*Software Testing*

# *Test Planning Project*

*Submitted*

*In partial fulfillment*

*For the award of the Degree of*

*Bachelor of Science*

*in Computing in Software Development (year 3)*

**

*Submitted By:*

*Muhammad Noman Junaid (G00351754)*

*GitHub Repository Link =* [*https://github.com/Nomijee/Software-Test-Planning-Project*](https://github.com/Nomijee/Software-Test-Planning-Project)

*Department of Computer Science*

*Table of Contents*

[*Test Planning Project* 1](#_Toc40392753)

[*Introduction* 4](#_Toc40392754)

[*OBJECTIVES AND TASKS* 4](#_Toc40392755)

[*SCOPE* 4](#_Toc40392756)

[*Testing Strategy* 4](#_Toc40392757)

[*Unit Testing* 5](#_Toc40392758)

[*Test 1:* 5](#_Toc40392759)

[*TEST 2* 5](#_Toc40392760)

[*TEST 3* 5](#_Toc40392761)

[*Test 4* 5](#_Toc40392762)

[*TEST 5* 6](#_Toc40392763)

[*TEST 6* 6](#_Toc40392764)

[*TEST 7* 6](#_Toc40392765)

[*TEST 8* 6](#_Toc40392766)

[*TEST 9* 6](#_Toc40392767)

[*TEST 10* 7](#_Toc40392768)

[*System and Integration Testing* 7](#_Toc40392769)

[*Performance and Stress Testing* 8](#_Toc40392770)

[*User Acceptance Testing* 8](#_Toc40392771)

[*Beta Testing* 8](#_Toc40392772)

[*Automated Regression Testing* 8](#_Toc40392773)

[*Test Schedule* 9](#_Toc40392774)

[*Resources/Roles & Responsibilities* 10](#_Toc40392775)

[*Risks/Assumptions* 10](#_Toc40392776)

[*Tools* 10](#_Toc40392777)

# *Introduction*

*This game will be a 2D side-scrolling platformer, inspired by the likes of ‘Salt and Sanctuary’, ‘Shovel Knight’, and ‘Fancy Pants’, with elements from ‘Skyrim’ (mainly in the way the player character and enemy characters attack). The artwork will be inspired mainly by Shovel Knight, which uses mainly pixel art to create its characters and world. The gameplay will be inspired by ‘Salt and Sanctuary’ and ‘Dark Souls’ and ‘Skyrim’, which will see the player navigate progressively difficult levels with a wizard type character that uses magic a lá ‘Skyrim’. Each level will have several enemies that the player must defeat to progress. Each level will also have a boss that the player must defeat to progress to the next level. Each level will contain pickups for the player, such as health pickups to replenish the player’s health.*

# *OBJECTIVES AND TASKS*

*The main aim of game testing is to find bugs in the game software. When a game is developed, the development lead and the QA lead develop test cases. The game testers then execute test cases to find bugs. They report these bugs to the developers, who then fix the bugs.*

# *SCOPE*

*Scope of game testing is quite narrow, but they are extensive individually. The game testing involves testing whether the game has the desired gameplay and functions (health pickups, control mechanism, levels, load and save game etc.), whether it works properly on all the devices it is intended to without crashing.*

# *Testing Strategy*

*In a simplistic view, testing is to identify bugs found in the software, so the problem can be removed. There are different forms of tests and testing that can be categorized as “Black-Box” testing and “Clear-Box” testing (“Clear Box” testing is also known as “White-Box” testing in the software industry).*

## *Unit Testing*

*Test 1:*

*The "Play Game" button will be clicked by the tester to check the Game has loaded.*

*Expected Results: Game’s first stage will be shown on screen, text will appear on-screen informing the player, final (int) number of lives and the enemies will start to attack the player as set up.*

## *TEST 2*

*Test Description: "settings" button will be clicked by the tester, option shown ensuring that the settings screen appears and contains required buttons which allow player to go settings.*

*Expected Results: Setting Menu, with option button as required in story line.*

## *TEST 3*

*Test Description: The "Load game" button will be clicked by the tester and ensure that the level that is presented will be the appropriate level as per the saved game data; For this test, the user will first play the game, then have the game save her progress to the data file, and then engage in this test.*

*Expected Results: This test will show the appropriate level on the screen with the gameplay functions working as expected.*

## *Test 4*

*Test Description: The tester will click the “Exit Game” button on the main menu and the game will exit without any errors*

*Expected Results: The game will be exited with no errors.*

## *TEST 5*

*Test Description: The " Delete Game " button will be clicked by the tester and they will assess that if work as intended and deletes game data.*

*Expected Results: This test will Delete saved game data.*

## *TEST 6*

*Test Description: The tester will click the load button and assess that the selected game loads as intended.*

*Expected Results: The game data loaded when the tester tested the load button.*

## *TEST 7*

*Test Description: The "settings" button will be clicked by the tester and they will assess that it takes to setting where user can change audio and music.*

*Expected Results: This test takes to the settings screen, and tester is able to control volume of the music and sound effects.*

## *TEST 8*

*Test Description: The tester will play the game and assess that the player and enemy graphics, sound effects and music work as expected.*

*Expected Results: The enemies and player graphics, sound effects and music work as expected*

## *TEST 9*

*Test Description: The tester will play the game and assess that the player moves according to the controls.*

*Expected Results: The player moves forward with Right arrow key/D, backward with Left arrow key/A, jumps with Up arrow key/W, crouch with c and attack with Left mouse click/R and the game is paused or resume with spacebar.*

## *TEST 10*

*Test Description: The tester will play the game and checks that the player lives functions as intended upon collision with enemy projectile. and when collides with health pickup.*

*Expected Results: The players lives increase by a red diamond upon health pickup and decreases by a red diamond upon collision with enemy projectile.*

## *System and Integration Testing*

*System Integration Testing is defined as a type of software testing carried out in an integrated hardware and software environment to verify the behavior of the complete system. It is testing conducted on a complete, integrated system to evaluate the system's compliance with its specified requirement.*

*Methodology:*

*System integration testing is a testing process that exercises a software system's coexistence with others.*

*Test Description:*

*The tester will play the game tests the gameplay. Then tester will test the pause menu and then will test the main menu of the game. Then tester will play the game starting from main menu and complete all three levels to ensure the game as a whole works appropriately.*

## *Performance and Stress Testing*

*Performance testing is carried out to check the system's performance under varying loads. Stress testing is carried out to check the behavior of the system under the sudden increased load. It contains load and stress testing as components.*

*Test Description:*

*AgileLoad is an enterprise-class performance testing solution for optimizing the performance of the game and will provide a deep analysis of test results*

## *User Acceptance Testing*

***User Acceptance Testing (UAT), also known as beta or end-user testing, is defined as testing the software by the user or client to determine whether it can be accepted or not.***

*Test Description:*

*The users will be called on site where they will test the game by playing it and the software testing team (Mars, Pluto, Saturn) will monitor them. And will provide feedback.*

## *Beta Testing*

*Beta Testing is one of the Acceptance Testing types, which adds value to the product as the end-user (intended real user) validates the product for functionality, usability, reliability, and compatibility.*

*Methodology:*

*The beat version of the game will be released and will be available for the user to play and provide feedback in the form of the surveys.*

## *Automated Regression Testing*

*Automated regression testing is a software testing technique that utilizes computer-based tools and techniques in testing software after it has been changed or updated. It is a test automation process that applies the work flow, plan, scripts and other processes within a regression testing methodology.*

# *Test Schedule*

|  |  |  |  |
| --- | --- | --- | --- |
| *Test Name* | *Test* | *Week* | *Day* |
| *Unit Testing* | *Test 1* | *Week 1* | *1* |
| *Unit Testing* | *Test 2* | *Week 1* | *2* |
| *Unit Testing* | *Test 3* | *Week 1* | *3* |
| *Unit Testing* | *Test 4* | *Week 1* | *4* |
| *Unit Testing* | *Test 5* | *Week 1* | *5* |
| *Unit Testing*  *System and Integration Testing* | *Test 6*  *Test 1* | *Week 1* | *6* |
| *Unit Testing*  *System and Integration Testing* | *Test 7*  *Test 2* | *Week 2* | *1* |
| *Unit Testing*  *System and Integration Testing* | *Test 8*  *Test 3* | *Week 2* | *2* |
| *Unit Testing and Performance and stress testing* | *Test 1,*  *Test 9 of unit testing* | *Week 2* | *3* |
| *User Acceptance Testing* | *All teams and Users* | *Week 2* | *3* |
| *Automated Regression Testing* | *If any bug found* | *Week 2* | *4* |
| *Beta Testing* | *Release beta version* | *Week 2* | *5* |

# *Resources/Roles & Responsibilities*

*Team Mars ……. Unit testing, Performance and Stress testing, User acceptance testing*

*Team Pluto ……..System and integration testing ,User acceptance testing*

*Team Saturn …… User acceptance testing, Beta Testing*

# *Risks/Assumptions*

* *Incorrect scheduling (wrong project assessment, distribution of tasks);*
* *Inaccurate customer requirements (requirements change during the work on the project);*
* *Violation of specifications (overload requirements/no exact requirements);*
* *Low productivity (unexperienced specialists).*

# *Tools*

1. Unity
2. AgileLoad for testing
3. Ronex