



Hedef4Yıldız

Onur Can Üner
Gürkan Karahan
Kader Belli
Serkan Tuna

REVENGE

System Test Documentation

Table of Contents

Table of Figures	4
Revision History	5
1 INTRODUCTION	6
1.1 Document Identifier	6
1.2 Scope	6
1.2.1 User Interface	6
1.2.2 Kinect Control	6
1.2.3 Network	6
1.2.4 Database	6
1.3 References	7
1.4 Level in the Overall Sequence	7
1.5 Test Classes and Overall Test Conditions	7
2 DETAILS FOR SYSTEM TEST PLAN	7
2.1 Test Items and Their Identifiers	7
2.2 Test Traceability Matrix	8
2.3 Features to Be Tested	9
2.4 Features not to Be Tested	9
2.5 Approach	10
2.6 Item Pass/Fail Criteria	10
2.7 Test Deliverables	10
3 TEST MANAGEMENT	10
3.1 Planned Activities and Task, Test Progression	10
3.2 Environment and Infrastructure	11
4 TEST CASE DETAILS	11
4.1 Interface Test Cases	12
4.1.1 Interface Test 1	12
4.1.2 Interface Test 2	12
4.1.3 Interface Test 3	13
4.1.4 Interface Test 4	13
4.1.5 Interface Test 5	14
4.1.6 Interface Test 6	14
4.1.7 Interface Test 7	15

4.1.8	Interface Test 8.....	15
4.1.9	Interface Test 9.....	16
4.1.10	Interface Test 10.....	16
4.1.11	Interface Test 11.....	17
4.1.12	Interface Test 12.....	17
4.2	Training Mode Test Cases.....	17
4.2.1	Training Test 1.....	17
4.2.2	Training Test 2.....	18
4.2.3	Training Test 3.....	18
4.2.4	Training Test 4.....	19
4.2.5	Training Test 5.....	19
4.2.6	Training Test 6.....	20
4.2.7	Training Test 7.....	20
4.2.8	Training Test 8.....	21
4.3	Multiplayer Mode Test Cases.....	21
4.3.1	Multiplayer Test 1.....	21
4.3.2	Multiplayer Test 2.....	22
4.3.3	Multiplayer Test 3.....	22
4.3.4	Multiplayer Test 4.....	23
4.3.5	Multiplayer Test 5.....	23
4.3.6	Multiplayer Test 6.....	24
4.3.7	Multiplayer Test 7.....	24
4.3.8	Multiplayer Test 8.....	25
4.3.9	Multiplayer Test 9.....	26
4.3.10	Multiplayer Test 10.....	26
4.3.11	Multiplayer Test 11.....	27
4.3.12	Multiplayer Test 12.....	27
4.3.13	Multiplayer Test 13.....	28
4.3.14	Multiplayer Test 14.....	28
5	SYSTEM TEST REPORT DETAILS.....	29
5.1	Overview of Test Results.....	29
5.2	Detailed Test Results.....	29
5.2.1	Interface Test Cases.....	29

5.2.2 Training Mode Test Cases..... 30

5.2.3 Multiplayer Mode Test Cases 30

5.3 Rationale for Decisions..... 30

5.4 Conclusions and Recommendations 31

Table of Figures

Table 1 – Test Traceability Matrix	8
Table 2 – Test Case: Interface Test 1	12
Table 3 – Test Case: Interface Test 2	12
Table 4 – Test Case: Interface Test 3	13
Table 5 – Test Case: Interface Test 4	13
Table 6 – Test Case: Interface Test 5	14
Table 7 – Test Case: Interface Test 6	14
Table 8 – Test Case: Interface Test 7	15
Table 9 – Test Case: Interface Test 8	15
Table 10 – Test Case: Interface Test 9	16
Table 11 – Test Case: Interface Test 10	16
Table 12 – Test Case: Interface Test 11	17
Table 13 – Test Case: Interface Test 12	17
Table 14 – Test Case: Training Test 1	17
Table 15 – Test Case: Training Test 2	18
Table 16 – Test Case: Training Test 3	18
Table 17 – Test Case: Training Test 4	19
Table 18 – Test Case: Training Test 5	19
Table 19 – Test Case: Training Test 6	20
Table 20 – Test Case: Training Test 7	20
Table 21 – Test Case: Training Test 8	21
Table 22 – Test Case: Multiplayer Test 1	21
Table 23 – Test Case: Multiplayer Test 2	22
Table 24 – Test Case: Multiplayer Test 3	22
Table 25 – Test Case: Multiplayer Test 4	23
Table 26 – Test Case: Multiplayer Test 5	23
Table 27 – Test Case: Multiplayer Test 6	24
Table 28 – Test Case: Multiplayer Test 7	24
Table 29 – Test Case: Multiplayer Test 8	25
Table 30 – Test Case: Multiplayer Test 9	26
Table 31 – Test Case: Multiplayer Test 10	26
Table 32 – Test Case: Multiplayer Test 11	27
Table 33 – Test Case: Multiplayer Test 12	27
Table 34 – Test Case: Multiplayer Test 13	28
Table 35 – Test Case: Multiplayer Test 14	28
Table 36 – Interface Test Results	29
Table 37 – Training Mode Test Results	30
Table 38 – Multiplayer Test Results	30

Revision History

Revision	Release Date
Revenge STD Version-1	May 3, 2015

1 INTRODUCTION

This section identifies the document, and the types of tests and test conditions for the system testing.

1.1 Document Identifier

This document is the 1st version of the System Test Document of the project Revenge, which is developed by the team Hedef4Yildiz. Writers of this document are the team members, who are four senior students in the Computer Engineering Department of METU. The document is developed based on IEEE Standard for Software and System Test Documentation (IEEE 829-2008).

1.2 Scope

Revenge is a multiplayer online fight game which is played with Kinect motion sensor. The product is designed in Unity3D game engine, and has four main components to be tested: User interface, Kinect control, network and database.

1.2.1 User Interface

The product has 7 different interfaces, which are designed to simplify user's navigation within the game environment. On each interface, there are several UI elements, such as buttons, fields, images and indicators. In addition, lots of classes are written to achieve functionality on each interface. Each element and corresponding scripts will be tested to make sure that they satisfy necessary functionality.

1.2.2 Kinect Control

In two of the user interfaces, game characters are controlled by input from Kinect motion sensor. Zigfu Kinect SDK is employed to receive input from Kinect, and several scripts are written to filter, control and bind received input. Kinect component will be tested for its visual adequacy, and performance on reflecting user's actual movements.

1.2.3 Network

The product supports two players' playing the game over Internet. Network component is employed at the time two users connect to a game and play against each other. Network component is tested for its speed and performance while transferring data between two players.

1.2.4 Database

Users are expected to register to use the product. Data related to registered users and their characters in the game are stored in game database. Database component is tested for its consistency, and synchronization with the game environment.

1.3 References

- IEEE Standard for Software and System Test Documentation (IEEE 829-2008).
- Revenge Software Requirements Specification Document
- Revenge Software Design Description Document
- Amazon Web Services
- Unity Technologies

1.4 Level in the Overall Sequence

Revenge is not part of larger system, it is a stand-alone application. Therefore, tests will be applied as one level system tests.

1.5 Test Classes and Overall Test Conditions

The product will be tested as the whole system. At the system level test, the following test classes are available.

- Positive testing of input values that should be processed successfully.
- Negative testing of input that should raise a warning or an error.
- Normal values based on user's interaction with the application.

The system level test will focus on satisfying system requirements, based on the four system components, which are described in Section 1.2.

2 DETAILS FOR SYSTEM TEST PLAN

This section describes the specific items to be tested, and provides a Test Traceability Matrix that relates the items to be tested with the requirements. In addition, general test approach is described with the pass/fail criteria, and test deliverables.

2.1 Test Items and Their Identifiers

Test items are the four main components of the system, namely user interface, Kinect component, network and database, as described in section 1.2. Each component will be tested for its functionality in several use cases. Features to be tested are listed in section 2.3. In addition, detailed use case descriptions are available in Software Design Description of the Revenge project.

Application installation will be performed through Revenge website¹. The downloaded file is a single executable, supported by Unity Game Engine. A Kinect sensor will necessary to properly use the application. Actually, since the application is a multiplayer game, at least two sensors, each connected to a different computer will be required for comprehensive system testing. Network connection is achieved through an external web server, supplied by Amazon Web Services². System database is stored in Amazon servers, as well. MySQL is the database management system, which is used for database connection in the game.

Software Requirements Specification and Software Design Description documents of the Revenge project can be downloaded from the Revenge website.

Project website and game scenario are the parts which will be excluded from system testing.

2.2 Test Traceability Matrix

Use Cases \ Test Cases	Login	Main Menu	Game Settings	User Profile	Character Selection	Training Mode	Match Mode	Exit
Interface Test 1,2	X	X						
Interface Test 3,4,5,6		X	X					
Interface Test 7		X		X				
Interface Test 8,9,10		X			X			
Interface Test 11,12		X						X
Training Test 1		X				X		
Training Test 2,3,4,5						X		
Training Test 6,7,8		X				X		
Multiplayer Test 1,2		X					X	
Multiplayer Test 3,4,5,6,7,8,9							X	
Multiplayer Test 10,11,12,13,14		X					X	

Table 1 – Test Traceability Matrix

¹ senior.ceng.metu.edu.tr/2015/hedef4yildiz

² aws.amazon.com

2.3 Features to Be Tested

Features which will be tested as a part of system level test of Revenge are listed below. These features are functional requirements provided by the product.

- Login
- Display main menu
- Change game settings
 - Music volume
 - Full screen display
- Display user profile
- Character selection
 - Display character properties
 - Choose game character
- Training mode
 - Display
 - Game play
 - Level completion
- Match mode
 - Two characters display
 - Game play
 - End of match
- Exit

2.4 Features not to Be Tested

Features which will not be tested in the scope of this document are listed below. The reason for not testing the specified features is that, the structures to provide these features in the project are not complete, yet.

- Product website
- User registration
- Product installation
- Game scenario
 - User and character experience
 - Progress and levels

2.5 Approach

The approach which is followed at the system level test can be classified as a combination of black box testing and analysis testing. Inputs to the system are user interactions with fields and buttons, and player's body coordinates coming from Kinect sensor. System output will be visual components of the game, which will be displayed on user interface elements. In addition, the system accepts input from network and database, and sends output to them. It is needed to test whether user's interactions with the system result in appropriate data communication with server, and necessary updates on database tables. In that sense, the test approach will require analysis in addition to visual evaluation.

2.6 Item Pass/Fail Criteria

The test items are considered to pass the test if they produce expected outputs, which are specified in description of system use cases. Test items are considered to fail if the output is different from the expected output, or causes an anomaly in the system afterwards, or the output is an unintended error.

2.7 Test Deliverables

Deliverable of the system test is this document itself. The following documents are included in a combined form under sections of this document.

- System Test Plan
- System Test Design
- System Test Cases
- System Test Procedure
- System Test Report

3 TEST MANAGEMENT

This section describes the test activities and progression for the system level test. In addition, the infrastructure and test environments are described in this section, as well.

3.1 Planned Activities and Task, Test Progression

In order to start testing, each developer needs to have the product ready to run on his/her computer. The product is a game with 7 main interfaces. Each interface has its own structure, and expected functionality. In addition, interfaces are combined with each other, so that screen transitions take place only if specific conditions are satisfied.

Each test case is designed to test functionality of some of the interfaces. Test cases will be applied, and test results will be evaluated and recorded. Some test cases are designed as continuation of previous test cases, so for some cases, previous steps will be assumed completed.

At the development stage, all interfaces are tested many times inside the simulator of the Unity game engine for achieving the best visual quality. Scenes are assumed to be completed when they offer error-free game play. The same procedure is planned to be followed while preparing this document. In other words, each scene will be tested for its functionality, and the project will be assumed completed if all scenes passes all test cases.

3.2 Environment and Infrastructure

- For Interface Test Cases:
 - One personal computer which has enough capacity to run a standalone Unity game
- For Training Mode Test Cases:
 - One personal computer with enough capacity to run a standalone Unity game
 - Kinect motion sensor connected to the computer
 - A large and well-illuminated testing room, so that Kinect can sense user motions effectively
- For Multiplayer Mode Test Cases:
 - Two personal computers with enough capacity to run a standalone Unity game
 - Two Kinect sensors, each connected to a different computer
 - Large movement area for users, so that Kinect can sense motions
- Network connection
- Running game and database server

4 TEST CASE DETAILS

This section gives detailed description for all test cases stated in Section 2.2. Each test case is explained together with its identifier, objectives, inputs, expected outputs and dependencies.

4.1 Interface Test Cases

4.1.1 Interface Test 1

Test Case Identifier	Interface Test 1
Objectives	This test case aims to test login panel's functionality when a valid input is supplied.
Inputs	<ul style="list-style-type: none"> • User runs the application. • Login screen is displayed. • User enters her/his username and password. • User clicks "LOGIN" button.
Outcomes	<ul style="list-style-type: none"> • Login is successful. • User's login is recorded in the database. • User is directed to the main menu. • Main menu audio starts playing. • User's username is correctly displayed at upper-right corner of the main menu. • User's current character figure stands at the right side of the screen.
Environmental Needs	Internet connection to connect to the database.
Special Procedural Requirements	User should enter correct username – password pair.
Intercase Dependencies	None.

Table 2 – Test Case: Interface Test 1

4.1.2 Interface Test 2

Test Case Identifier	Interface Test 2
Objectives	This test case aims to test login panel's functionality when an invalid input is supplied.
Inputs	<ul style="list-style-type: none"> • User runs the application. • Login screen is displayed. • User enters a username and/or password. • User clicks "LOGIN" button.
Outcomes	<ul style="list-style-type: none"> • Login request is denied. • An error message is displayed under the login panel: "Invalid user name or password! Try again..."
Environmental Needs	Internet connection to connect to the database.
Special Procedural Requirements	<ul style="list-style-type: none"> • User enters incorrect username – password pair. or • S/he leaves empty one or both of the username – password fields.
Intercase Dependencies	None.

Table 3 – Test Case: Interface Test 2

4.1.3 Interface Test 3

Test Case Identifier	Interface Test 3
Objectives	This test case aims to test “Settings” menu’s functionality when user changes music volume.
Inputs	<ul style="list-style-type: none"> On the main menu, user clicks “SETTINGS” button. Settings screen is displayed with two sliders and one toggle switch. User changes the value of “Music Volume” slider. User clicks “DONE” button.
Outcomes	<ul style="list-style-type: none"> Game settings are updated, new value for music volume is applied. User is directed to main menu.
Environmental Needs	None.
Special Procedural Requirements	None.
Intercase Dependencies	Interface Test 1 In order to display main menu, user should have logged in.

Table 4 – Test Case: Interface Test 3

4.1.4 Interface Test 4

Test Case Identifier	Interface Test 4
Objectives	This test case aims to test “Settings” menu’s functionality when user cancels operation after changing volume settings.
Inputs	<ul style="list-style-type: none"> On the main menu, user clicks “SETTINGS” button. Settings screen is displayed with two sliders and one toggle switch. User changes the value of “Music Volume” slider. User clicks “CANCEL” button.
Outcomes	<ul style="list-style-type: none"> Changes on game settings are ignored, current values are preserved. User is directed to main menu.
Environmental Needs	None.
Special Procedural Requirements	None.
Intercase Dependencies	Interface Test 1 In order to display main menu, user should have logged in.

Table 5 – Test Case: Interface Test 4

4.1.5 Interface Test 5

Test Case Identifier	Interface Test 5
Objectives	This test case aims to test “Settings” menu’s functionality when user changes full screen toggle switch.
Inputs	<ul style="list-style-type: none"> On the main menu, user clicks “SETTINGS” button. Settings screen is displayed with two sliders and one toggle switch. User chooses “Off” option for full screen toggle switch. User clicks “DONE” button.
Outcomes	<ul style="list-style-type: none"> Game settings are updated. Full screen display is disabled, and game is displayed on a window. User is directed to main menu.
Environmental Needs	None.
Special Procedural Requirements	Before the test is applied, full screen should be enabled, which is the default behavior of the game.
Intercase Dependencies	Interface Test 1 In order to display main menu, user should have logged in.

Table 6 – Test Case: Interface Test 5

4.1.6 Interface Test 6

Test Case Identifier	Interface Test 6
Objectives	This test case aims to test “Settings” menu’s functionality when user cancel changes after changing full screen toggle switch.
Inputs	<ul style="list-style-type: none"> On the main menu, user clicks “SETTINGS” button. Settings screen is displayed with two sliders and one toggle switch. User chooses “On” option for full screen toggle switch. User clicks “CANCEL” button.
Outcomes	<ul style="list-style-type: none"> Changes on settings are ignored. Current settings are preserved, full screen display remains inactive. User is directed to main menu.
Environmental Needs	None.
Special Procedural Requirements	Before the test is applied, full screen should be disabled.
Intercase Dependencies	Interface Test 1, Interface Test 5 In order to apply this test, user should have logged in and disabled full screen display

Table 7 – Test Case: Interface Test 6

4.1.7 Interface Test 7

Test Case Identifier	Interface Test 7
Objectives	This test case aims to test if user data is correctly displayed in “Profile” menu.
Inputs	<ul style="list-style-type: none"> On the main menu, user clicks “PROFILE” button. User profile (user name, password and e-mail) is displayed. User clicks “DONE” button.
Outcomes	<ul style="list-style-type: none"> Correct user data is displayed on the “Profile” screen. After clicking “DONE” button, user is directed to main menu.
Environmental Needs	None.
Special Procedural Requirements	None.
Intercase Dependencies	Interface Test 1 User should have logged in to display main menu and user profile.

Table 8 – Test Case: Interface Test 7

4.1.8 Interface Test 8

Test Case Identifier	Interface Test 8
Objectives	This test case tests whether “Character Selection” menu displays game characters’ properties correctly.
Inputs	<ul style="list-style-type: none"> On the main menu, user clicks “CHANGE” button. Character selection menu is displayed. User selects among available character options.
Outcomes	<ul style="list-style-type: none"> On the character selection menu, user’s current character is selected by default. Characters user owns are active for selection, and the others are inactive. As user changes her/his selection, selected character’s data (Character name, experience and durability) is displayed on the screen.
Environmental Needs	None.
Special Procedural Requirements	User should have at least two available characters, which is the default behavior of the game.
Intercase Dependencies	Interface Test 1 User should have logged in to display main menu and characters screen.

Table 9 – Test Case: Interface Test 8

4.1.9 Interface Test 9

Test Case Identifier	Interface Test 9
Objectives	This test case tests whether selected character is updated in all screens when it is changed in “Character Selection” menu.
Inputs	<ul style="list-style-type: none"> On the main menu, user clicks “CHANGE” button. Character selection menu is displayed. User selects a different game character from available options. User clicks “DONE” button.
Outcomes	<ul style="list-style-type: none"> User is directed to main menu. User’s selected character figure is displayed on main menu. User’s character selection is recorded in the database.
Environmental Needs	Internet connection to connect to the database.
Special Procedural Requirements	User should have at least two available characters, which is the default behavior of the game.
Intercase Dependencies	Interface Test 1 User should have logged in to display main menu and characters screen.

Table 10 – Test Case: Interface Test 9

4.1.10 Interface Test 10

Test Case Identifier	Interface Test 10
Objectives	This test case tests whether selected character is preserved in all screens when selection is cancelled in “Character Selection” menu.
Inputs	<ul style="list-style-type: none"> On the main menu, user clicks “CHANGE” button. Character selection menu is displayed. User selects a different game character from available options. User clicks “CANCEL” button.
Outcomes	<ul style="list-style-type: none"> User is directed to main menu. User’s character selection is ignored. Current character is preserved and character figure is displayed in main menu.
Environmental Needs	None.
Special Procedural Requirements	User should have at least two available characters, which is the default behavior of the game.
Intercase Dependencies	Interface Test 1 User should have logged in to display main menu and characters screen.

Table 11 – Test Case: Interface Test 10

4.1.11 Interface Test 11

Test Case Identifier	Interface Test 11
Objectives	This test case tests “EXIT” button’s functionality.
Inputs	<ul style="list-style-type: none"> On the main menu, user clicks “EXIT” button.
Outcomes	<ul style="list-style-type: none"> Application is closed.
Environmental Needs	None.
Special Procedural Requirements	None.
Intercase Dependencies	Interface Test 1 User should have logged in to display main menu.

Table 12 – Test Case: Interface Test 11

4.1.12 Interface Test 12

Test Case Identifier	Interface Test 12
Objectives	This test case tests “ESC” button’s functionality on main menu.
Inputs	<ul style="list-style-type: none"> On the main menu, user presses “ESC” key on keyboard.
Outcomes	<ul style="list-style-type: none"> Application is closed.
Environmental Needs	None.
Special Procedural Requirements	None.
Intercase Dependencies	Interface Test 1 User should have logged in to display main menu.

Table 13 – Test Case: Interface Test 12

4.2 Training Mode Test Cases

4.2.1 Training Test 1

Test Case Identifier	Training Test 1
Objectives	This test case tests whether user interface elements are displayed correctly in the Training Mode scene.
Inputs	<ul style="list-style-type: none"> On the main menu, user clicks “TRAINING” button.
Outcomes	<ul style="list-style-type: none"> Revenge splash screen is displayed. Training Mode scene is opened. Training Mode scene audio starts playing. User’s current character figure is displayed at the center of the screen, near the punching bag figure. User’s and current character’s information is displayed on UI elements on the screen.
Environmental Needs	None.
Special Procedural Requirements	None.
Intercase Dependencies	Interface Test 1 User should have logged in to display main menu and training mode.

Table 14 – Test Case: Training Test 1

4.2.2 Training Test 2

Test Case Identifier	Training Test 2
Objectives	This test case tests whether Kinect component is properly integrated into the Training Mode scene.
Inputs	<ul style="list-style-type: none"> User opens training mode scene. User gets position in front of the Kinect sensor so that the sensor radar detects her/him. User changes her/his position in front of the Kinect sensor.
Outcomes	<ul style="list-style-type: none"> Game character on the scene repeats user's movements.
Environmental Needs	<ul style="list-style-type: none"> A Kinect sensor should be connected to computer. The environment should be properly illuminated.
Special Procedural Requirements	None.
Intercase Dependencies	Training Test 1 Training mode scene should be opened, and working properly.

Table 15 – Test Case: Training Test 2

4.2.3 Training Test 3

Test Case Identifier	Training Test 3
Objectives	This test case tests whether first person display works correctly in Training Mode scene.
Inputs	<ul style="list-style-type: none"> User opens training mode scene, and activates Kinect control. User presses 'F' key on keyboard.
Outcomes	<ul style="list-style-type: none"> First person camera is activated. User starts displaying the scene from her/his eyes' position.
Environmental Needs	<ul style="list-style-type: none"> A Kinect sensor should be connected to computer. The environment should be properly illuminated.
Special Procedural Requirements	Before the test is applied, third person display should be active, which is the default behavior of the game.
Intercase Dependencies	Training Test 2 Training mode scene should be opened, and Kinect control should be activated.

Table 16 – Test Case: Training Test 3

4.2.4 Training Test 4

Test Case Identifier	Training Test 4
Objectives	This test case tests whether transition between first person and third person display is working properly in Training Mode scene.
Inputs	<ul style="list-style-type: none"> • Training Mode scene is opened. • User presses 'C' key on keyboard.
Outcomes	<ul style="list-style-type: none"> • First person camera is disabled, third person display is activated. • User starts displaying the scene from the audience's viewpoint.
Environmental Needs	<ul style="list-style-type: none"> • A Kinect sensor should be connected to computer. • The environment should be properly illuminated.
Special Procedural Requirements	Before the test is applied, first person display should be active.
Intercase Dependencies	Training Test 3 Training mode scene should be opened, and first person camera should be activated.

Table 17 – Test Case: Training Test 4

4.2.5 Training Test 5

Test Case Identifier	Training Test 5
Objectives	This test case tests whether user's interaction with punching bag gives expected results in the Training Mode.
Inputs	<ul style="list-style-type: none"> • Training Mode scene is opened, and Kinect control is activated. • User changes position in front of Kinect sensor. • User moves to make game character hit the punching bag.
Outcomes	<ul style="list-style-type: none"> • Game character hits the punching bag on the scene. • A punch effect appears at the collision point. • Punching bag responds to character's punch, and oscillates for a while in a physically realistic manner. • Collision indicator rises to show strength of the punch. • Value of the experience bar increases in accordance with the strength of the punch.
Environmental Needs	<ul style="list-style-type: none"> • A Kinect sensor should be connected to computer. • The environment should be properly illuminated.
Special Procedural Requirements	None.
Intercase Dependencies	Training Test 2 Training mode scene should be opened, and Kinect control should be enabled.

Table 18 – Test Case: Training Test 5

4.2.6 Training Test 6

Test Case Identifier	Training Test 6
Objectives	This test case tests whether user's exit request is responded correctly in the Training Mode scene.
Inputs	<ul style="list-style-type: none"> • Training Mode scene is opened. • User clicks "MENU" button at the upper right corner of the screen. • A warning panel appears, asking user to confirm the exit request. • User clicks "OK" on the warning panel.
Outcomes	<ul style="list-style-type: none"> • Training Mode is closed, user is directed to Main Menu. • Experience points user gained during the training are lost.
Environmental Needs	None.
Special Procedural Requirements	Level should not have been completed before the test.
Intercase Dependencies	Training Test 1 Training mode scene should be opened, and working properly.

Table 19 – Test Case: Training Test 6

4.2.7 Training Test 7

Test Case Identifier	Training Test 7
Objectives	This test case tests whether the system responds correctly if user cancels her/his exit request.
Inputs	<ul style="list-style-type: none"> • Training Mode scene is opened. • User clicks "MENU" button at the upper right corner of the screen. • A warning panel appears, asking user to confirm the exit request. • User clicks "CANCEL" on the warning panel.
Outcomes	<ul style="list-style-type: none"> • Warning panel disappears, and training goes on. • Experience points user gained during training are preserved.
Environmental Needs	None.
Special Procedural Requirements	Level should not have been completed before the test.
Intercase Dependencies	Training Test 1 Training mode scene should be opened, and working properly.

Table 20 – Test Case: Training Test 7

4.2.8 Training Test 8

Test Case Identifier	Training Test 8
Objectives	This test case tests whether Training Mode exits correctly when end conditions are satisfied.
Inputs	<ul style="list-style-type: none"> • Training Mode scene is opened. • User keeps hitting on the punching bag. • Value of the experience bar increases as user hits, and reaches its highest value.
Outcomes	<ul style="list-style-type: none"> • Training level is completed. • “Training Completed” effect appears on the screen. • User’s level completion is recorded in the database. • Current game character’s experience point is updated.
Environmental Needs	<ul style="list-style-type: none"> • A Kinect sensor should be connected to computer. • The environment should be properly illuminated.
Special Procedural Requirements	User should not leave the game till the end of the training.
Intercase Dependencies	Training Test 5 Training mode scene should be opened, and user’s punch should be tested to give rise to right updates.

Table 21 – Test Case: Training Test 8

4.3 Multiplayer Mode Test Cases

4.3.1 Multiplayer Test 1

Test Case Identifier	Multiplayer Test 1
Objectives	This test case tests whether two players can connect to a match.
Inputs	<ul style="list-style-type: none"> • On the main menu, two users clicks “MATCH NOW!” button.
Outcomes	<ul style="list-style-type: none"> • Server waits until both users click the button. • Server matches the players in one game. • Server sends user ids and match id to both players. • User ids and match id are logged in database.
Environmental Needs	Two computers, each having Internet connection are required to perform the test.
Special Procedural Requirements	None.
Intercase Dependencies	Interface Test 1 Both users should have logged in to display main menu.

Table 22 – Test Case: Multiplayer Test 1

4.3.2 Multiplayer Test 2

Test Case Identifier	Multiplayer Test 2
Objectives	This test case tests whether two players' data is displayed correctly on Match Mode scene.
Inputs	<ul style="list-style-type: none"> On the main menu, two users clicks "MATCH NOW!" button.
Outcomes	<ul style="list-style-type: none"> Users are matched in one game. Each user receives match id, and their opponents' user id from server. Revenge splash screen is displayed. Match Mode scene is opened. Each user's current character figure is displayed on the scene. Information of the two users and their current characters are displayed on UI elements on the scene. Information stated above should be displayed on screens of both players correctly.
Environmental Needs	Two computers, each having Internet connection are required to perform the test.
Special Procedural Requirements	None.
Intercase Dependencies	Multiplayer Test 1 Users should have been matched in a game.

Table 23 – Test Case: Multiplayer Test 2

4.3.3 Multiplayer Test 3

Test Case Identifier	Multiplayer Test 3
Objectives	This test case tests whether two players' actions in front of Kinect are transferred between the two scenes efficiently.
Inputs	<ul style="list-style-type: none"> Users are matched in a game in Match Mode. Each user gets position in front of the Kinect sensor which is connected to her/his computer, so that Kinect detects her/him. Each user changes position in front of her/his Kinect sensor.
Outcomes	<ul style="list-style-type: none"> Each user's movements are repeated by her/his game character on the screen. Each user's body coordinates are transferred to her/his opponent's computer through game server. Both users' movements are displayed simultaneously on both computers' screen.
Environmental Needs	Two computers, each having Internet and Kinect connection are required to perform the test.
Special Procedural Requirements	None.
Intercase Dependencies	Multiplayer Test 2 Match Mode scene should be opened, and working properly for both users.

Table 24 – Test Case: Multiplayer Test 3

4.3.4 Multiplayer Test 4

Test Case Identifier	Multiplayer Test 4
Objectives	This test case tests whether first person display is properly working in Match Mode scene.
Inputs	<ul style="list-style-type: none"> Match mode scene is opened, and Kinect control is activated. User presses 'F' key on the keyboard.
Outcomes	<ul style="list-style-type: none"> First person camera is activated for the user. User starts displaying the scene from her/his eyes' position.
Environmental Needs	Two computers, each having Internet and Kinect connection are required to perform the test.
Special Procedural Requirements	Before the test is applied, third person display should be active, which is the default behavior of the game.
Intercase Dependencies	Multiplayer Test 3 Match Mode scene should be opened, and Kinect control should be activated.

Table 25 – Test Case: Multiplayer Test 4

4.3.5 Multiplayer Test 5

Test Case Identifier	Multiplayer Test 5
Objectives	This test case tests whether transition between first person and third person display is properly working in Match Mode scene.
Inputs	<ul style="list-style-type: none"> Match mode scene is opened, and Kinect control is activated. User presses 'C' key on the keyboard.
Outcomes	<ul style="list-style-type: none"> First person camera is disabled for the user, third person display is activated. User starts displaying the scene from the audience's viewpoint.
Environmental Needs	Two computers, each having Internet and Kinect connection are required to perform the test.
Special Procedural Requirements	Before the test is applied, first person display should be active.
Intercase Dependencies	Multiplayer Test 4 Match Mode scene should be opened, and first person camera should be activated.

Table 26 – Test Case: Multiplayer Test 5

4.3.6 Multiplayer Test 6

Test Case Identifier	Multiplayer Test 6
Objectives	This test case tests effects of a character's punch on her/his own body.
Inputs	<ul style="list-style-type: none"> Match mode scene is opened, and Kinect control is activated. User moves in front of the Kinect sensor, so that the game character's boxing gloves hit its own body parts.
Outcomes	<ul style="list-style-type: none"> Boxing gloves' collision is detected. Since character collides with her/his own body, no action is taken. No values are updated on the UI elements on screen.
Environmental Needs	Two computers, each having Internet and Kinect connection are required to perform the test.
Special Procedural Requirements	No other collision should occur on the scene while performing this test.
Intercase Dependencies	Multiplayer Test 3 Match Mode scene should be opened, and Kinect control should be activated.

Table 27 – Test Case: Multiplayer Test 6

4.3.7 Multiplayer Test 7

Test Case Identifier	Multiplayer Test 7
Objectives	This test case tests effects of a character's punch hitting on opponent's boxing gloves.
Inputs	<ul style="list-style-type: none"> Match mode scene is opened, and Kinect control is activated. User moves in front of the Kinect sensor, so that the game character's boxing gloves hit the opponent's boxing gloves.
Outcomes	<ul style="list-style-type: none"> Boxing gloves' collision is detected. Hitting on boxing gloves does not change any values on UI elements. Punch on gloves effect is displayed on user's screen. No visual effect is displayed on opponent's screen.
Environmental Needs	Two computers, each having Internet and Kinect connection are required to perform the test.
Special Procedural Requirements	No other collision should occur on the scene while performing this test.
Intercase Dependencies	Multiplayer Test 3 Match Mode scene should be opened, and Kinect control should be activated.

Table 28 – Test Case: Multiplayer Test 7

4.3.8 Multiplayer Test 8

Test Case Identifier	Multiplayer Test 8
Objectives	This test case tests effects of a character's punch hitting on opponent's body parts.
Inputs	<ul style="list-style-type: none"> • Match mode scene is opened, and Kinect control is activated. • User moves in front of the Kinect sensor, so that the game character's boxing gloves hit the opponent's body parts, other than boxing gloves.
Outcomes	<ul style="list-style-type: none"> • Boxing gloves' collision is detected. • Punch effect is displayed on user's screen. • User's punch indicator rises to show strength of the punch. • Punch on me effect is displayed on opponent's screen (See Section 4.3.9). • Opponent's health value is reduced in accordance with strength of the collision. • User's updated indicator and health values are displayed on both users' screens simultaneously.
Environmental Needs	Two computers, each having Internet and Kinect connection are required to perform the test.
Special Procedural Requirements	No other collision should occur on the scene while performing this test.
Intercase Dependencies	Multiplayer Test 3 Match Mode scene should be opened, and Kinect control should be activated.

Table 29 – Test Case: Multiplayer Test 8

4.3.9 Multiplayer Test 9

Test Case Identifier	Multiplayer Test 9
Objectives	This test case tests effects of the opponent's punch hitting on user's game character's body parts.
Inputs	<ul style="list-style-type: none"> Match mode scene is opened, and Kinect control is activated. Both users move in front of Kinect sensors, so that the opponent's boxing gloves hit user's game character's body parts other than boxing gloves.
Outcomes	<ul style="list-style-type: none"> User's game character detects opponent's hit on its body. Punch on me effect is displayed on user's screen. User's health is reduced in accordance with strength of the punch. Punch effect is displayed on opponent's screen. (See Section 4.3.8) Opponent's punch indicator rises to show strength of the punch. Updated indicator and health values are displayed on both users' screens simultaneously.
Environmental Needs	Two computers, each having Internet and Kinect connection are required to perform the test.
Special Procedural Requirements	No other collision should occur on the scene while performing this test.
Intercase Dependencies	Multiplayer Test 3 Match Mode scene should be opened, and Kinect control should be activated.

Table 30 – Test Case: Multiplayer Test 9

4.3.10 Multiplayer Test 10

Test Case Identifier	Multiplayer Test 10
Objectives	This test case tests whether user's exit request is responded appropriately on Match Mode scene.
Inputs	<ul style="list-style-type: none"> Match Mode scene is opened. User clicks "MENU" button at the upper right corner of the screen. A warning panel appears, asking user to confirm the exit request. User clicks "OK" on the warning panel.
Outcomes	<ul style="list-style-type: none"> Match mode scene is closed on user's computer. Opponent is informed about user's exit from the game. Opponent wins the game by default, and the user loses. Match result is recorded in the database. Experience of both users and their game characters are updated. Both users are directed to main menu.
Environmental Needs	Two computers, each having Internet connection are required to perform the test.
Special Procedural Requirements	Match should not have been ended before the test.
Intercase Dependencies	Multiplayer Test 2 Match Mode scene should be opened, and working properly.

Table 31 – Test Case: Multiplayer Test 10

4.3.11 Multiplayer Test 11

Test Case Identifier	Multiplayer Test 11
Objectives	This test case tests whether the system responds correctly if user cancels her/his exit request.
Inputs	<ul style="list-style-type: none"> Match Mode scene is opened. User clicks “MENU” button at the upper right corner of the screen. A warning panel appears, asking user to confirm the exit request. User clicks “CANCEL” on the warning panel.
Outcomes	<ul style="list-style-type: none"> Warning panel disappears and match goes on. Opponent can keep playing even if user’s warning panel is displayed on the screen. When warning panel is closed, user’s health is updated according to opponent’s actions.
Environmental Needs	Two computers, each having Internet connection are required to perform the test.
Special Procedural Requirements	Match should not have been ended before the test.
Intercase Dependencies	Multiplayer Test 2 Match Mode scene should be opened, and working properly.

Table 32 – Test Case: Multiplayer Test 11

4.3.12 Multiplayer Test 12

Test Case Identifier	Multiplayer Test 12
Objectives	This test case tests whether the user is correctly properly notified if opponent leaves the game.
Inputs	<ul style="list-style-type: none"> Match Mode scene is opened. Opponent user leaves the game (See Section 4.3.10).
Outcomes	<ul style="list-style-type: none"> User is notified about opponent’s leaving the game. User wins the game by default, and opponent loses. Match result is recorded in the database. Experience of both users and their game characters are updated. Both users are directed to main menu
Environmental Needs	Two computers, each having Internet connection are required to perform the test.
Special Procedural Requirements	Match should not have been ended before the test.
Intercase Dependencies	Multiplayer Test 2 Match Mode scene should be opened, and working properly.

Table 33 – Test Case: Multiplayer Test 12

4.3.13 Multiplayer Test 13

Test Case Identifier	Multiplayer Test 13
Objectives	This test case tests whether the system is properly updated when user wins a match.
Inputs	<ul style="list-style-type: none"> Match Mode scene is opened, and Kinect control is activated. Both users keep playing the game. Opponent's health value becomes zero.
Outcomes	<ul style="list-style-type: none"> User detects that opponent's health value becomes zero. "You win!" effect is displayed on user's screen. Server is notified for end of match. Match result is recorded in the database. Experience of both users and their game characters is updated. Both users are directed to Main Menu.
Environmental Needs	Two computers, each having Internet and Kinect connection are required to perform the test.
Special Procedural Requirements	Users should not leave the game until the end of the match.
Intercase Dependencies	Multiplayer Test 6,7,8,9 Match Mode scene should be opened, and punch effects should be tested.

Table 34 – Test Case: Multiplayer Test 13

4.3.14 Multiplayer Test 14

Test Case Identifier	Multiplayer Test 14
Objectives	This test case tests whether the system is properly updated when user loses a match.
Inputs	<ul style="list-style-type: none"> Match Mode scene is opened, and Kinect control is activated. Both users keep playing the game. User's health value becomes zero.
Outcomes	<ul style="list-style-type: none"> User detects that her/his health value becomes zero. "You lose!" effect is displayed on user's screen. Server is notified for end of match. Match result is recorded in the database. Experience of both users and their game characters is updated. Both users are directed to Main Menu.
Environmental Needs	Two computers, each having Internet and Kinect connection are required to perform the test.
Special Procedural Requirements	Users should not leave the game until the end of the match.
Intercase Dependencies	Multiplayer Test 6,7,8,9 Match Mode scene should be opened, and punch effects should be tested.

Table 35 – Test Case: Multiplayer Test 14

5 SYSTEM TEST REPORT DETAILS

This section provides an overview of the test results, all of the detailed test results, rationale for all decisions, and the final conclusions and recommendations.

5.1 Overview of Test Results

All test cases whose details are explained in Section 4 are applied to the product. All features specified in Section 2.3 are tested. Environmental needs, procedural requirements and intercase dependencies are taken into account for each test case. Consequently, some test cases are performed by one user, while others are performed by two users simultaneously. Similarly, some test cases need to be performed using Kinect, while others can be performed without Kinect. Some test cases need to be applied sequentially, while some are individual tests.

5.2 Detailed Test Results

Test results for each test case are specified in this section.

5.2.1 Interface Test Cases

Test Case Identifier	Test Result
Interface Test 1	PASSED
Interface Test 2	PASSED
Interface Test 3	PASSED
Interface Test 4	PASSED
Interface Test 5	PASSED
Interface Test 6	PASSED
Interface Test 7	PASSED
Interface Test 8	PASSED
Interface Test 9	PASSED
Interface Test 10	PASSED
Interface Test 11	PASSED
Interface Test 12	PASSED

Table 36 – Interface Test Results

5.2.2 Training Mode Test Cases

Test Case Identifier	Test Result
Training Test 1	PASSED
Training Test 2	PASSED
Training Test 3	PASSED
Training Test 4	PASSED
Training Test 5	PASSED
Training Test 6	PASSED
Training Test 7	PASSED
Training Test 8	FAILED

Table 37 – Training Mode Test Results

5.2.3 Multiplayer Mode Test Cases

Test Case Identifier	Test Result
Multiplayer Test 1	PASSED
Multiplayer Test 2	PASSED
Multiplayer Test 3	PASSED
Multiplayer Test 4	PASSED
Multiplayer Test 5	PASSED
Multiplayer Test 6	PASSED
Multiplayer Test 7	PASSED
Multiplayer Test 8	PASSED
Multiplayer Test 9	PASSED
Multiplayer Test 10	FAILED
Multiplayer Test 11	PASSED
Multiplayer Test 12	FAILED
Multiplayer Test 13	FAILED
Multiplayer Test 14	FAILED

Table 38 – Multiplayer Test Results

5.3 Rationale for Decisions

Since the project is not completed to its final extend, 5 test cases failed to result in expected outcomes. Those failures were expected because some more development is required for the product to offer the features tested in those cases.

5.4 Conclusions and Recommendations

Revenge system passes 85% of test cases. This figure reveals the fact that the project is 85% completed. Since the system is developed well-structured from the beginning, it will be easy to introduce missing feature. However, in order to release the product prototype in an acceptable time, product development should be completed as soon as possible to obtain 100% success on system test.