

Test Plan

Top Down action RPG

Document version 0.1

Overview	2
Point of interest	3
What we hope the testers will find	3
Which part need to be tested	3
How are we going to test	3
Why do we need testing	3
What are we going to do with the results	3
Testers	4
Criteria of a valid tester	4
Minimum amount of testers	4
Questions	5
Asked questions	5
How do people answer these questions	5
Aspects that should be observed	5
Procedure	6
After testing	7
Collisions	7
Arrows flying through walls	7
Spawning and get immediately killed	7

Overview

The current state of the game is nearly finished. Although we don't have the boss yet we've completed our first level with incredible much sound effects. Also all bugs we've noticed so far are already fixed. The game now includes a fully working main menu, overworld and first level. Including: save games, audio, music, movement, controller support, 2 kind of enemies, money, shop, health power up, shooting. Completely random level generating is working perfectly fine as well.

Point of interest

What we hope the testers will find

We'd like to know if there are any bugs that we don't know or just didn't see. Also we'd like to know if the game is fun and if the controls feel natural. We'd also like to know what kind of people would play this game. Hardcore PC gamers or generic ps4 gamers? And do people still like this old school graphic design?

Which part need to be tested

Since the main menu, overworld and first level are finished. These are the scenes that should be tested. To put it in a little list: Settings, save games, killing enemies, power ups, potions, shop, random generate system etc. And of course finishing the level.

How are we going to test

Testing will be done with a prototype. I'll select a few random people from Friesland College. And give them a link to a Google docs form and a download to a windows and mac version of the game. I'm also going to drop the game in different Discord channels to get more people. In the Google forms we'll add a few questions to determine which audience someone belongs to.

Why do we need testing

Testing a game is very important. As the developer I exactly know how to play the game and what to do. We have to find out if the controls really feel natural and if actual players know what to do in the game as well. Testing is also very handy for finding bugs because players who don't know how to play the game will do different kind of things than me. This can cause bugs that we can fix later on.

What are we going to do with the results

The results will be individually checked and processed. This means that we will check if the purpose of the game was clear, if it was fun and if it was entertaining to players. We'll process the forms from the testers to make the game better. To make sure the game is entertaining, fun and feels natural. We are going to try to process every feedback that we get.

Testers

Criteria of a valid tester

At the beginning of our form we have a few questions for our testers to place them in a certain type of target audience. This makes sure we are processing the feedback from our target audience. A valid tester should be a Handhold or PC gamer since the game is coming out for PC but the game looks like a handheld game. The age should at least be 16+ but there is no maximum age. A valid tester should like the pixel art style and it would be nice if the tester used to play 8/16 bit games in the past.

Minimum amount of testers

We need at least 5 testers to get some good feedback. If we have to less testers the change that we find bugs is way to low. We also won't have enough input if the game is fun and clear. This is why we need at least 5 testers.

Questions

Asked questions

First: Target audience

- What is your age?
- Which study do you do?
- How many average hours do you game per week?
- What kind of console do you use most for gaming?
- Which game genres do you play?

Second: Ratings about our game

- What rating do you give the overall look of the game? (1-10)
- Does the music fit within the game? (1-10)
- Did you know what you had to do, was it clear? (1-10)
- Does the game run smooth? (1-10)
- Do the controls feel natural to you? (1-10)
- Is the game challenging enough? (1-10)
- Would you consider sharing this game with your friends? (1-10)

Third: Open questions

- What kind of feeling do you have during the game?
- Did you find any bugs or mistakes?
- If you could change 1 thing about this game, what would it be?
- How much do you think this game is going to cost?
- Do you know more games like this?

How do people answer these questions

We'll send a link to a Google form page where people can fill in these answers.

Aspects that should be observed

The following aspects need to be observed properly:

- How long does it take before the player is walking the correct way?
- How do they react when they enter a level for the first time and see the enemies?
- Do players struggle with anything?
- Do the players quickly stop playing or do they enjoy it and do they want more?

Procedure

1. I have made a paper with a few instructions on how to download and start the game and where to fill in the form.
2. I've printed 10 of these papers.
3. I've uploaded a build to Google Drive and hand out the papers to random people at Friesland College.
4. They had to go to the Google Drive url to download the game.
5. They had to unzip it.
6. They had to start Game.exe
7. And now they could play while I was observing their reactions.
8. I let them all fill in my Google Form document.
9. I've checked all Google Form documents and took notes of the answers that needed a change or fix.
10. I've fixed these issues and let them test it again to check if these issues were solved.

After testing

Collisions

After the first tests we found out that Unity had a bug in the Tilemap colliders. They were not working in the build. This issue came up after the first 2 testers and could be quickly resolved finding a big fix on the Unity forms.

Arrows flying through walls

We used a `isStatic` check to see whenever the arrow is colliding with a wall. This was also not supported in a build but only within the Unity editor. We made our own tag and compare it with that to resolve this issue.

Spawning and get immediately killed

A few players were pretty annoyed that they were killed very quickly by suicide flies. We made a system that the player can't take damage during transitioning of rooms. And we made the flies spawn further away.