Introduction

This document is a Testing Techniques and Testing Plan for my game ZDungeons. The purpose of the first part of the document is to go over the testing techniques that I have researched and implemented into my game. The second part of the document goes over my testing plan for the game explaining the plan for the testing of the game and how I have done it.

Inside the Investigation into testing techniques and types part of this document, I go over two techniques which can be used to efficiently test a game and I explain the advantages of using the techniques. I then go on to the types and methods section in which I talk about the importance of performance testing and profiling.

Investigation into Testing Techniques and Types

Techniques:

Functionality Testing

The purpose of functionality testing is to follow a set of specifications and then check if the video game works according to the specifications. Examples of problems followed by functionality testing are generic graphics, interface, stability or mechanical issues. Other examples can be bugs like freezing, crashing and progression blockages. All of these examples can be solved by using this technique.

More examples of problems faced by functionality testing can be audio-video and asset integrity. Audio is a very important part of a game that can make or break a game for the player. As a result, this is a very important thing to check when testing a game.

Using this technique game testers will watch out for any graphical corruptions in the interface. Examples are missing graphics, missing colours, issues with location and issues with animation and clippings. The team will also validate the game's performance during restart switch-off and crash situations. The testing team creates these environments by considering real-life situations that could happen to a player.

Once all bugs and errors have been identified, game testers will then make a list and send them back to the game developers to be fixed. Once the bugs and errors are fixed they are then sent back to the testers for retesting to ensure that the issues are resolved.

Ad-Hoc

I have decided to write up on this technique for game testing as an extra note to this document as I feel that it is likely to be used a lot when looking for simple game bugs. Ad-Hoc is an informal technique in which a game developer tries to detect bugs through a simple game of error guessing and consists of detecting bugs through a random approach. As a result, no special documentation is required to be created.

Types or methods

Performance Testing/Profiling

The first factor that comes into play is Frame rate. A games developer will need to take into consideration the frame budget. This is the difference between frame rate and frame time. Frame rate is the amount of frames that are being rendered within a second whereas frame time is the milliseconds per frame. Due to the difference in milliseconds per frame, a player might be able to notice a shutter effect in the case that the last frame takes a quarter of a second to render. To ensure that the game keeps a steady performance and runs smoothly on the client's machine when being played the game developer will need to consider creating a frame budget per frame when profiling and optimizing the game.

The second factor in performance testing is reducing memory access operations. DRAM access is typically a power-hungry operation across any device but typically worse on mobile devices. The best way to reduce the amount of memory being used by the game is to reduce the frame rate, game resolution and to create simpler meshes to reduce vertex count and attribute precision.

When profiling it is better to use a high to low level approach. You will want to start with deep profiling disabled and use a high-level approach to collect data and take notes on which scenarios cause unwanted managed allocations or too much CPU time in the core game loop areas.

You will want to start profiling early in the game's development cycle and do it often so that you can understand and memorize a performance signature for the game. That way if you notice a performance decrease you will be able to easily spot when things go wrong and you will then be able to do something about it.

The most accurate profiling results will always come from running builds of target devices to see how they react as they will be the devices that the game is releasing to.

The final factor to consider is what parts of your game are CPU-bound and which parts are GPU-bound. This is very important to consider as you need to ensure that your game runs as smoothly as possible without putting too much strain on the CPU and GPU.

Testing Plan: Suites and Cases

I will be testing my game using some functionality testing alongside some Ad-Hoc for some smaller parts of the game. The following are the suites that I have created for the test cases that I feel will make testing more organised and readable for game testers and developers.

Playable Character Controls

Test Case	Environment	Instruction	Expected Behaviour
1	Xbox Series X/S	Press Y button	Switches Weapon
2	Xbox Series X/S	Press X button	Reloads Weapon
3	Xbox Series X/S	Press B button	Interact with object. E.G: Chest
4	Xbox Series X/S	Press LT button	Aims Gun
5	Xbox Series X/S	Press RT button	Shoots Bullets from gun
6	Xbox Series X/S	Press Settings button	Pauses Game and open Pause menu
7	Xbox Series X/S	Left Toggle up, down, left and right	Character moves forwards, backwards, left and right
8	Xbox Series X/S	Right Toggle up, down, left and right	Character looks up down, down, left and right around the room

Game Start User Interface

Test Case	Environment	Instruction	Expected Behaviour
9	Xbox Series X/S	Press the Start Game Button	Game Loads up

10	Xbox Series X/S	Press the Exit Game Button	Safely Closes the game
11	Xbox Series X/S	Press the Settings Button	Opens up the settings menu

Game Over User Interface

Test Case	Environment	Instruction	Expected Behaviour
12	Xbox Series X/S	Press the Replay Button	Loads up a new game
13	Xbox Series X/S	Press the exit button	Exits to the start menu UI

Game Escape User Interface

Test Case	Environment	Instruction	Expected Behaviour
14	Xbox Series X/S	Press Play Again button	Loads up a new game
15	Xbox Series X/S	Press the exit button	Exits to the start menu UI

In-Game HUD User Interface

Test Case	Environment	Instruction	Expected Behaviour
16	Xbox Series X/S	Test health bar animation by getting attacked.	The health bar should play a decreasing animation and change the health number.
17	Xbox Series X/S	Test health bar animation by collecting health points from a chest	The health bar should play an animation and change the health number.
18	Xbox Series X/S	Test Chests opened count	When the player opens a chest the number of chests opened should increase in number.
19	Xbox Series X/S	Test Bosses defeated count	When the player defeats a boss the number of bosses defeated should increase.
20	Xbox Series X/S	Test able to escape animation	Complete all of the required tasks to test if the Able to Escape text displays on the HUD.

AI Character Mechanics

Test Case	Environment	Instruction	Expected Behaviour
21	Xbox Series X/S	Test Zombie attack animation and sounds	When zombies attack they should play an attacking animation and make attacking sounds
22	Xbox Series X/S	Test Zombie attack damage	When zombies attack they should deal damage to the player taking away half a health point per hit
23	Xbox Series X/S	Test Zombie death animation	When zombies die they should play a death animation and disappear from the game.

24	Xbox Series X/S	Test Zombie running animation	When a player gets within the sensory range of the zombies they will start running
25	Xbox Series X/S	Test Boss jumping animation	When a boss jumps it should play an animation in which it jumps up in the air and jumps towards the player
26	Xbox Series X/S	Test Boss Armour health	When a player shoots at the armour they should see the health bar for each piece of armour decreasing
27	Xbox Series X/S	Test Boss Armour death animation	When a player destroys the armour it should disappear from the boss
28	Xbox Series X/S	Test that when all boss armour falls of the main boss health bar appears	When a player shoots all of the armour off the boss they should then see the main health bar appear for the boss.
29	Xbox Series X/S	Test Boss health decrease	When a player shoots the Boss the health bar should decrease
30	Xbox Series X/S	Test Boss death animation and key drop animation	When a player kills the boss the death animation should play and then a key should appear which can be collected.
31	Xbox Series X/S	Test Boss Damage	When the boss hits the player it should deal three damage taking away three health points from the player

Game Environment

Test Case	Environment	Instruction	Expected Behaviour
32	Xbox Series X/S	Test Navigation Mesh for each normal room	The player should be able to easily navigate around each room without any issues.
33	Xbox Series X/S	Test Boss room Navigation Mesh	The player should be able to easily navigate around boss rooms without any issues
34	Xbox Series X/S	Test that objects such as Tables, Chairs, Chests, Lights and paintings cannot be walked through and don't cause an obstruction	The player should be able to walk around objects without any problems. The player should not be able to walk through objects.
35	Xbox Series X/S	Test that the player cannot walk through walls and doesn't get stuck on them	The player should not be able to walk through a wall. Instead, they should be forced to turn around and walk in another direction.

Object Functionality

Test Case	Environment	Instruction	Expected Behaviour
36	Xbox Series X/S	Test that Lights flash in intense scenes	When the player is running away from zombies the lights should start flashing.
37	Xbox Series X/S	Test that Doors open and close	When a player interacts with a door it should open. If the player interacts with an open door it should close.
38	Xbox Series X/S	Test that the escape door can be interacted with	When a player interacts with the escape door it should be able to load up the escape ui.

Playable Character Mechanics

Test Case	Environment	Instruction	Expected Behaviour
39	Xbox Series X/S	Test that when the player shoots at a zombie the gun loses rounds and the zombie loses health	When a player shoots at a zombie the gun should lose rounds and the zombie should lose health
40	Xbox Series X/S	Test that when the player aims their gun you can visually see the gun aiming and that you can look around the room	When a player aims their gun they should be able to see their gun whilst looking around the room
41	Xbox Series X/S	Test Gun Shooting sounds	When the player shoots their guns they should hear shooting sounds
42	Xbox Series X/S	Test Gun Reloading sounds	When the player reloads their gun they should hear reloading sounds
43	Xbox Series X/S	Test player death animation	When a player loses all of their health points the game should play a death animation and load up the game over UI.

Game Visuals and Performance

Test Case	Environment	Instruction	Expected Behaviour
44	Xbox Series X/S	Test that the game stays at a steady frame rate when shooting	When a player shoots the frame rate should stay steady to keep a smooth performance.
45	Xbox Series X/S	Test that the game stays at a steady frame rate when walking around	When a player is walking around the frame rate should stay steady to keep a smooth performance.
46	Xbox Series X/S	Test that when a player looks around and turns around the game stays at a steady frame rate	When a player turns around the game should not jump. It should smoothly turn around in the room. When a player looks

			around it should be smooth with frame jumping.
47	Xbox Series X/S	Test that the visuals are bright enough to be able to see	When a player plays the game they should always be able to see what's in the room. Lighting should be affected in certain rooms and certain moments but the game should always be playable so they should always be able to see.
48	Xbox Series X/S	Test that all game assets are coloured correctly and don't look saturated or greyed.	When a player is playing the game they should be able to see what everything is and understand what it is. One of the key factors in making this possible is ensuring that the assets all have the correct colours.

Bibliography:

Starloop. (2022). 6 Types of Game Testing Techniques [Online]. Available:

https://starloopstudios.com/6-types-of-game-testing-techniques/

Qable. (2023, Dec 26). 7 Different Types of Game Testing Techniques [Online]. Available:

https://www.qable.io/blog/7-different-types-of-game-testing-techniques

Unity. (2023). Performance profiling tips for game developers [Online]. Available:

https://unity.com/how-to/best-practices-for-profiling-game-performance