IZMIR UNIVERSITY OF ECONOMICS

Department of Software Engineering

2017-2018 Spring

SE 497

SOFTWARE TEST DOCUMENTATION

Bora Armağan Koyuncu

Doğa Ünal

Sinan Küçükyılmaz

1. INTRODUCTION

1.1 Document identifier

This document shows the methods and results used to test our game.

1.2 Scope

The purpose of this document is to provide the test cases of TOWIN Beast. This document gives information the objective, scenario, expected outcomes, procedural requirements and requirements type for each test case. This has allowed us to simulate the problems/bugs that the real players can have with our game.

2. DETAILS FOR SYSTEM TEST PLAN

These parts provide describe test items; use cases and test cases, traceability matrix and evaluation criteria for pass/fail test.

2.1 Test items and their identifiers

In this game, we use black box testing methods. We manually tested the use cases. use cases were tested by random people from our social network etc... The results of the test cases were determined by whether the function passed the tests. And at some point, we also used manual control to test some conditions by ourselves.

2.1.1 Use Cases

1- **Movement:** The player can move the character vertically / horizontally.
2- **Killing:** The character can kill enemy via gun shot.
3- **Get damage:** The character can get damage from enemy and environmental objects.
4- **Interaction with vending machine:** The character can interact to the vending machine
5- **Buy Weapon:** The character can buy weapons from vending machine.
6- **Collect coin:** The character can collect coins from the ground.
7- **Door interaction:** The door should be activated when character come close to door.
8- **Jetpack activation:** Character can fly with jetpack

2.1.2 Test Cases

Test Case 1: Move

Test Case 2: Shoot the enemy

Test Case 3: Use the vending machine

Test Case 4: Collect a heart

Test Case 5: Collect a coin

Test Case 6: Open the door

Test Case 7: Get hurt

Test Case 8: Buy a gun

Test Case 9: Get damage from laser traps

Test Case 10: Fly with jetpack

Test Case 11: Key keeper

Test Case 12: Key keeper placement

Test Case 13: Obstacle tiles

Test Case 14: Health regeneration

Test Case 15: Sound of picking coin

Test Case 16: Sound of gun

Test Case 17: Sound of hit

Test Case 18: Get damage from spine traps

Test Case 19: Reach the boss

Test Case 20: Get damage from boss

Test Case 21: Give damage to boss

Test Case 22: Kill the boss

Test Case 23: Dead

Test Case 24: Get damage from turret

Test Case 25: The Boss' second phase activation

Test Case 26: The Boss gets rest

Test Case 27: The Boss' fires his gun

Test Case 28: Restart

Test Case 29: Key binding to shooting action

2.2 Test Traceability Matrix

Traceability Matrix	Use Case 1	Use Case 2	Use Case3	Use Case4	Use Case 5	Use Case 6	Use Case 7	Use Case 8
Test Case 1	✓							
Test Case 2		✓						
Test Case 3				✓	✓			
Test Case 4	✓							
Test Case 5	✓					✓		
Test Case 6							✓	
Test Case 7			✓					
Test Case 8				✓	✓			
Test Case 9			✓					
Test Case 10	✓							✓
Test Case 11							✓	
Test Case 15						✓		
Test Case 18			✓					
Test Case 19	✓						✓	
Test Case 20			✓					
Test Case 21		✓						

Test Case 22		✓						
Test Case 23			✓					
Test Case 24			✓					

2.3 Item pass/fail criteria

The tested items are considered to pass the test if the output is exactly the same as test cases' output

3. TEST CASE DETAILS

This section will explain the detailed information about the test case. Each test case includes objective, input, outcome, environmental needs, special procedural requirements and requirement type.

Test Case ID	1
Objective	Move
Input	Left arrow button, right arrow button
Outcome	The game character moves in the ground.
Environmental needs	Gamepad for Windows OS
Special procedural	User should connect gamepad to the PC.
Type	Functional

Test Case ID	2
Objective	Shoot the enemy

Input	Button X
Outcome	Character shoots fire.
Environmental needs	Gamepad for Windows OS
Special procedural	User should connect gamepad to the PC.
Type	Functional

Test Case ID	3
Objective	Use the vending machine
Input	Button A
Outcome	Vending machine buying menu pop up on screen
Environmental needs	Gamepad for Windows OS
Special procedural	User should connect gamepad to the PC.
Type	Functional

Test Case ID	4
Objective	Collect a heart
Input	Left arrow button, right arrow button, button A

Outcome	The character collects heart from the ground
Environmental needs	Gamepad for Windows OS
Special procedural	User should connect gamepad to the PC.
Type	Functional

Test Case ID	5
Objective	Collect a coin
Input	Left arrow button, right arrow button, button A
Outcome	The character collects coin from the ground
Environmental needs	Gamepad for Windows OS
Special procedural	User should connect gamepad to the PC.
Type	Functional

Test Case ID	6
Objective	Open the door
Input	Interaction with gate, button A
Outcome	The character opens the door
Environmental needs	Gamepad for Windows OS

Special procedural	User should connect gamepad to the PC.
Type	Functional

Test Case ID	7
Objective	Get hurt
Input	Interaction with enemy's bullet
Outcome	The character gets damage, loss some of his health points
Environmental needs	Gamepad for Windows OS
Special procedural	User should connect gamepad to the PC.
Type	Functional

Test Case ID	8
Objective	Buy a gun
Input	Interaction with vending machine, button A, up arrow button, down arrow button
Outcome	The character picks one of gun and buys it
Environmental needs	Gamepad for Windows OS
Special procedural	User should connect gamepad to the PC.
Type	Functional

Test Case ID	9
Objective	Get damage from laser traps
Input	Interaction with laser, button A, left arrow button, right arrow button
Outcome	The character gets damage from laser and loss some of his health points
Environmental needs	Gamepad for Windows OS
Special procedural	User should connect gamepad to the PC.
Type	Functional

Test Case ID	10
Objective	Fly with jetpack
Input	Button A
Outcome	The character flies up through the level
Environmental needs	Gamepad for Windows OS
Special procedural	User should connect gamepad to the PC.
Type	Functional

Test Case ID	11
Objective	Key keeper
Input	
Outcome	The key keeper should be behind one of the doors.
Environmental needs	Gamepad for Windows OS
Special procedural	User should connect gamepad to the PC.
Type	Functional

Test Case ID	12
Objective	Key keeper's placement
Input	
Outcome	The key keeper should have assigned behind one of the doors randomly when level started.
Environmental needs	Gamepad for Windows OS
Special procedural	User should connect gamepad to the PC.
Type	Functional

Test Case ID	13

Objective	Obstacle tiles
Input	
Outcome	The character should not be able to pass through the tiles if tiles' type is obstacle
Environmental needs	Gamepad for Windows OS
Special procedural	User should connect gamepad to the PC.
Type	Functional

Test Case ID	14
Objective	Health regeneration
Input	Collecting hearth from the ground
Outcome	The character's health point should increase
Environmental needs	Gamepad for Windows OS
Special procedural	User should connect gamepad to the PC.
Type	Functional

Test Case ID	15
Objective	Sound of picking coin
Input	Collecting coin from the ground

Outcome	The collecting coin sound should play
Environmental needs	Gamepad for Windows OS
Special procedural	User should connect gamepad to the PC.
Type	Functional

Test Case ID	16
Objective	Sound of gun
Input	Shooting with gun
Outcome	The firing sound should play
Environmental needs	Gamepad for Windows OS
Special procedural	User should connect gamepad to the PC.
Type	Functional

Test Case ID	17
Objective	Sound of hit
Input	Get hit
Outcome	The getting hurt sound should play
Environmental needs	Gamepad for Windows OS
Special procedural	User should connect gamepad to the PC.
Type	Functional

Test Case ID	18
Objective	Get damage from spine traps
Input	İnteraction with a spine trap
Outcome	The character gets damage
Environmental needs	Gamepad for Windows OS
Special procedural	User should connect gamepad to the PC.
Type	Functional

Test Case ID	19
Objective	Reach the boss
Input	İnteraction with the boss' room's door
Outcome	The character dives into the boss's room
Environmental needs	Gamepad for Windows OS
Special procedural	User should connect gamepad to the PC.
Type	Functional

Test Case ID	20

Objective	Get damage from the boss
Input	Gets hit from the boss
Outcome	The character gets hit from the boss
Environmental needs	Gamepad for Windows OS
Special procedural	User should connect gamepad to the PC.
Type	Functional

Test Case ID	21
Objective	Give damage to the boss
Input	Gives hit to the boss
Outcome	The character gives successful hit to the boss
Environmental needs	Gamepad for Windows OS
Special procedural	User should connect gamepad to the PC.
Type	Functional

Test Case ID	22
Objective	Kill the boss
Input	Gives many hits to the boss

Outcome	The character kills the boss
Environmental needs	Gamepad for Windows OS
Special procedural	User should connect gamepad to the PC.
Type	Functional

Test Case ID	23
Objective	Dead
Input	Gets many damage
Outcome	The character dies
Environmental needs	Gamepad for Windows OS
Special procedural	User should connect gamepad to the PC.
Type	Functional

Test Case ID	24
Objective	Get damage from turret
Input	Gets hits from a turret
Outcome	The character gets damage
Environmental needs	Gamepad for Windows OS
Special procedural	User should connect gamepad to the PC.

Type	Functional

Test Case ID	25
Objective	The Boss' second phase activation
Input	Beats up the boss' first phase
Outcome	The boss evolves to his second phase
Environmental needs	Gamepad for Windows OS
Special procedural	User should connect gamepad to the PC.
Type	Functional

Test Case ID	26
Objective	The Boss gets rest
Input	The boss fires many times.
Outcome	The boss gets exhausted and rests for a while.
Environmental needs	Gamepad for Windows OS
Special procedural	User should connect gamepad to the PC.
Type	Functional

Test Case ID	27
Objective	The Boss' fires his gun
Input	
Outcome	The boss fires his gun.
Environmental needs	Gamepad for Windows OS
Special procedural	User should connect gamepad to the PC.
Type	Functional

Test Case ID	28
Objective	Restart
Input	
Outcome	The game starts again
Environmental needs	Gamepad for Windows OS
Special procedural	User should connect gamepad to the PC.
Type	Functional

Test Case ID	29
Objective	Key binding to shooting action
Input	

Outcome	Finding the most comfortable button to shooting with a gun
Environmental needs	Gamepad for Windows OS
Special procedural	User should connect gamepad to the PC.
Type	Non-functional

4. SYSTEM TEST REPORT DETAILS

4.1 Overview of test results

In this section of the document, the results of the test cases were given. Game has been functioning properly. This means that the game has passed the tests successfully. However, these results cannot prove that the game works perfectly.

4.1.1 Test Environments

Our project is developed on Windows 7. Our programming language preference is C# and our game engine preference is Unity 2D.

- Windows 10, 8 GB RAM, 4.0 GHz Processor and 3 GB video card

- Windows 10, 16 GB RAM, 3.3 GHz Processor

- Windows 7, 4 GB RAM, 2.3 GHz Processor

4.2 Detailed test results

Detailed test results summary has been given in the following table.

Move	PASSED
Shoot the enemy	PASSED
Use the vending machine	PASSED
Collect a heart	PASSED

Collect a coin	PASSED
Open the door	PASSED
Get hurt	PASSED
Buy a gun	PASSED
Get damage from laser traps	PASSED
Fly with jetpack	PASSED
Key keeper	PASSED
Key keeper placement	PASSED
Obstacle tiles	PASSED
Health regeneration	PASSED
Sound of picking coin	PASSED
Sound of gun shoot	PASSED
Sound of hit	PASSED

Get damage from spine traps	PASSED
Reach the boss	PASSED
Get damage from boss	PASSED
Give damage to boss	PASSED
Kill the boss	PASSED

Dead	PASSED
Get damage from turret	PASSED
The Boss' second phase activation	PASSED
The Boss gets rest	PASSED
The Boss' fires his gun	PASSED
Restart	PASSED
Key binding to shooting action	PASSED

4.3 Rationale for decisions

Black box testing technique and visual inspection are used for the
tests.

4.4 Conclusions and recommendations

In this document, definition of test cases is given so that making sure that game is working properly and meeting the requirements. The game is expected to pass all those test cases in order to be able to be ready to play.