|  |
| --- |
|  |
| Test Planning Project  Software Testing |
| |  |  | | --- | --- | | G00332617@gmit.ie | Robert Ryan | |

***Contents***

[1.0 Introduction 2](#_Toc40274449)

[2.0 Objectives and tasks 2](#_Toc40274450)

[2.1 Objectives 2](#_Toc40274451)

[2.2 Tasks 3](#_Toc40274452)

[3.0 Scope 3](#_Toc40274453)

[4.0 Testing Strategy 4](#_Toc40274454)

[4.1 Unit testing 4](#_Toc40274455)

[4.2 System and Integration testing 4](#_Toc40274456)

[4.3 Performance and stress testing 4](#_Toc40274457)

[4.4 User Acceptance testing 4](#_Toc40274458)

[4.5 Automated Regression testing 4](#_Toc40274459)

[5.0 Test schedule 4](#_Toc40274460)

[6.0 Control procedures 4](#_Toc40274461)

[7.0 Features to be tested 4](#_Toc40274462)

[8.0 Features not to be tested 4](#_Toc40274463)

[9.0 Resources/Roles & Responsibilities 4](#_Toc40274464)

[10.0 Schedules 4](#_Toc40274465)

[11.0 Risks/Assumptions 5](#_Toc40274466)

[12.0 Tools 5](#_Toc40274467)

Prepared by: Robert Ryan (G00332617)

Date: 08/05/20

# Introduction

The test plan document is to outline the procedure that will be implemented to demonstrate that the game meets the design specifications/requirements. This document consists of guidelines that will aid the people involved in testing this game.

# Objectives and tasks

## Objectives

The objective of this test plan is to ensure that the game is free from input errors as well as all game logic will preform as intended to do so.

* When game is loaded menu is displayed with play, settings and exit buttons displaying and fully functional.
* Once settings is selected users is able to control the volume of the music and game sound effects.
* In game pause menu has full functionality with “Resume game”, “Settings” and “Restart level” and “ Exit game” e.g. If user chooses “Resume” it will allow the player carry on from the exact position they paused the game at.
* Restarting a level in the pause menu will bring the player back to the beginning of the level with all the enemies being reset to their starting position also
* Controls for both PC and Mobile will work fully. “W/Up Arrow”, “A/Left Arrow”, “S/Down Arrow”, “D/Right Arrow” will control movement for PC while movement on mobile will be controlled by onscreen buttons represented by arrows. “C” will control the crouch ability on PC while on an onscreen button positioned towards the top left of the screen will be used for mobile. “Spacebar” will be used for the pause functionality on PC, with mobile having a button positioned at the top right of the screen to control the pause menu. Left/Right mouse click will allow the user to attack.
* Interaction with enemies will show a deduction in enemies health if attacked by the player or the players health will decrease if the enemy attacks them.
* Health will be displayed for both play and enemy at the top of the screen for both PC and mobile. The health will be represented by diamonds for both player and enemy in separate colours.

## Tasks

* Unit Testing: Unit testing is where individual components of the game will be tested. This is to show that each component of the game performs as designed.
* Integration Testing: Integration testing is where the individual components are combined and tested together as a group. This is to expose the faults between integrated components.
* System Testing: System testing will be used to test the complete and integrated game. The purpose of this is to assess the games compliance with the previously stated requirements.
* Performance Testing: Performance testing is to test the game with focus on responsiveness and stability under a certain load.
* Stress Testing: Stress testing is used to test the game under a heavy load to ensure that the game will not crash.
* User Acceptance Testing (UAT): UAT is the one of the final tests we will carry out, actual game users will test the game to see if it can handle the required tasks in real-world scenarios.
* Automated Regression Testing: Automated regression testing is used to speed up the testing process, as testing everything manually is very time consuming and costly.

# Scope

To begin we will be testing the menu functionality. We need to make sure that every button does as required, Play game button brings the player to the first level. Settings button allows the users to control the audio both sound effects and music in game. Exit game button will close the game for the user when pressed. All these buttons will be tested on PC and mobile.

Next to be tested is how the character reacts to the movement inputs by the user “W, A, S, D as well as the arrow keys”. “C” for crouching will also be included in the testing of movement. Similarly the onscreen buttons for movement on mobile will also be tested.

The pause menu will be next to be tested as it is the last of the controls to be tested. The spacebar button is what triggers the game to pause and the pause menu to be displayed. The buttons accessible through the pause menu will be, “Resume”, “Restart Level”, “Settings” and “Exit”. The resume button show should allow the user to carry on where they left off before pausing the game. The restart button should reset the user to the beginning of the level and reset everything within the level the user is currently playing. The settings button should allow the user the same functionality as the settings button in the start menu, audio control. The exit game button should as the same as the exit button in the start menu by closing the game when it is pressed.

The next thing to be tested is how the users character interacts when faced with an enemy. The users health should deplete when being attacked by an enemy. When the health is empty the user should have a screen displayed to them that they has died/failed, with an option to restart.

How the user’s attacks affect the enemy, using the left/right mouse click will make the user perform an attack. How the enemy interacts with being attacked will be tested. This is important as all the enemies are not the same so a different amount of hits are needed to defeat different enemies.

Health pickups are scattered throughout the levels, they will be used to replenish the users health bar when acquired. The health pickups will need to be test on how they work when the users attempted to pick one up with full health and when they have been damaged.

Once the user has defeated all the enemies in the level, the users should be loaded into the following level. The game will consist of 3 levels, following the completion of the final level the user will be given the option to start again from the first level or exit the game.

# Testing Strategy

## Unit testing

* Start Menu

The first component to be tested is the start menu. The features of this component to be tested is the setting and exit. We will be testing the play game feature when the first level is integrated, this is due to the play feature being dependent on another component. The setting feature will be tested by changing the volume of the music, the default will be max volume and will be changed to confirm that the music volume will change.

* Character Movement
* Enemy Interaction
* Health Pickups
* Level Completion

## System and Integration testing

* Play Feature (Start Menu)
* Pause Menu
* Completion Screen

## Performance and stress testing

* Number of enemies spawned before game begins showing signs of potential failure or slowing down.

## User Acceptance testing

## Automated Regression testing

# Test schedule

# Control procedures

# Features to be tested

# Features not to be tested

# Resources/Roles & Responsibilities

* Jane Doe – Unit testing
* John Smith – Integration testing
* Ian O’Connor – System testing
* Eoghan Murphy – Performance/Stress testing

# Schedules

# Risks/Assumptions

# 12.0 Tools