

Initial State: After the game starts.

Input: "D" key held by player 1.

Output: Player 1's tank goes right, which is shown on the screen.

How test will be performed: After the game starts, the corresponding tank should go right if the player 1 holds "D" key from the keyboard.

2. MT-GR-2

Type: Manual, Dynamic, Functional

Initial State: After the game starts.

Input: "D" key pressed by player 1.

Output: Player 1's tank goes right by 1 unit of length, which is shown on the screen.

How test will be performed: After the game starts, the corresponding tank should go right by 1 unit of length if the player 1 presses "D" key once from the keyboard.

3. MT-GR-3

Type: Manual, Dynamic, Functional

Initial State: After the double-life tank is shot by a bullet and the ultimate skill is activated.

Input: "D" key held by player 1.

Output: Player 1's double-life tank still goes right, which is shown on the screen.

How test will be performed: After the double-life tank is shot and the ultimate skill is activated, the tank can ignore any damages for 1 second and player 1 can still let the tank move right by holding the "D" key from the keyboard.

4. MT-GR-4

Type: Manual, Dynamic, Functional

Initial State: After the double-life tank is shot by a bullet and the ultimate skill is activated.

Input: "D" key pressed by player 1.

Output: Player 1's double-life tank still goes right by 1 unit of length, which is shown on the screen.

How test will be performed: After the double-life tank is shot and the ultimate skill is activated, the tank can ignore any damages for 1 second and player 1 can still let the tank move right for 1 unit of length by pressing the "D" key from the keyboard.

5. MT-GR-5

Type: Manual, Dynamic, Functional

Initial State: After the game starts.

Input: Rightwards Arrow key held by player 2.

Output: Player 2's tank goes right, which is shown on the screen.

How test will be performed: After the game starts, the corresponding tank should go right if the player 2 holds rightwards arrow key from the keyboard.

6. MT-GR-6

Type: Manual, Dynamic, Functional

Initial State: After the game starts.

Input: Rightwards arrow key pressed by player 2.

Output: Player 2's tank goes right by 1 unit of length, which is shown on the screen.

How test will be performed: After the game starts, the corresponding tank should go right for 1 unit of length if the player 2 presses rightwards arrow key once from the keyboard.

7. MT-GR-7

Type: Manual, Dynamic, Functional

Initial State: After the double-life tank is shot by a bullet and the ultimate skill is activated.

Input: Rightwards arrow key held by player 2.

Output: Player 2's double-life tank still goes right, which is shown on the screen.

How test will be performed: After the double-life tank is shot and the ultimate skill is activated, the tank can ignore any damages for 1 second and player 2 can still let the tank move right by holding the rightwards arrow key from the keyboard.

8. MT-GR-8

Type: Manual, Dynamic, Functional

Initial State: After the double-life tank is shot by a bullet and the ultimate skill is activated.

Input: Rightwards arrow key pressed by player 2.

Output: Player 2's double-life tank still goes right by 1 unit of length, which is shown on the screen.

How test will be performed: After the double-life tank is shot and the ultimate skill is activated, the tank can ignore any damages for 1 second and player 2 can still let the tank move right for 1 unit of length by pressing the rightwards arrow key from the keyboard.

Test for Walls and Boundaries

1. MT-WB-1

Type: Manual, Dynamic, Functional

Initial State: After the game starts.

Input: Player 1's tank is facing to a brick wall and there is no distance between the tank and the wall. Player 1 presses the corresponding key

from "WASD" keys in order to make the tank move towards the brick wall.

Output: Player 1's tank does not move.

How test will be performed: After the game starts, player 1's tank should not be moving if player 1 controls the tank and tries to move the tank through a brick wall using "WASD" keys from the keyboard.

2. MT-WB-2

Type: Manual, Dynamic, Functional

Initial State: After the game starts.

Input: Player 1's tank is facing to a iron wall and there is no distance between the tank and the wall. Player 1 presses the corresponding key from "WASD" keys in order to make the tank move towards the iron wall.

Output: Player 1's tank does not move.

How test will be performed: After the game starts, player 1's tank should not be moving if player 1 controls the tank and tries to move the tank through a iron wall using "WASD" keys from the keyboard.

3. MT-WB-3

Type: Manual, Dynamic, Functional

Initial State: After the game starts.

Input: Player 2's tank is facing to a brick wall and there is no distance between the tank and the wall. Player 2 presses the corresponding key from arrow keys in order to make the tank move towards the brick wall.

Output: Player 2's tank does not move.

How test will be performed: After the game starts, player 2's tank should not be moving if player 2 controls the tank and tries to move the tank through a brick wall using arrow keys from the keyboard.

4. MT-WB-4

Type: Manual, Dynamic, Functional

Initial State: After the game starts.

Input: Player 2's tank is facing to a iron wall and there is no distance between the tank and the wall. Player 2 presses the corresponding key from arrow keys in order to make the tank move towards the iron wall.

Output: Player 2's tank does not move.

How test will be performed: After the game starts, player 2's tank should not be moving if player 2 controls the tank and tries to move the tank through a iron wall using arrow keys from the keyboard.

5. MT-WB-5

Type: Manual, Dynamic, Functional

Initial State: After the game starts.

Input: Player 2's tank is facing to the boundary of the map and there is no distance between the tank and the boundary. Player 2 presses the corresponding key from arrow keys in order to make the tank move towards the boundary.

Output: Player 2's tank does not move.

How test will be performed: After the game starts, player 2's tank should not be moving if player 2 controls the tank and tries to move the tank out of the map using arrow keys from the keyboard.

6. MT-WB-6

Type: Manual, Dynamic, Functional

Initial State: After the game starts.

Input: Player 1's tank is facing to the boundary of the map and there is no distance between the tank and the boundary. Player 1 presses the corresponding key from "WASD" keys in order to make the tank move towards the boundary.

Output: Player 1's tank does not move.

How test will be performed: After the game starts, player 1's tank should not be moving if player 1 controls the tank and tries to move the tank out of the map using "WASD" keys from the keyboard.

3.1.2 Area of Testing² for Tank Attacking

Test for Shooting Bullets

1. TAT-SB-1

Type: Manual, Dynamic, Functional

Initial State: After the game starts.

Input: "J" key pressed by player 1.

Output: Player 1's tank shoots a bullet.

How test will be performed: After the game starts, the corresponding tank should shoot a bullet if player 1 presses "J" key from the keyboard.

2. TAT-SB-2

Type: Manual, Dynamic, Functional

Initial State: After the game starts.

Input: "J" key held by player 1.

Output: Player 1's tank shoots bullets continuously.

How test will be performed: After the game starts, the corresponding tank should shoot bullets continuously if player 1 holds "J" key from the keyboard.

3. TAT-SB-3

Type: Manual, Dynamic, Functional

Initial State: After the game starts.

Input: ",," key pressed by player 2.

Output: Player 2's tank shoots a bullet.

How test will be performed: After the game starts, the corresponding tank should shoot a bullet if player 2 presses ",," key from the keyboard.

4. TAT-SB-4

Type: Manual, Dynamic, Functional

Initial State: After the game starts.

Input: ",," key held by player 2.

Output: Player 2's tank shoots bullets continuously.

How test will be performed: After the game starts, the corresponding tank should shoot bullets continuously if player 2 holds ",," key from the keyboard.

5. TAT-SB-5

Type: Manual, Dynamic, Functional

Initial State: After the double-life tank get shot for the first time.

Input: ",," key held by player 2.

Output: Player 2's double-life tank still shoots bullets continuously.

How test will be performed: After the double-life tank get shot for the first time, the corresponding tank should still shoot bullets continuously if player 2 holds ",," key from the keyboard.

6. TAT-SB-6

Type: Manual, Dynamic, Functional

Initial State: After the double-life tank get shot for the first time.

Input: "J" key held by player 1.

Output: Player 1's double-life tank still shoots bullets continuously.

How test will be performed: After the double-life tank get shot for the first time, the corresponding tank should still shoot bullets continuously if player 1 holds "J" key from the keyboard.

7. TAT-SB-7

Type: Manual, Dynamic, Functional

Initial State: After the double-life tank get shot for the first time.

Input: ",," key pressed by player 2.

Output: Player 2's double-life tank still shoots a bullet.

How test will be performed: After the double-life tank get shot for the first time, the corresponding tank should still shoot a bullet if player 2 presses "," key from the keyboard.

8. TAT-SB-8

Type: Manual, Dynamic, Functional

Initial State: After the double-life tank get shot for the first time.

Input: "J" key pressed by player 1.

Output: Player 1's double-life tank still shoots a bullet.

How test will be performed: After the double-life tank get shot for the first time, the corresponding tank should still shoot a bullet if player 1 presses "J" key from the keyboard.

Test for Using Ultimate Skills

1. TAT-UUS-1

Type: Manual, Dynamic, Functional

Initial State: After the game starts.

Input: "K" key pressed by player 1.

Output: Player 1's tank use ultimate skills.

How test will be performed: After the game starts, the corresponding tank should use the corresponding ultimate skill if player 1 presses "K" key from the keyboard.

2. TAT-UUS-2

Type: Manual, Dynamic, Functional

Initial State: After the game starts.

Input: "." key pressed by player 2.

Output: Player 2's tank use ultimate skills.

How test will be performed: After the game starts, the corresponding tank should use the corresponding ultimate skill if player 2 presses "." key from the keyboard.

3. TAT-UUS-3

Type: Manual, Dynamic, Functional

Initial State: After the game starts.

Input: "K" key held by player 1.

Output: ~~Player 1's tank use ultimate skills just once.~~ Player 1's tank use ultimate skills continuously.

How test will be performed: ~~After the game starts, the corresponding tank should use the corresponding ultimate skill just once even if player 1 holds "K" key from the keyboard.~~ After the game starts, the corresponding tank should use the corresponding ultimate skill continuously if player 1 holds "K" key from the keyboard.

4. TAT-UUS-4

Type: Manual, Dynamic, Functional

Initial State: After the game starts.

Input: "." key held by player 2.

Output: ~~Player 2's tank use ultimate skills just once.~~ Player 2's tank use ultimate skills continuously.

How test will be performed: ~~After the game starts, the corresponding tank should use the corresponding ultimate skill just once even if player 2 holds "K" key from the keyboard.~~ After the game starts, the corresponding tank should use the corresponding ultimate skill continuously if player 2 holds "K" key from the keyboard.

3.1.3 Area of Testing³ for Tank Properties

Test for Double-Life Tank

1. TPT-DLT-1

Type: Manual, Dynamic, Functional

Initial State: After the game starts.

Input: Player chooses double-life tank before Entering the game and the player presses "K" key or "." key to activate the ultimate skill.

Output: The double-life tank is still alive when the ultimate skill is activated and the tank gets shot simultaneously.

How test will be performed: After the game starts, the double-life tank should be still alive if the enemy tank shoots the double-life tank and the player presses "K" key or "." key on the keyboard to activate the ultimate skill which protects the double-life tank from any damages. After 1 second, the damages should be fatal to the double-life tank.

Test for High-Speed Tank

1. TPT-HST-1

Type: Manual, Dynamic, Functional

Initial State: After the game starts.

Input: Player chooses high-speed tank before Entering the game and the player presses "K" key or "." key to activate the ultimate skill.

Output: During the tank movement, the tank rushes for ~~2-seconds~~ **1 second** after the ultimate skill is activated. The direction can be changed by the player during the rushing process by pressing another key on the keyboard.

How test will be performed: After the game starts, the high-speed tank should be able to rush for ~~2-seconds~~ **1 second** when the player presses "K" key or "." key on the keyboard to activate the ultimate skill. During the rushing process, player 1 or player 2 can change the rushing direction by pressing "WASD" or arrow keys respectively on the keyboard. After ~~2-seconds~~ **1 second**, the high-speed tank should not have the ability of rushing.

Test for Double-Bullet Tank

1. TPT-DBT-1

Type: Manual, Dynamic, Functional

Initial State: After the game starts.

Input: Player chooses double-bullet tank before Entering the game and the player presses "K" key or "." key on the keyboard to activate the ultimate skill.

Output: During attacking, the double-bullet tank shoots two bullets simultaneously when the ultimate skill activates.

How test will be performed: After the game starts, the double-bullet tank should be able to shoot two bullets simultaneously when the player presses "K" key or "." key on the keyboard to activate the ultimate skill. The ultimate skill should allow the tank to shoot double bullets once. After that, there should only be one bullet coming out from the double-bullet tank per shooting.

3.1.4 Area of Testing⁴ for Gaming Process

Test for PVP Mode Selection

1. GPT-PPMS-1

Type: Manual, Dynamic, Functional

Initial State: Player 1 opens the software.

Input: Player 1 selects PVP mode using "~~J~~"~~key~~ ~~Enter~~ ~~key~~ as the button of confirming from the keyboard.

Output: The PVP map loading section shows on the screen.

How test will be performed: After opening the software, there should be three modes for player 1 to choose. If the player 1 chooses PVP mode, there should be the interface for PVP ~~map-selection~~ ~~tank selection~~ shown on the screen.

Test for PVE Mode Selection

1. GPT-PEMS-1

Type: Manual, Dynamic, Functional

Initial State: Player 1 opens the software.

Input: Player 1 selects PVE mode using ~~”J” key~~ Enter key as the button of confirming from the keyboard.

Output: The PVE map loading section shows on the screen.

How test will be performed: After opening the software, there should be three modes for player 1 to choose. If the player 1 chooses PVE mode, there should be the interface for PVE ~~map-selection~~ tank selection shown on the screen.

Test for Map Editing Mode Selection

1. GPT-MEMS-1

Type: Manual, Dynamic, Functional

Initial State: Player 1 opens the software.

Input: Player 1 selects map editing mode using ~~”J” key~~ Enter key as the button of confirming from the keyboard.

Output: The map editing interface shows on the screen.

How test will be performed: After opening the software, there should be three modes for player 1 to choose. If the player 1 chooses map editing mode, there should be the interface for ~~map-editing~~ map selection shown on the screen, ~~which allows the player to do further operations like add and delete walls.~~

Test for PVP Map Loading

1. GPT-PPML-1

Type: Manual, Dynamic, Functional

Initial State: Player 1 opens the software and selects PVP mode using ~~”J” key~~ Enter key as the button of confirming from the keyboard.

Input: Player 1 selects one of the PVP maps in the PVP map selecting section.

Output: The selected PVP map shows on the screen.

How test will be performed: After opening the software and selecting PVP mode using "J" key as the button of confirming from the keyboard, the system should load the corresponding PVP map and show on the screen if player 1 selects one of the PVP maps in the PVP map selecting section by pressing "J" key as the button of confirming from the keyboard.

Test for PVE Map Loading

1. GPT-PEML-1

Type: Manual, Dynamic, Functional

Initial State: Player 1 opens the software and selects PVE mode using ~~"J" key~~ Enter key as the button of confirming from the keyboard.

Input: Player 1 selects one of the PVE maps in the PVE map selecting section.

Output: The selected PVE map shows on the screen.

How test will be performed: After opening the software and selecting PVE mode using "J" key as the button of confirming from the keyboard, the system should load the corresponding PVE map and show on the screen if player 1 selects one of the PVE maps in the PVE map selecting section by pressing "J" key as the button of confirming from the keyboard.

Test for PVP Tank Selection

1. GPT-PPTS-1

Type: Manual, Dynamic, Functional

Initial State: Player 1 opens the software, selects PVP mode, ~~and selects one of the PVP maps in the PVP map selecting section~~ by using ~~"J" key~~ Enter key as the button of confirming from the keyboard.

~~"J" key~~ Enter key Player 1 chooses one from three different kinds of tanks by using "J" key as the button of confirming from the keyboard.

Output: The corresponding tank shows on the selected PVP map.

How test will be performed: After opening the software, selecting PVP mode, and selecting one of the PVP maps, the system should show a tank selection interface which allows player 1 to choose a tank among three different kinds of tanks by using ~~"J" key~~ RETURN key as the button of confirming from the keyboard. After choosing, the corresponding tank should show on the selected PVP map.

2. GPT-PPTS-2

Type: Manual, Dynamic, Functional

Initial State: Player 1 opens the software, selects PVP mode, ~~and selects one of the PVP maps in the PVP map selecting section~~ by using ~~"J" key~~ Enter key as the button of confirming from the keyboard.

Input: Player 2 chooses one from three different kinds of tanks using ~~"," key~~ Enter key as the button of confirming from the keyboard.

Output: The corresponding tank shows on the selected PVP map.

How test will be performed: After opening the software, selecting PVP mode, and selecting one of the PVP maps, the system should show a tank selection interface which allows player 2 to choose a tank among three different kinds of tanks using ~~"," key~~ Enter key as the button of confirming from the keyboard. After choosing, the corresponding tank should show on the selected PVP map.

Test for PVE Tank Selection

1. GPT-PETS-1

Type: Manual, Dynamic, Functional

Initial State: Player 1 opens the software, selects PVE mode, ~~and selects one of the PVP maps in the PVP map selecting section~~ by using ~~"J" key~~ Enter key as the button of confirming from the keyboard.

Input: Player 1 chooses one from three different kinds of tanks by using ~~"J" key~~ Enter key as the button of confirming from the keyboard.

Output: The corresponding tank shows on the selected PVE map.

How test will be performed: After opening the software, selecting PVE mode, and selecting one of the PVE maps, the system should show a tank selection interface which allows player 1 to choose a tank among three different kinds of tanks. After choosing, the corresponding tank should show on the selected PVE map.

2. GPT-PETS-2

Type: Manual, Dynamic, Functional

Initial State: Player 1 opens the software, selects PVE mode, ~~and selects one of the PVP maps in the PVP map selecting section~~ by using ~~"J" key~~ Enter key as the button of confirming from the keyboard.

Input: Player 2 chooses one from three different kinds of tanks using ~~"J" key~~ Enter key as the button of confirming from the keyboard.

Output: The corresponding tank shows on the selected PVE map.

How test will be performed: After opening the software, selecting PVE mode, and selecting one of the PVE maps, the system should show a tank selection interface which allows player 2 to choose a tank among three different kinds of tanks using ~~"J" key~~ Enter key as the button of confirming from the keyboard. After choosing, the corresponding tank should show on the selected PVE map.

3.1.5 Area of Testing⁵ for Map Editing

Test for Adding Brick Wall for PVP Map

1. ME-PVPAB-1

Type: Functional, Dynamic, Manual.

Initial State: The map editing for PVP mode is running, and there is empty space in front of the player's tank.

Input: ~~"K" key pressed.~~ "J" key pressed

Output: A brick wall is added right in front of the player's tank.

How test will be performed: The map editing for PVP mode will be run and player's tank will stay at a location with at least one unit of empty space in front of it, then the "K" key will be pressed and check if a brick wall is correctly added to that empty unit.

2. ME-PVPAB-2

Type: Functional, Dynamic, Manual.

Initial State: The map editing for PVP mode is running, and the player's tank is facing to a brick wall.

Input: ~~"K" key pressed.~~ "J" key pressed

Output: Nothing is changed.

How test will be performed: The map editing for PVP mode will be run and the player's tank will be moved to face to and with no distance with a existing brick wall, then the "K" key will be pressed and check if nothing is changed.

3. ME-PVPAB-3

Type: Functional, Dynamic, Manual.

Initial State: The map editing for PVP mode is running, and the player's tank is facing to a iron wall.

Input: ~~"K" key pressed.~~ "J" key pressed

Output: Nothing is changed.

How test will be performed: The map editing for PVP mode will be run and the player's tank will be moved to face to and with no distance with a existing iron wall, then the "K" key will be pressed and check if nothing is changed.

4. ME-PVPAB-4

Type: Functional, Dynamic, Manual.

Initial State: The map editing for PVP mode is running, and the player's tank is facing the boundary of the map.

Input: ~~"K" key pressed.~~ "J" key pressed

Output: Nothing is changed.

How test will be performed: The map editing for PVP mode will be run and the player's tank will be moved to face to and with no distance with the boundary of the map, then the "K" key will be pressed and check if nothing is changed.

Test for Adding Iron Wall for PVP Map

1. ME-PVPAI-1

Type: Functional, Dynamic, Manual.

Initial State: The map editing for PVP mode is running, and there is empty space in front of the player's tank.

Input: ~~"L" key pressed.~~ "K" key pressed

Output: A iron wall is added right in front of the player's tank.

How test will be performed: The map editing for PVP mode will be run and player's tank will stay at a location with at least one unit of empty space in front of it, then the "L" key will be pressed and check if a iron wall is correctly added to that empty unit.

2. ME-PVPAI-2

Type: Functional, Dynamic, Manual.

Initial State: The map editing for PVP mode is running, and the player's tank is facing to a brick wall.

Input: ~~"L" key pressed.~~ "K" key pressed

Output: Nothing is changed.

How test will be performed: The map editing for PVP mode will be run and the player's tank will be moved to face to and with no distance with a existing brick wall, then the "L" key will be pressed and check if nothing is changed.

3. ME-PVPAI-3

Type: Functional, Dynamic, Manual.

Initial State: The map editing for PVP mode is running, and the player's tank is facing to a iron wall.

Input: ~~"L" key pressed.~~ "K" key pressed

Output: Nothing is changed.

How test will be performed: The map editing for PVP mode will be run and the player's tank will be moved to face to and with no distance with a existing iron wall, then the "L" key will be pressed and check if nothing is changed.

4. ME-PVPAI-4

Type: Functional, Dynamic, Manual.

Initial State: The map editing for PVP mode is running, and the player's tank is facing to the boundary of the map.

Input: ~~"L" key pressed.~~ "K" key pressed

Output: Nothing is changed.

How test will be performed: The map editing for PVP mode will be run and the player's tank will be moved to face to and with no distance with the boundary of the map, then the "L" key will be pressed and check if nothing is changed.

Test for Deleting Wall for PVP Map

1. ME-PVPDW-1

Type: Functional, Dynamic, Manual.

Initial State: The map editing for PVP mode is running, and the player's tank is aim at a brick wall.

Input: ~~"J" key pressed.~~ "L" key pressed

Output: The brick wall is deleted from the map.

How test will be performed: The map editing for PVP mode will be run and the player's tank will be moved to aim at a brick wall, then the "J" key will be pressed and check if the brick wall is deleted.

2. ME-PVPDW-2

Type: Functional, Dynamic, Manual.

Initial State: The map editing for PVP mode is running, and the player's tank is aim at a iron wall.

Input: ~~"J" key pressed.~~ "L" key pressed

Output: The iron wall is deleted from the map.

How test will be performed: The map editing for PVP mode will be run and the player's tank will be moved to aim at a iron wall, then the "J" key will be pressed and check if the iron wall is deleted.

3. ME-PVPDW-3

Type: Functional, Dynamic, Manual.

Initial State: The map editing for PVP mode is running, and the player's tank is aim at nothing.

Input: ~~"J" key pressed.~~ "L" key pressed

Output: Nothing is changed in the map.

How test will be performed: The map editing for PVP mode will be run and the player's tank will be moved to aim at nothing, then the "J" key will be pressed and check if nothing is changed.

Test for Adding Brick Wall for PVE Map

1. ME-PVEAB-1

Type: Functional, Dynamic, Manual.

Initial State: The map editing for PVE mode is running, and there is empty space in front of the player's tank.

Input: ~~"K" key pressed.~~ "J" key pressed

Output: A brick wall is added right in front of the player's tank.

How test will be performed: The map editing for PVE mode will be run and player's tank will stay at a location with at least one unit of empty space in front of it, then the "K" key will be pressed and check if a brick wall is correctly added to that empty unit.

2. ME-PVEAB-2

Type: Functional, Dynamic, Manual.

Initial State: The map editing for PVE mode is running, and the player's tank is facing to a brick wall.

Input: ~~"K" key pressed.~~ "J" key pressed

Output: Nothing is changed.

How test will be performed: The map editing for PVE mode will be run and the player's tank will be moved to face to and with no distance with a existing brick wall, then the "K" key will be pressed and check if nothing is changed.

3. ME-PVEAB-3

Type: Functional, Dynamic, Manual.

Initial State: The map editing for PVE mode is running, and the player's tank is facing to a iron wall.

Input: ~~"K" key pressed.~~ "J" key pressed

Output: Nothing is changed.

How test will be performed: The map editing for PVE mode will be run and the player's tank will be moved to face to and with no distance with a existing iron wall, then the "K" key will be pressed and check if nothing is changed.

4. ME-PVEAB-4

Type: Functional, Dynamic, Manual.

Initial State: The map editing for PVE mode is running, and the player's tank is facing the boundary of the map.

Input: ~~"K" key pressed.~~ "J" key pressed

Output: Nothing is changed.

How test will be performed: The map editing for PVE mode will be run and the player's tank will be moved to face to and with no distance with the boundary of the map, then the "K" key will be pressed and check if nothing is changed.

Test for Adding Iron Wall for PVE Map

1. ME-PVEAI-1

Type: Functional, Dynamic, Manual.

Initial State: The map editing for PVE mode is running, and there is empty space in front of the player's tank.

Input: ~~"L" key pressed.~~ "K" key pressed

Output: A iron wall is added right in front of the player's tank.

How test will be performed: The map editing for PVE mode will be run and player's tank will stay at a location with at least one unit of empty space in front of it, then the "L" key will be pressed and check if a iron wall is correctly added to that empty unit.

2. ME-PVEAI-2

Type: Functional, Dynamic, Manual.

Initial State: The map editing for PVE mode is running, and the player's tank is facing to a brick wall.

Input: ~~"L" key pressed.~~ "K" key pressed

Output: Nothing is changed.

How test will be performed: The map editing for PVE mode will be run and the player's tank will be moved to face to and with no distance with a existing brick wall, then the "L" key will be pressed and check if nothing is changed.

3. ME-PVEAI-3

Type: Functional, Dynamic, Manual.

Initial State: The map editing for PVE mode is running, and the player's tank is facing to a iron wall.

Input: ~~"L" key pressed.~~ "K" key pressed

Output: Nothing is changed.

How test will be performed: The map editing for PVE mode will be run and the player's tank will be moved to face to and with no distance with a existing iron wall, then the "L" key will be pressed and check if nothing is changed.

4. ME-PVEAI-4

Type: Functional, Dynamic, Manual.

Initial State: The map editing for PVE mode is running, and the player's tank is facing to the boundary of the map.

Input: ~~"L" key pressed.~~ "K" key pressed

Output: Nothing is changed.

How test will be performed: The map editing for PVE mode will be run and the player's tank will be moved to face to and with no distance with the boundary of the map, then the "L" key will be pressed and check if nothing is changed.

Test for Deleting Wall for PVE Map

1. ME-PVEDW-1

Type: Functional, Dynamic, Manual.

Initial State: The map editing for PVE mode is running, and the player's tank is aim at a brick wall.

Input: ~~"J" key pressed.~~ "L" key pressed

Output: The brick wall is deleted from the map.

How test will be performed: The map editing for PVE mode will be run and the player's tank will be moved to aim at a brick wall, then the "J" key will be pressed and check if the brick wall is deleted.

2. ME-PVEDW-2

Type: Functional, Dynamic, Manual.

Initial State: The map editing for PVE mode is running, and the player's tank is aim at a iron wall.

Input: ~~"J" key pressed.~~ "L" key pressed

Output: The iron wall is deleted from the map.

How test will be performed: The map editing for PVE mode will be run and the player's tank will be moved to aim at a iron wall, then the "J" key will be pressed and check if the iron wall is deleted.

3. ME-PVEDW-3

Type: Functional, Dynamic, Manual.

Initial State: The map editing for PVE mode is running, and the player's tank is aim at nothing.

Input: ~~"J" key pressed.~~ "L" key pressed

Output: Nothing is changed in the map.

How test will be performed: The map editing for PVE mode will be run and the player's tank will be moved to aim at nothing, then the "J" key will be pressed and check if nothing is changed.

Test for Saving Maps for PVP

1. ME-PVPSM-1

Type: Functional, Dynamic, Manual.

Initial State: The player is in map editing mode and a map is created.

Input: Press the "Enter" key, then ~~Enter a valid file name and select the existing files and~~ press "Enter" key again.

Output: A file stands for the map created is shown in the folder for PVP maps.

How test will be performed: Create a map in map editing mode, then press "Enter" key and Enter the file name. Press the "Enter" again and check if the map file is correctly saved in the folder for PVP maps.

Test for Saving Maps for PVE

1. ME-PVESM-1

Type: Functional, Dynamic, Manual.

Initial State: The player is in map editing mode and a map is created.

Input: Press the "Enter" key, then ~~Enter a valid file name and select the existing files and~~ press "Enter" key again.

Output: A file stands for the map created is shown in the folder for PVE maps.

How test will be performed: Create a map in map editing mode, then press "Enter" key and Enter the file name. Press the "Enter" again and check if the map file is correctly saved in the folder for PVE maps.

3.1.6 Area of Testing6 for Game Rules

Test for General Rules about the Wall

1. GR-GRW-1

Type: Functional, Dynamic, Manual.

Initial State: After the PVP game starts.

Input: Player 1 presses "J" key from the keyboard to shoot a bullet towards the brick wall and the bullet hits the brick wall.

Output: The brick wall is destroyed by the bullet. The bullet and the brick wall disappear simultaneously.

How test will be performed: After PVP game starts, the brick wall should be destroyed if player 1 presses "J" key from the keyboard to shoot a bullet towards the brick wall. After the bullet hits the brick

wall, the bullet and the brick wall should disappear at the same time and the tank should be able to move through the location of the destroyed brick wall.

2. GR-GRW-2

Type: Functional, Dynamic, Manual.

Initial State: After the PVE game starts.

Input: Player 1 presses "J" key from the keyboard to shoot a bullet towards the brick wall and the bullet hits the brick wall.

Output: The brick wall is destroyed by the bullet. The bullet and the brick wall disappear simultaneously.

How test will be performed: After PVE game starts, the brick wall should be destroyed if player 1 presses "J" key from the keyboard to shoot a bullet towards the brick wall. After the bullet hits the brick wall, the bullet and the brick wall should disappear at the same time and the tank should be able to move through the location of the destroyed brick wall.

3. GR-GRW-3

Type: Functional, Dynamic, Manual.

Initial State: After the PVE game starts.

Input: Player 2 presses "," key from the keyboard to shoot a bullet towards the brick wall and the bullet hits the brick wall.

Output: The brick wall is destroyed by the bullet. The bullet and the brick wall disappear simultaneously.

How test will be performed: After PVE game starts, the brick wall should be destroyed if player 2 presses "," key from the keyboard to shoot a bullet towards the brick wall. After the bullet hits the brick wall, the bullet and the brick wall should disappear at the same time and the tank should be able to move through the location of the destroyed brick wall.

4. GR-GRW-4

Type: Functional, Dynamic, Manual.

Initial State: After the PVP game starts.

Input: Player 2 presses "," key from the keyboard to shoot a bullet towards the brick wall and the bullet hits the brick wall.

Output: The brick wall is destroyed by the bullet. The bullet and the brick wall disappear simultaneously.

How test will be performed: After PVP game starts, the brick wall should be destroyed if player 2 presses "," key from the keyboard to shoot a bullet towards the brick wall. After the bullet hits the brick wall, the bullet and the brick wall should disappear at the same time and the tank should be able to move through the location of the destroyed brick wall.

5. GR-GRW-5

Type: Functional, Dynamic, Manual.

Initial State: After the PVP game starts.

Input: Player 1 presses "J" key from the keyboard to shoot a bullet towards the iron wall and the bullet hits the iron wall.

Output: The iron wall is not destroyed by the bullet. The bullet disappear.

How test will be performed: After PVP game starts, the iron wall should not be destroyed if player 1 presses "J" key from the keyboard to shoot a bullet towards the iron wall. After the bullet hits the iron wall, the bullet should disappear and the iron wall should be still at the original location.

6. GR-GRW-6

Type: Functional, Dynamic, Manual.

Initial State: After the PVE game starts.

Input: Player 1 presses "J" key from the keyboard to shoot a bullet towards the iron wall and the bullet hits the iron wall.

Output: The iron wall is not destroyed by the bullet. The bullet disappear.

How test will be performed: After PVE game starts, the iron wall should not be destroyed if player 1 presses "J" key from the keyboard to shoot a bullet towards the iron wall. After the bullet hits the iron wall, the bullet should disappear and the iron wall should be still at the original location.

7. GR-GRW-7

Type: Functional, Dynamic, Manual.

Initial State: After the PVE game starts.

Input: Player 2 presses "," key from the keyboard to shoot a bullet towards the iron wall and the bullet hits the iron wall.

Output: The iron wall is not destroyed by the bullet. The bullet disappear.

How test will be performed: After PVE game starts, the iron wall should not be destroyed if player 2 presses "," key from the keyboard to shoot a bullet towards the iron wall. After the bullet hits the iron wall, the bullet should disappear and the iron wall should be still at the original location.

8. GR-GRW-8

Type: Functional, Dynamic, Manual.

Initial State: After the PVP game starts.

Input: Player 2 presses "," key from the keyboard to shoot a bullet towards the iron wall and the bullet hits the iron wall.

Output: The iron wall is not destroyed by the bullet. The bullet disappear.

How test will be performed: After PVP game starts, the iron wall should not be destroyed if player 2 presses "," key from the keyboard to shoot a bullet towards the iron wall. After the bullet hits the iron wall, the bullet should disappear and the iron wall should be still at the original location.

Test for General Rules about Buff

1. ~~GR-GRB-1~~

~~Type: Functional, Dynamic, Manual.~~

~~Initial State: After the PVP game starts.~~

~~Input: Two players Enter the PVP game successfully.~~

~~Output: Different buffs randomly appear in the different locations of the map during a random time period.~~

~~How test will be performed: After PVP game starts, the system should offer buffs in random locations of the map and during the random time period. There should be three kinds of buffs that the system can offer: moving speed-enhanced buff, wall-enhanced buff, and bullet-speed-enhanced buff.~~

2. GR-GRB-2

Type: Functional, Dynamic, Manual.

Initial State: After the PVE game starts.

Input: Two players Enter the PVE game successfully.

Output: Different buffs randomly appear in the different locations of the map during a random time period.

How test will be performed: After PVE game starts, the system should offer buffs in random locations of the map and during the random time period. There should be three kinds of buffs that the system can offer: moving speed-enhanced buff, wall-enhanced buff, and bullet-speed-enhanced buff.

3. ~~GR-GRB-3~~

~~Type: Functional, Dynamic, Manual.~~

~~Initial State: After the PVP game starts.~~

~~Input: The player gets the moving-speed-enhanced buff during the battle.~~

~~Output: The player's tank gets a higher speed after getting the moving-speed-enhanced buff. The moving-speed-enhanced buff disappear after the player's tank gets it.~~

~~How test will be performed: After PVP game starts, the corresponding tank will get a higher speed after the player controls the tank to get the moving-speed-enhanced buff. The moving-speed-enhanced buff should disappears after the player's tank gets it. The buff should be available only for ten seconds. After ten seconds, the speed of the tank should go back to normal.~~

~~4. GR-GRB-4~~

~~Type: Functional, Dynamic, Manual.~~

~~Initial State: After the PVP game starts.~~

~~Input: The player gets the wall-enhanced buff during the battle.~~

~~Output: The player's home base is surrounded by iron walls instead of brick walls. The wall-enhanced buff disappears after the player's tank gets it.~~

~~How test will be performed: After PVP game starts, the corresponding player's home base should get surrounded by iron walls if the player controls the tank to get a wall-enhanced buff. Iron walls can withstand any damages. The wall-enhanced buff should disappears after the player's tank gets it. The buff should be available only for ten seconds. After ten seconds, the home base will be surrounded by normal brick walls again.~~

~~5. GR-GRB-5~~

~~Type: Functional, Dynamic, Manual.~~

~~Initial State: After the PVP game starts.~~

~~Input: The player gets the bullet-speed-enhanced buff during the battle.~~

~~Output: Bullets shot by the player's tank gets a higher moving speed. The bullet-speed-enhanced buff disappears after the player's tank gets it.~~

~~How test will be performed: After PVP game starts, the bullets shot by corresponding player's tank should get a higher moving speed if the player controls the tank to get a wall-enhanced buff. The bullet-speed-enhanced buff should disappears after the player's tank gets it. The buff should be available only for ten seconds. After ten seconds, the speed of the bullet shot by the player's tank should go back to normal.~~

6. GR-GRB-6

Type: Functional, Dynamic, Manual.

Initial State: After the PVE game starts.

Input: The player gets the moving-speed-enhanced buff during the battle.

Output: The player's tank gets a higher speed after getting the moving-speed-enhanced buff. The moving-speed-enhanced buff disappear after the player's tank gets it.

How test will be performed: After PVE game starts, the corresponding tank will get a higher speed after the player controls the tank to get the moving-speed-enhanced buff. The moving-speed-enhanced buff should disappears after the player's tank gets it. The buff should be available only for ten seconds. After ten seconds, the speed of the tank should go back to normal.

7. GR-GRB-7

Type: Functional, Dynamic, Manual.

Initial State: After the PVE game starts.

Input: The player gets the wall-enhanced buff during the battle.

Output: The player's home base is surrounded by iron walls instead of brick walls. The wall-enhanced buff disappears after the player's tank gets it.

How test will be performed: After PVE game starts, the corresponding player's home base should get surrounded by iron walls if the player controls the tank to get a wall-enhanced buff. Iron walls can withstand any damages. The wall-enhanced buff should disappears after

the player's tank gets it. The buff should be available only for ten seconds. After ten seconds, the home base will be surrounded by normal brick walls again.

8. GR-GRB-8

Type: Functional, Dynamic, Manual.

Initial State: After the PVE game starts.

Input: The player gets the bullet-speed-enhanced buff during the battle.

Output: Bullets shot by the player's tank gets a higher moving speed. The bullet-speed-enhanced buff disappears after the player's tank gets it.

How test will be performed: After PVE game starts, the bullets shot by corresponding player's tank should get a higher moving speed if the player controls the tank to get a wall-enhanced buff. The bullet-speed-enhanced buff should disappears after the player's tank gets it. The buff should be available only for ten seconds. After ten seconds, the speed of the bullet shot by the player's tank should go back to normal.

Test for General Rules about Lifes

1. GR-GRL-1

Type: Functional, Dynamic, Manual.

Initial State: After the PVP game starts.

Input: Player 1 controls the tank using "WASD" keys from the keyboard and drives the tank towards the bullet shot by the player 2's tank for three times. The bullet hits player 1's tank three times.

Output: Player 1's tank relives for the first and second time after hit by the bullet shot by player 2's tank. Player 1's tank does not relive after the third-time hit by the bullet.

How test will be performed: After PVP game starts, player 1's tank could only relive twice if player 2's tank shoots bullets to the player 1's tank. When the bullet hit player 1's tank for the third time, player 1's tank should be directly destroyed and have no chance for relive.

2. GR-GRL-2

Type: Functional, Dynamic, Manual.

Initial State: After the PVP game starts.

Input: Player 2 controls the tank using "WASD" keys from the keyboard and drives the tank towards the bullet shot by the player 1's tank for three times. The bullet hits player 2's tank three times.

Output: Player 2's tank relives for the first and second time after hit by the bullet shot by player 1's tank. Player 2's tank does not relive after the third-time hit by the bullet.

How test will be performed: After PVP game starts, player 2's tank could only relive twice if player 1's tank shoots bullets to the player 2's tank. When the bullet hit player 2's tank for the third time, player 2's tank should be directly destroyed and have no chance for relive.

3. GR-GRL-3

Type: Functional, Dynamic, Manual.

Initial State: After the PVE game starts.

Input: Player 1 controls the tank using "WASD" keys from the keyboard and drives the tank towards the bullet shot by the enemy tanks for three times. The bullet hits player 1's tank three times.

Output: Player 1's tank relives for the first and second time after hit by the bullet shot by enemy tanks. Player 1's tank does not relive after the third-time hit by the bullet.

How test will be performed: After PVE game starts, player 1's tank could only relive twice if enemy tanks shoot bullets to the player 1's tank. When the bullet hit player 1's tank for the third time, player 1's tank should be directly destroyed and have no chance for relive.

4. GR-GRL-4

Type: Functional, Dynamic, Manual.

Initial State: After the PVE game starts.

Input: Player 2 controls the tank using "WASD" keys from the keyboard and drives the tank towards the bullet shot by the enemy tanks for three times. The bullet hits player 2's tank three times.

Output: Player 2's tank relives for the first and second time after hit by the bullet shot by enemy tanks. Player 2's tank does not relive after the third-time hit by the bullet.

How test will be performed: After PVE game starts, player 2's tank could only relive twice if enemy tanks shoot bullets to the player 2's tank. When the bullet hit player 2's tank for the third time, player 2's tank should be directly destroyed and have no chance for relive.

Test for General Rules about Timing

1. GR-GRT-1

Type: Functional, Dynamic, Manual.

Initial State: After the PVP game starts.

Input: Player 1 and player 2 Enter the PVP game successfully.

Output: Player 1 and player 2 participate a PVP game for at most 3 minutes, if the two home bases are not destroyed and no player's tank is destroyed for 3 times.

How test will be performed: After PVP game starts, both players should participate the game for at most 3 minutes if the two home bases are not destroyed and no player's tank is destroyed for 3 times. In other words, the game should keep running for 3 minutes if there is no event that can affect the result of the game happens during the gaming process.

2. GR-GRT-2

Type: Functional, Dynamic, Manual.

Initial State: After the PVE game starts.

Input: Player 1 and player 2 Enter the PVE game successfully.

Output: Player 1 and player 2 participate a PVE game for at most 3 minutes, if the home base is not destroyed and no player's tank is destroyed for 3 times.

How test will be performed: After PVE game starts, both players should participate the game for at most 3 minutes if the home base is not destroyed and no player's tank is destroyed for 3 times. In other words, the game should keep running for 3 minutes if there is no event that can affect the result of the game happens during the gaming process.

Test for PVP Rules

1. GR-PPR-1

Type: Functional, Dynamic, Manual.

Initial State: After the PVP game starts.

Input: Player 1 and player 2 Enter the PVP game successfully.

Output: There are two home bases shown on the PVP map located on the left and right hand side of the map.

How test will be performed: After PVP game starts, the system should provide two home bases for the two players if they successfully Enter the PVP mode. The home base is an element that can possibly affect the result of the game.

2. GR-PPR-2

Type: Functional, Dynamic, Manual.

Initial State: After the PVP game starts.

Input: Player 1 and player 2 play for 3 minutes. No tank is destroyed three times and no tank destroys the other player's home base during this time period.

Output: A "Tie" is shown on the screen as the result of the game.

How test will be performed: After PVP game starts, the game should be considered as a tie if no tank is destroyed three times and no tank

destroys the other player's home base in 3 minutes. The system should show the result on the screen.

3. GR-PPR-3

Type: Functional, Dynamic, Manual.

Initial State: After the PVP game starts.

Input: Player 1's tank destroys player 2's tank three times during the 3-minute period.

Output: A "**Player 1 Win**" is shown on the screen as the result of the game.

How test will be performed: After PVP game starts, the game should be considered as a win if player 1's tank destroys player 2's tank three times during the 3-minute period. In this case, player 1 can be considered as winning without destroying player 2's home base.

4. GR-PPR-4

Type: Functional, Dynamic, Manual.

Initial State: After the PVP game starts.

Input: Player 2's tank destroys player 1's tank three times during the 3-minute period.

Output: A "**Lose Player 2 Win**" is shown on the screen as the result of the game.

How test will be performed: After PVP game starts, the game should be considered as a lose if player 2's tank destroys player 1's tank three times during the 3-minute period. In this case, player 2 can be considered as winning without destroying player 1's home base.

5. GR-PPR-5

Type: Functional, Dynamic, Manual.

Initial State: After the PVP game starts.

Input: Player 2's tank destroys player 1's home base during the 3-minute period.

Output: A "Lose Player 2 Win" is shown on the screen as the result of the game.

How test will be performed: After PVP game starts, the game should be considered as a lose if player 2's tank destroys player 1's home base during the 3-minute period. In this case, player 2 can be considered as winning without destroying player 1's tank three times during the 3-minute period.

6. GR-PPR-6

Type: Functional, Dynamic, Manual.

Initial State: After the PVP game starts.

Input: Player 1's tank destroys player 2's home base during the 3-minute period.

Output: A "Player 1 Win" is shown on the screen as the result of the game.

How test will be performed: After PVP game starts, the game should be considered as a win if player 1's tank destroys player 2's home base during the 3-minute period. In this case, player 1 can be considered as winning without destroying player 2's tank three times during the 3-minute period.

Test for PVE Rules

1. GR-PER-1

Type: Functional, Dynamic, Manual.

Initial State: After the PVE game starts.

Input: Player 1 and player 2 Enter the PVE game successfully.

Output: There is one home base shown on the PVE map located on the bottom of the PVE map.

How test will be performed: After PVE game starts, the system should provide one home base for the two players if they successfully Enter the PVE mode. The home base is an element that can possibly affect the result of the game.

2. GR-PER-2

Type: Functional, Dynamic, Manual.

Initial State: After the PVE game starts.

Input: Player 1 and player 2 Enter the PVE game successfully.

Output: There is 5 enemy tanks created on the PVE map after the PVE mode starts.

How test will be performed: After PVE game starts, the system should provide five enemy tanks and show them in random locations of the map.

3. GR-PER-3

Type: Functional, Dynamic, Manual.

Initial State: After the PVE game starts.

Input: Player 1 and player 2 Enter the PVE game successfully.

Output: Enemy tanks move by themselves.

How test will be performed: After PVE game starts, the system should allow the five enemy tanks to move automatically so that the player 1 and player 2 can distinguish the enemy tanks from their own tanks. The moving directions of enemy tanks are randomly chosen.

4. GR-PER-4

Type: Functional, Dynamic, Manual.

Initial State: After the PVE game starts.

Input: Player 1 shoots a bullet to the enemy tank using "J" key from the keyboard. The bullet hits the enemy tank.

Output: Enemy tanks is destroyed by the bullet.

How test will be performed: After PVE game starts, the enemy tank should be destroyed and disappeared if a bullet is shot by player 1's tank to the enemy tank using "J" key from the keyboard and hit the enemy tank.

5. GR-PER-5

Type: Functional, Dynamic, Manual.

Initial State: After the PVE game starts.

Input: Player 2 shoots a bullet to the enemy tank using "," key from the keyboard. The bullet hits the enemy tank.

Output: Enemy tanks is destroyed by the bullet.

How test will be performed: After PVE game starts, the enemy tank should be destroyed and disappeared if a bullet is shot by player 2's tank to the enemy tank using "," key from the keyboard and hit the enemy tank.

6. GR-PER-6

Type: Functional, Dynamic, Manual.

Initial State: After the PVE game starts.

Input: Player 1 and player 2 Enter the PVE game successfully.

Output: Enemy tanks shoot bullets automatically.

How test will be performed: After PVE game starts, the enemy tank should be able to shoot bullets automatically if player 1 and player 2 Enter the PVE game successfully. The shooting frequency of enemy tanks should be random. The shooting direction of enemy tanks should be the same as the moving direction.

7. GR-PER-7

Type: Functional, Dynamic, Manual.

Initial State: After the PVE game starts.

Input: A enemy tank is destroyed.

Output: The enemy tank relives.

How test will be performed: After PVE game starts, the enemy tank should be able to relive after it is destroyed by the players' tanks. Therefore the total number of enemy tanks should be 5 at all time.

Note: Since we do not have scoring mechanism, testing for scores will be redundant. Therefore we choose not to include any testing for scoring mechanism.

8. GR-PER-8

Type: Functional, Dynamic, Manual.

Initial State: After the PVE game starts.

Input: Players' home base is destroyed by the bullet shot by an enemy tank during the 3-minute period.

Output: A "Lose" is shown on the screen as the result of the game.

How test will be performed: After PVE game starts, the game should be considered as a lose if the enemy tanks destroys the players' home base by shooting bullet in 3 minutes. In this case, enemy tanks can be considered as winning without destroying player 1's and player 2's tanks three times each during the 3-minute period.

9. GR-PER-9

Type: Functional, Dynamic, Manual.

Initial State: After the PVE game starts.

Input: The enemy tanks destroys player 1's and player 2's tanks three times each by using bullets during the 3-minute period.

Output: A "Lose" is shown on the screen as the result of the game.

How test will be performed: After PVE game starts, the game should be considered as a lose if the enemy tanks destroys player 1's and player 2's tanks three times each using bullets during the 3-minute period. In this case, enemy tanks can be considered as winning without destroying player 1 and player 2's home base during the 3-minute period.

10. GR-PER-10

Type: Functional, Dynamic, Manual.

Initial State: The PVE game starts.

Input: Player 1 and player 2 participate the PVE game for 3 minutes. During the game, the home base is not destroyed and no player's tank is destroyed for 3 times.

Output: A "Win" is shown on the screen as the result of the game.

How test will be performed: After PVE game starts, the game should be considered as a win if the home base is not destroyed and no player's tank is destroyed for 3 times during the 3-minute period.

3.2 Tests for Nonfunctional Requirements

Note: The PVE and PVP mode are two different pygame systems. This is the reason why they have to be tested separately.

3.2.1 Area of Testing¹ for Look and Feel

Test for Visual Appearance

1. LF-VA-1

Type: Manual, dynamic, functional

Initial State: The game is at PVE mode. PVE map is loaded and the players have chose tanks.

Input/Condition: Players start the PVE game.

Output/Result: The screen shows the game in a 2-D drawing style.

How test will be performed: The software ~~developers~~ testers are asked to evaluate the drawing style to identify whether the game drawing style is in 2-D.

2. LF-VA-2

Type: Manual, dynamic, functional

Initial State: The game is at PVP mode. PVP map is loaded and the players have chose tanks.

Input/Condition: Players start the PVP game.

Output/Result: The screen shows the game in a 2-D drawing style.

How test will be performed: The software ~~developers~~ testers are asked to evaluate the drawing style to identify whether the game drawing style is in 2-D.

3. LF-VA-3

Type: Manual, dynamic, functional

Initial State: The game is installed on the computer.

Input/Condition: The game is just launched.

Output/Result: The screen shows the background of the game system in black.

How test will be performed: The software ~~developers~~ testers can see the background in black.

4. LF-VA-4

Type: Manual, dynamic, functional

Initial State: The game is at PVE mode. PVE map is loaded and the players have chose tanks.

Input/Condition: Players start the PVE game.

Output/Result: The screen shows the brick walls and iron walls in a grid map.

How test will be performed: The software ~~developers~~ testers can see the brick walls and iron walls in a grid map.

5. LF-VA-5

Type: Manual, dynamic, functional

Initial State: The game is at PVP mode. PVP map is loaded and the players have chose tanks.

Input/Condition: Players start the PVP game.

Output/Result: The screen shows the brick walls and iron walls in a grid map.

How test will be performed: The software ~~developers~~ testers can see the brick walls and iron walls in a grid map.

6. LF-VA-6

Type: Manual, dynamic, functional

Initial State: The game is at PVE mode. PVE map is loaded and the players have chose tanks.

Input/Condition: Players start the PVE game.

Output/Result: The screen shows brick walls in orange and iron walls consisted of 4 iron cube in sliver, while the screen displays a home which looks like a eagle.

How test will be performed: The software ~~developers~~ testers can see brick walls in orange and iron walls consisted of 4 iron cube in sliver, while the screen displays a home which looks like a eagle.

7. LF-VA-7

Type: Manual, dynamic, functional

Initial State: The game is at PVP mode. PVP map is loaded and the players have chose tanks.

Input/Condition: Players start the PVP game.

Output/Result: The screen shows brick walls in orange and iron walls consisted of 4 iron cube in sliver, while the screen displays a home which looks like a eagle.

How test will be performed: The software ~~developers~~ testers can see brick walls in orange and iron walls consisted of 4 iron cube in sliver, while the screen displays a home which looks like a eagle. The test will be considered as success.

8. LF-VA-8

Type: Manual, dynamic, functional

Initial State: The game is at PVE mode and the PVE map is loaded.

Input/Condition: The two players choose two different types of tanks among double-life tank, high-speed tank, double-bullet tank. Then the players start the PVE game.

Output/Result: The two different types of tanks show on the screen with different colours.

How test will be performed: The software ~~developers~~ ~~testers~~ can see the two different types of tanks on the screen with different colours. By repeating the above operation with different combination of the two types of tanks, the test should cover all possible conditions.

9. LF-VA-9

Type: Manual, dynamic, functional

Initial State: The game is at PVP mode. PVP map is loaded and the players have chose tanks.

Input/Condition: The two players choose two different types of tanks among double-life tank, high-speed tank, double-bullet tank. Then the players start the PVP game.

Output/Result: The two different types of tanks show on the screen with different colours.

How test will be performed: The software ~~developers~~ ~~testers~~ can see the two different types of tanks on the screen with different colours. By repeating the above operation with different combination of the two types of tanks, the test should cover all possible conditions.

10. LF-VA-10

Type: Manual, dynamic, functional

Initial State: The game is at PVE mode. PVE map is loaded and the players have chose tanks.

Input/Condition: The players start the PVE game.

Output/Result: The drawing style of TankWar game is very similar to the drawing style of the original game "Battle City".

How test will be performed: By surveying 50 test users, 80 percent of the test users consider the drawing style of ~~TankWall~~ TankWar shall be more than 90 percent similar with Battle City.

11. LF-VA-11

Type: Manual, dynamic, functional

Initial State: The game is at PVP mode. PVP map is loaded and the players have chose tanks.

Input/Condition: Players start the PVP game.

Output/Result: The drawing style of TankWar game is very similar to the drawing style of the original game "Battle City".

How test will be performed: By surveying 50 test users, 80 percent of the test users consider the drawing style of ~~TankWall~~ TankWar shall be more than 90 percent similar with Battle City.

3.2.2 Area of Testing² for Usability and Humanity

Test for Ease of Use

1. UH-EU-1

Type: Manual, dynamic, functional

Initial State: The game is at PVE mode. PVE map is loaded.

Input/Condition: The players start to choose tanks.

Output/Result: The screen displays the photos and descriptions of different types of tanks to give a brief introduction when users are choosing tanks.

How test will be performed: When the software ~~developer~~ tester are choosing tanks, there are the photos and descriptions are displayed on the screen.

2. UH-EU-2

Type: Manual, dynamic, functional

Initial State: The game is at PVP mode. PVP map is loaded ~~and the players have chose tanks.~~

Input/Condition: The players start to choose tanks.

Output/Result: The screen displays the photos and descriptions of different types of tanks to give a brief introduction when users are choosing tanks.

How test will be performed: When the software ~~developer~~ tester are choosing tanks, there are the photos and descriptions are displayed on the screen.

3. UH-EU-3

Type: Manual, dynamic, functional

Initial State: The game is at PVE mode. PVE map is loaded and the players have chose tanks.

Input/Condition: The players press "Enter" key to start the PVE game.

Output/Result: The rules of the game are displayed ~~for 3 seconds, then on the screen. If the players press the key "Enter", the game starts will start.~~

How test will be performed: After the software ~~developer~~ tester press "Enter" key to start the PVE game, the rules of the game are displayed ~~for 3 seconds. The users press the key "Enter".~~ Then, the game starts.

4. UH-EU-4

Type: Manual, dynamic, functional

Initial State: The game is at PVP mode. PVP map is loaded and the players have chose tanks.

Input/Condition: The players press "Enter" key to start the PVP game.

Output/Result: The rules of the game are displayed ~~for 3 seconds, then on the screen. If the players press the key "Enter", the game starts will start.~~

How test will be performed: After the software ~~developer~~ tester press "Enter" key to start the PVP game, the rules of the game are displayed ~~for 3 seconds~~. The users press the key "Enter". Then, the game starts.

Test for Learning and Understandability

1. UH-LU-1

Type: Manual, dynamic, functional

Initial State: The game is at PVE mode. PVE map is loaded and the players have chose tanks.

Input/Condition: The players press "Enter" key to start the PVE game. The players read the rules of the game and operation instruction displayed on the screen ~~for 3 seconds~~, then the game starts.

Output/Result: The players who are above 12 years old and have basic computer operation knowledge know how to play the game after read the rules and instruction.

How test will be performed: After the player read the rules of PVE and operation instruction, the players should have a basic knowledge. By surveying 50 test players who are above 12 years old and have basic computer operation knowledge, 80 percent of the test players knows how to play the game after read the rules and instruction.

2. UH-LU-2

Type: Manual, dynamic, functional

Initial State: The game is at PVP mode. PVP map is loaded and the players have chose tanks.

Input/Condition: The players press "Enter" key to start the PVP game. The players read the rules of the game and operation instruction displayed on the screen ~~for 3 seconds~~, then the game starts.

Output/Result: The players who are above 12 years old and have basic computer operation knowledge know how to play the game after read the rules and instruction.

How test will be performed: After the player read the rules of PVP and operation instruction, the players should have a basic knowledge. By surveying 50 test players who are above 12 years old and have basic computer operation knowledge, 80 percent of the test players knows how to play the game after read the rules and instruction.

3. UH-LU-3

Type: Manual, dynamic, functional

Initial State: The game is just launched on the computer.

Input/Condition: The players choose one of the 3 modes including PVP, PVE, map editing mode.

Output/Result: The players above 12 years old with basic computer operation knowledge would consider the words and icons showed on the screen are easily understandable.

How test will be performed: By surveying 50 test players who are above 12 years old and have basic computer operation knowledge, 80 percent of the test players would consider the words and icons showed on the screen are easily understandable.

3.2.3 Area of Testing³ for Performance

Test for Speed and Latency

1. PT-SL-1

Type: Manual, dynamic, functional

Initial State: The game is just launched and the game system is presented on the screen.

Input/Condition: The players could perform any operations in the game system.

Output/Result: The system shall respond to a user operation input less than 100 milliseconds.

How test will be performed: The software ~~developer~~ tester will use the python time library to calculate the response time of an operation.

When the players start performing the operations until the responds been made by the system, the respond time is counted. The system respond time is less than 100 milliseconds.

2. PT-SL-2

Type: Manual, dynamic, functional

Initial State: The game is in map editing mode and the player has created their own map.

Input/Condition: The player presses "Enter" key to save the map and names the map. Then the player presses "Enter" key again to start saving process.

Output/Result: The system shall take less than 5 second to save a map on the local disk.

How test will be performed: The software ~~developer~~ tester will use the python time library to calculate the time for saving a map. The time for saving a map is less than 5 second.

3. PT-SL-3

Type: Manual, dynamic, functional

Initial State: The game is in the PVP mode.

Input/Condition: The player chooses the map to be loaded.

Output/Result: The system take less than 5 second to load the PVP map from local disk.

How test will be performed: The software ~~developer~~ tester will use the python time library to calculate the time for loading a PVP map. The time for loading a PVP map is less than 5 second.

4. PT-SL-4

Type: Manual, dynamic, functional

Initial State: The game is in the PVE mode.

Input/Condition: The player chooses the map to be loaded.

Output/Result: The system take less than 5 second to load the PVE map from local disk.

How test will be performed: The software ~~developer~~ tester will use the python time library to calculate the time for loading a PVE map. The time for loading a PVE map is less than 5 second.

Test for Accuracy

1. PT-AT-1

Type: Manual, static, structural

Initial State: The game code is functional and is implemented in python.

Input/Condition: There are floating point numbers used in the code.

Output/Result: Floating point numbers used in the code is in double precision.

How test will be performed: The floating point numbers in the source code is in the standard floating point number data type of python.

Test for Reliability and Availability

1. PT-RA-1

Type: Manual, dynamic, functional

Initial State: The game is launched.

Input/Condition: The two players are both engaged in the game.

Output/Result: The system crashes do not exceed 3 times within 6-hour constantly running.

How test will be performed: By starting a test, 10 TankWar games are constantly played by 20 test players in group of 2 for 6 hour, while crashing happens less than 3 time each game.

2. PT-RA-2

Type: Manual, dynamic, functional

Initial State: The game is in the local computer. The system is installed python and pygame.

Input/Condition: The game is launched on the flowing operating system including Windows 7, Windows 8, Windows 10, Mac OS 10.8+, and Linux.

Output/Result: The game is able to run on ~~Windows 7, Windows 8,~~ Windows 10, Mac OS 10.8, and Linux.

How test will be performed: The game will be installed on the computers with the specified operating systems respectively. Then, the game will be tested to determine whether the game is fully functional on each operating system.

Test for Robustness

1. PT-RT-1

Type: Functional, Dynamic, Manual.

Initial State: The game is launched.

Input: The keys besides the desired input keys mentioned in the functional requirements are triggered.

Output: The program would ignore the undesired game input and only identify the desired input keys.

How test will be performed: In each screen, the three testers randomly press 100 input keys other than the desired input keys. If the game performs any action, the test is considered as fail. If no action is performed, the test is considered as pass.

~~Type: Functional, Dynamic, Manual.~~

~~Initial State: The player is in map editing mode and a map is created.~~

~~Input: Press the "Enter" key, then Enter an existing file name and press "Enter" key again.~~

~~Output: The program prompts the user and says "File Already Exist".~~

~~How test will be performed: Create a map in map editing mode, then press "Enter" key and Enter the file name. Press the "Enter" again and check if the program prompts the users and says "File Already Exist".~~

2. ~~PT-RT-2~~

~~Type: Functional, Dynamic, Manual.~~

~~Initial State: The player is in map editing mode and a map is created.~~

~~Input: Press the "Enter" key, then Enter an existing file name and press "Enter" key again.~~

~~Output: The program prompts the user and says "File Already Exist".~~

~~How test will be performed: Create a map in map editing mode, then press "Enter" key and Enter the file name. Press the "Enter" again and check if the program prompts the users and says "File Already Exist".~~

Test for Capacity

1. PT-CT-1

Type: Manual, dynamic, functional

Initial State: The game is in the PVE mode. The PVE map is loaded and the players has chosen their tanks.

Input/Condition: The PVE game is started and two players both perform their operation to the tanks at the same time.

Output/Result: The game is able to support the two players' operations at the same time.

How test will be performed: The two players both perform their operation to the tanks at the same time. The game is able to support the two players' operations at the same time.

2. PT-CT-2

Type: Manual, dynamic, functional

Initial State: The game is in the PVP mode. The PVP map is loaded and the players has chosen their tanks.

Input/Condition: The PVP game is started and two players both perform their operation to the tanks at the same time.

Output/Result: The game is able to support the two players' operations at the same time.

How test will be performed: The two players both perform their operation to the tanks at the same time. The game is able to support the two players' operations at the same time.

3.2.4 Area of Testing⁴ for Operational and Environmental

Test for Physical Environment

1. OE-PE-1

Type: Manual, dynamic, functional

Initial State: The game is installed on the computer with 2-cores processor and 1GB RAM. Python and pygame is intalled too.

Input/Condition: The players launch and play the game.

Output/Result: The game is fully functional on the computer.

How test will be performed: The players play the PVE and PVP games and the game is fully functional.

Test for Productization and Release

1. OE-PR-1

Type: Manual, functional

Initial State: The game is fully functional and is developed well.

Input/Condition: The product is in the delivery state.

Output/Result: The product is wrap up as a file package and can be easily downloaded from Git Lab as an open source project.

How test will be performed: The product is in a file package and can be easily downloaded from Git Lab as an open source project.

2. OE-PR-2

Type: Manual, static, structural

Initial State: The game is developed and before releasing.

Input/Condition: The game is in the testing process.

Output/Result: The system is released after full verifying and validation.

How test will be performed: The test developer conducts an inspection followed by the requirements specification and test plan to fully verify and validate the game.

3.2.5 Area of Testing⁵ for Maintainability and Support

Test for Supportability

1. MS-ST-1

Type: Manual, static, structural

Initial State: The project state is in the software development process and code implementation.

Input/Condition: The naming convention is needed in the software development. The variables in the code need to be created in the code implementation.

Output/Result: The naming convention of the variables and development process are consistent.

How test will be performed: The software ~~developer~~ **tester** would conduct a inspection on the code and development documentation to check the naming consistence.

2. MS-ST-2

Type: Manual, static, structural

Initial State: The code of the game is implementing.

Input/Condition: The software developer is implementing the code.

Output/Result: The code implementation include the comments that generates Doxygen documentation.

How test will be performed: The Doxygen documentation is generated and have all the important information about the functions, calls, and variables.

3.2.6 Area of Testing6 for Security

Test for Integrity and Privacy

1. ST-IP-1

Type: Manual, dynamic, functional

Initial State: The game is in the map editing mode. The map is created.

Input/Condition: The players try to save the map.

Output/Result: The map is safely saved on the local disk and there is no loss of the information of the map.

How test will be performed: The software ~~developer~~ ~~tester~~ would design and save 10 maps in either PVE and PVP map creation. The maps saved on the local disk should have the information as the software ~~developer~~ ~~tester~~ design and there is no information loss.

2. ST-IP-2

Type: Manual, dynamic, functional

Initial State: The game is developed well.

Input/Condition: The game is in testing process.

Output/Result: There is no any mechanism to save personal information from users.

How test will be performed: The software ~~developers~~ ~~testers~~ would conduct a walk-through on the code to exam whether there is mechanism to collect any personal information from users.

3.2.7 Area of Testing⁷ for Cultural

Test for Cultural

1. CT-CT-1

Type: Manual, dynamic, functional

Initial State: The game is developed well and fully functional.

Input/Condition: The game is in the testing process.

Output/Result: The game system is in English with Canadian spelling. The game does not contain any content that could be considered offensive.

How test will be performed: The software ~~developer~~ **tester** would conduct a walk-through on the code to check the language and spelling to make sure no offensive contents in the game.

3.3 Traceability Between Test Cases and Requirements

The traceability between test cases and requirement is presented in the following two table, including Table 4: Taceability Matrix for Functional Requirement, and and Table 5: Taceability Matrix for Non-functional Requirement.

Functional Requirement ID	Priority	Test Case ID
FR1	5	GPT-PPMS-1, GPT-PEMS-1, GPT-MEMS-1
FR2	5	ME-PVPAB-1, ME-PVPAB-2, ME-PVPAB-3, ME-PVPAB-4, ME-PVPAI-1, ME-PVPAI-2, ME-PVPAI-3, ME-PVPAI-4, ME-PVPDW-1, ME-PVPDW-2, ME-PVPDW-3, ME-PVEAB-1, ME-PVEAB-2, ME-PVEAB-3, ME-PVEAB-4, ME-PVEAI-1, ME-PVEAI-2, ME-PVEAI-3, ME-PVEAI-4, ME-PVEDW-1, ME-PVEDW-2, ME-PVEDW-3
FR3	4	ME-PVPSM-1, ME-PVESM-1
FR4	5	MT-GU-1, MT-GU-2, MT-GU-3, MT-GU-4, MT-GU-5, MT-GU-6, MT-GU-7, MT-GU-8, MT-GD-1, MT-GD-2, MT-GD-3, MT-GD-4, MT-GD-5, MT-GD-6, MT-GD-7, MT-GD-8, MT-GL-1, MT-GL-2, MT-GL-3, MT-GL-4, MT-GL-5, MT-GL-6, MT-GL-7, MT-GL-8, MT-GR-1, MT-GR-2, MT-GR-3, MT-GR-4, MT-GR-5, MT-GR-6, MT-GR-7, MT-GR-8.
FR5	5	TAT-UUS-1, TAT-UUS-2, TAT-UUS-3, TAT-UUS-4

Functional Requirement ID	Priority	Test Case ID
FR6	5	TAT-SB-1, TAT-SB-2, TAT-SB-3, TAT-SB-4, TAT-SB-5, TAT-SB-6, TAT-SB-7, TAT-SB-8
FR7	5	GR-GRW-1, GR-GRW-2, GR-GRW-3, GR-GRW-4
FR8	5	GR-GRW-5, GR-GRW-6, GR-GRW-7, GR-GRW-8
FR9	5	MT-WB-1, MT-WB-2, MT-WB-3, MT-WB-4, MT-WB-5, MT-WB-6
FR10	5	GPT-PPTS-1, GPT-PPTS-2, GPT-PETS-1, GPT-PETS-2
FR11	5	GR-PPR-2, GR-PPR-3, GR-PPR-4, GR-PPR-5, GR-PPR-6, GR-PER-8, GR-PER-9, GR-PER-10
FR12	5	GR-GRB-1 , GR-GRB-2, GR-GRB-3 , GR-GRB-4 , GR-GRB-5 , GR-GRB-6, GR-GRB-7, GR-GRB-8
FR13	5	GR-PER-3
FR14	5	GR-PER-4, GR-PER-5
FR15	5	GR-PER-7
FR16	5	GR-PER-2, GR-PER-7
FR17	5	GR-PER-1
FR18	5	GR-PPR-1
FR19	5	GR-PER-8, GR-PER-9, GR-PER-10
FR20	5	GR-PPR-2, GR-PPR-3, GR-PPR-4, GR-PPR-5, GR-PPR-6
FR21	5	GR-GRL-1, GR-GRL-2, GR-GRL-3, GR-GRL-4

Table 4: Traceability Matrix for Functional Requirement

Non-Functional Requirement ID	Priority	Test Case ID
LF1	5	LF-VA-1, LF-VA-2
LF2	5	LF-VA-3
LF3	4	LF-VA-4, LF-VA-5
LF4	5	LF-VA-6, LF-VA-7
LF5	5	LF-VA-6, LF-VA-7
LF6	5	LF-VA-8, LF-VA-9
LF7	5	LF-VA-6, LF-VA-7
LF8	5	LF-VA-10, LF-VA-11
UH1	5	UH-EU-1, UH-EU-2
UH2	5	UH-EU-3, UH-EU-4
UH4	5	UH-LU-1, UH-LU-2
UH5	5	UH-LU-3
PR1	5	PT-SL-1
PR2	5	PT-SL-2, PT-SL-3, PT-SL-4
PR4	5	PT-AT-1
PR5	5	PT-RA-1
PR6	5	PT-RA-2
PR7	5	PT-RT-1. PT-RT-2
PR8	5	PT-CT-1, PT-CT-2
OE1	5	OE-PE-1
OE3	5	OE-PR-1
OE4	5	OE-PR-1
OE5	5	OE-PR-2
MS2	5	MS-ST-1
MS3	5	MS-ST-2
MS4	5	PT-RA-2
SR2	5	ST-IP-1
SR3	5	ST-TP-2
CR1	5	CT-CT-1
CR2	5	CT-CT-1

Table 5: Traceability Matrix for Non-functional Requirement

4 Tests for Proof of Concept

4.1 Area of Testing1 for Map Saving

Map Match Verification

1. MS-MV-1

Type : Functional, Static, Automated.

Initial State: A map is designed follow a certain known map file and saved in the map editing mode.

Input: The saved file for the map.

Output: A Boolean value stands for the equality between the saved map file and the known map file.

How test will be performed: After designing and saving a map which means to be the same as a certain known map file, write a unit test function to compare the two files and check if the test passes.

2. MS-MV-2

Type : Functional, Dynamic, Manual.

Initial State: A map is designed follow a certain known map and saved in the map editing mode.

Input: Load the saved map at the process of loading map.

Output: The GUI shows the saved map.

How test will be performed: After designing and saving a map which means to be the same as a certain known map, loading the map to the game and check if the screen shows the expected map.

4.2 Area of Testing2 for the Test coverage

Test the coverage of the tests

1. TC-TC-1

Type: Functional, Static, Manual.

Initial State: A finished test plan.

Input: All the group members inspect the test plan.

Output: The test plan covers all the requirements and the code coverage should be at least 80 percent.

How test will be performed: After the test plan is finished, all the three group members will inspect the document, and check if it covers all the requirements in SRS and also to see if the code coverage is satisfied.

4.3 Area of Testing for the User Acceptance

Test the user acceptance

1. UA-UA-1

Type: Functional, Dynamic, Manual.

Initial State: A running new game.

Input: Let the sample users try this new game.

Output: The acceptance of the game from the sample users.

How test will be performed: After the game is correctly running, some 50 sample users who are above 12 years old will be invited to try this game and ask them to finish a survey designed for the acceptance. The test will pass if the user acceptance is higher than 80 percent.

5 Comparison to Existing Implementation

None.

6 Unit Testing Plan

6.1 Unit testing of internal functions

Python unittest will be used to test the internal functions. To finish the test, several inputs which mean to cover all the cases will be designed for each requirement, and the outputs will be compared with the expected outputs to see if the requirements are met.

1. Test Name: UBT-SS-1
Type : Functional, Static, Automated.
Initial State: A bullet object is created.
Input: An integer represents the bullet's speed.
The bullet's speed is changed to the input integer.
How test will be performed: A unit test case will be wrote to compare the output with the expected output, and the test will pass if they are equivalent.
2. Test Name: UBT-SL-1
Type : Functional, Static, Automated.
Initial State: A bullet object is created.
Input: A boolean value represents the bullet's life.
Output: The bullet's life is changed to the boolean value.
How test will be performed: A unit test case will be wrote to compare the output with the expected output, and the test will pass if they are equivalent.
3. Test Name: UBT-SST-1
Type : Functional, Static, Automated.
Initial State: A bullet object is created.
Input: A boolean value represents that the bullet is strong or not.
Output: The bullet's value of strong is changed to the boolean value.
How test will be performed: A unit test case will be wrote to compare the output with the expected output, and the test will pass if they are equivalent.
4. Test Name: UDT-INIT-1
Type : Functional, Static, Automated.
Initial State: A prepared decTime module.

Input: Create a decTime object with an input integer represents the number of seconds.

Output: The number of seconds are changed into the form of hours:minutes:seconds.

How test will be performed: A unit test case will be wrote to compare the output with the expected output, and the test will pass if they are equivalent.

5. Test Name: UFD-CH-1

Type : Functional, Static, Automated.

Initial State: A food object is created.

Input: Call the function change.

Output: The value of life is changed to True.

How test will be performed: A unit test case will be wrote to compare the output with the expected output, and the test will pass if they are equivalent.

6. Test Name: UET-ST-1

Type : Functional, Static, Automated.

Initial State: An EnemyTank object is created.

Input: Call the function shoot.

Output: The value of the tank's bullet's life is changed to True.

How test will be performed: A unit test case will be wrote to compare the output with the expected output, and the test will pass if they are equivalent.

7. Test Name: UMT-ST-1

Type : Functional, Static, Automated.

Initial State: An MyTank object is created.

Input: Call the function shoot.

Output: The value of the tank's bullet's life is changed to True.

How test will be performed: A unit test case will be wrote to compare the output with the expected output, and the test will pass if they are equivalent.

8. Test Name: UMT-LU-1 & UMT-LU-2

Description: These two tests are similar. The first one is to increase the level by one, and the second one is to test when the level is already maximized.

Type : Functional, Static, Automated.

Initial State: An MyTank object is created.

Input: Call the function levelup.

Output: Nothing is changed if tank's level is 2, otherwise the level is increased by 1.

How test will be performed: A unit test case will be wrote to compare the output with the expected output, and the test will pass if they are equivalent.

9. Test Name: UMT-LD-1 & UMT-LD-2

Description: These two tests are similar. The first one is to decrease the level by one, and the second one is to test when the level is already minimized.

Type : Functional, Static, Automated.

Initial State: An MyTank object is created.

Input: Call the function leveldown.

Output: Nothing is changed if tank's level is 0, otherwise the level is decreased by 1, and if the level is decreased to 0, the tank's speed is changed to 6 and the bullet's value of strong is changed to False.

How test will be performed: A unit test case will be wrote to compare the output with the expected output, and the test will pass if they are equivalent.

10. Test Name: UDT-BPS-1
Type : Functional, Static, Automated.
Initial State: An doubleLifeTank object is created.
Input: Call the function bulletproof_start.
Output: The value of the bullet proof is changed to True.
How test will be performed: A unit test case will be wrote to compare the output with the expected output, and the test will pass if they are equivalent.
11. Test Name: UDT-BPE-1
Type : Functional, Static, Automated.
Initial State: An doubleLifeTank object is created.
Input: Call the function bulletproof_end.
Output: The value of the bullet proof is changed to False.
How test will be performed: A unit test case will be wrote to compare the output with the expected output, and the test will pass if they are equivalent.
12. Test Name: UDT-GBP-1
Type : Functional, Static, Automated.
Initial State: An doubleLifeTank object is created.
Input: Call the function get_bullet_proof.
Output: The value of the bullet proof is returned.
How test will be performed: A unit test case will be wrote to compare the output with the expected output, and the test will pass if they are equivalent.
13. Test Name: UFT-DB-1
Type : Functional, Static, Automated.
Initial State: An fastBulletTank object is created.

Input: Call the function `double_bullet`.

Output: The two bullets' life are both changed to `True`.

How test will be performed: A unit test case will be wrote to compare the output with the expected output, and the test will pass if they are equivalent.

14. Test Name: UHT-LS-1

Type : Functional, Static, Automated.

Initial State: An `highSpeedTank` object is created.

Input: Call the function `leap_start`.

Output: The tank's speed is increased by 3.

How test will be performed: A unit test case will be wrote to compare the output with the expected output, and the test will pass if they are equivalent.

15. Test Name: UHT-LE-1

Type : Functional, Static, Automated.

Initial State: An `highSpeedTank` object is created.

Input: Call the function `leap_end`.

Output: The tank's speed is changed to 3.

How test will be performed: A unit test case will be wrote to compare the output with the expected output, and the test will pass if they are equivalent.

6.2 Unit testing of output files

The outputs of our project include the saved maps designed by the players and the GUI of the game. The saved maps will be tested by writing a unit test function to compare the saved map file with the expected map file to see if they are equivalent. For the GUI part, it will be difficult to do the automated testing so that manual tests will be applied to check if the GUI shows the right thing.

7 Appendix

7.1 Symbolic Parameters

None.

7.2 Usability Survey Questions?

1. After reading the rules of PVE game and the operation instruction, do you know how to control a tank?
 - a. Yes
 - b. No
2. After reading the rules of PVE game and the operation instruction, do you know how to win the PVE game?
 - a. Yes
 - b. No
3. After reading the rules of PVE game and the operation instruction, ~~how much parentage that you can understand~~ how much percentage do you understand the rules of the PVE game?
 - a. 0%
 - b. 20%
 - c. 40%
 - d. 60%
 - e. 80%
 - f. 100%
4. After reading the rules of PVP game and the operation instruction, do you know how to control a tank?
 - a. Yes
 - b. No
5. After reading the rules of PVP game and the operation instruction, do you know how to win the PVP game?

- a. Yes
 - b. No
6. After reading the rules of PVP game and the operation instruction, ~~how much parentage that you can understand~~ how much percentage do you understand the rules of the PVP game?
- a. 0%
 - b. 20%
 - c. 40%
 - d. 60%
 - e. 80%
 - f. 100%
7. ~~For the words and icons showed in the game, how much percentage that you think that~~ How much percentage do you understand the words and icons showed in the game?
- a. 0%
 - b. 20%
 - c. 40%
 - d. 60%
 - e. 80%
 - f. 100%
8. ~~How much parentage that you can understand~~ How much percentage would you consider that it is easy to play the game?
- a. 0%
 - b. 20%
 - c. 40%
 - d. 60%
 - e. 80%
 - f. 100%

9. How much parentage would you consider that the drawing style of TankWar is similar to the original game Battle City?
- a. 0%
 - b. 20%
 - c. 40%
 - d. 60%
 - e. 80%
 - f. 100%