Develop and Refine a Computer Game

Files are too big to attach, here are the links:

GitHub repository: [Dungeon-Resurgence]

GitHub releases: [Dungeon-Resurgence-final-relase]

Scenario:

You have a job as a trainee programmer at a software developer who creates computer games.

You have been asked to write a computer game demo for the game company to expand into a complete game aimed at kids 13-17.

The basic requirements:

The game involves navigating a character around an area collecting items and avoiding hazards, must have 2 levels.

Collecting items adds to your score, and colliding with hazards uses up lives, of which each game character only has a limited number.

For example, the player could direct a mouse around a house collecting pieces of cheese, but the mouse needs to avoid cats (hazards) which occupy the house. The game should be timed, with player performance based on the score achieved over a certain time. It should also have levels of difficulty with more difficult levels having more hazards, fewer lives or a shorter time. The game should be exciting and fun to play, as well as visually appealing.

Throughout your assignment you must present evidence that you have shown individual responsibility, effective time-management in your design and development of a computer game, making high-

quality justified recommendations and decisions. For example, you need to show how you have:

- **Planned and managed your time and met targets.**
- **Reviewed and responded to outcomes including the use of feedback from others**
- **Behaved appropriately while completing the assignment – including professionalism, etiquette, supportive of others, timely and appropriate leadership, accountability and individual responsibility**
- **Evaluated outcomes to help inform high-quality justified recommendations and decisions**

Used appropriate methods of communication effectively

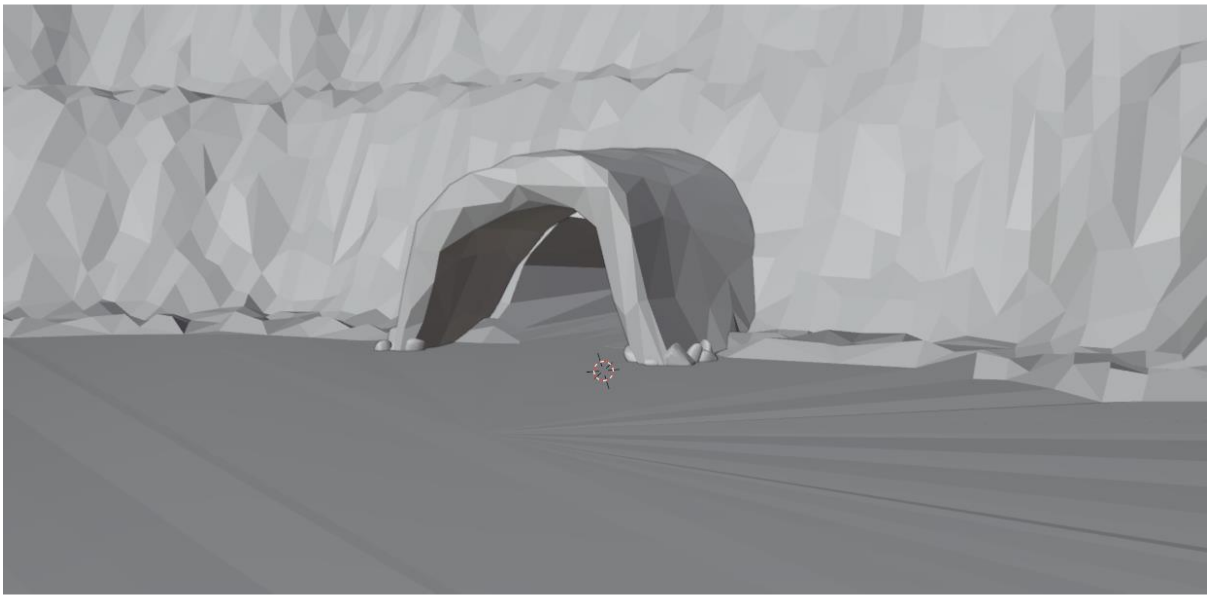
Starting Ideas

When thinking about what kind of game to make I was thinking about expandability and replayability as this game company wants to make this game into a complete game to sell to customers. So I wanted something super replayable and simple yet with a lot of progressions to keep the user engaged.

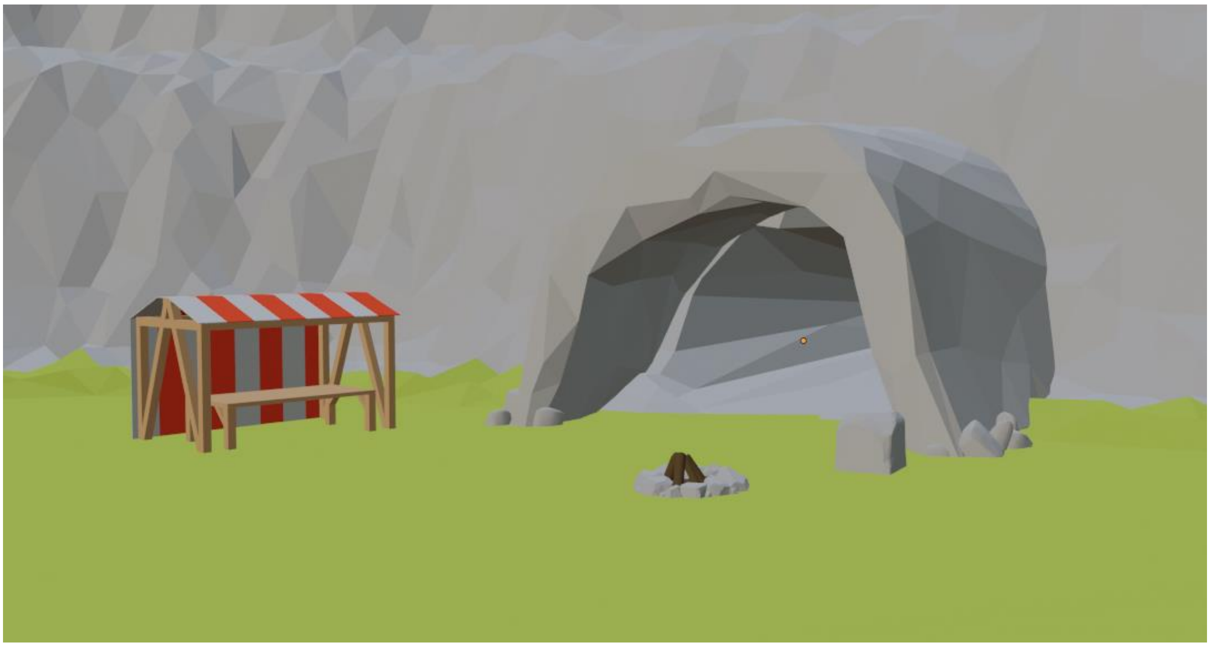
In the end, I settled on a generic dungeon crawler with rogue-lite aspects and named the game "Dungeon Resurgence". The basic premise of the game is that the character has X amount of coins and buys a weapon then goes down into the dungeon and fights monsters. The dungeon is somewhat randomly generated but very similar between dungeon runs. According to the client's requirements, the game has to have a minimum of 2 levels so I decided to 1-up this requirement and make the dungeon endless. When the character dies he will lose one of the items he collected or 10% of his coins. You can buy better weapons and items outside of the dungeon to incentivise getting to the harder levels of the dungeon and getting more coins and even better items. It has a very simple and easy-to-follow idea but has a lot of complex elements with weapon choices for the players who take the game more seriously.

Designs of the game

The Design process of the game happened right at the beginning however continued designing was necessary in order to keep up with the ever-more changing and complex game. In the beginning, I had no idea how I wanted the outside to look, so I took to a blender and started to model some terrain for the main menu.
After some time I came up with this terrain and used it as a baseline for the rest of the game, following the same low-Polly feel and design.

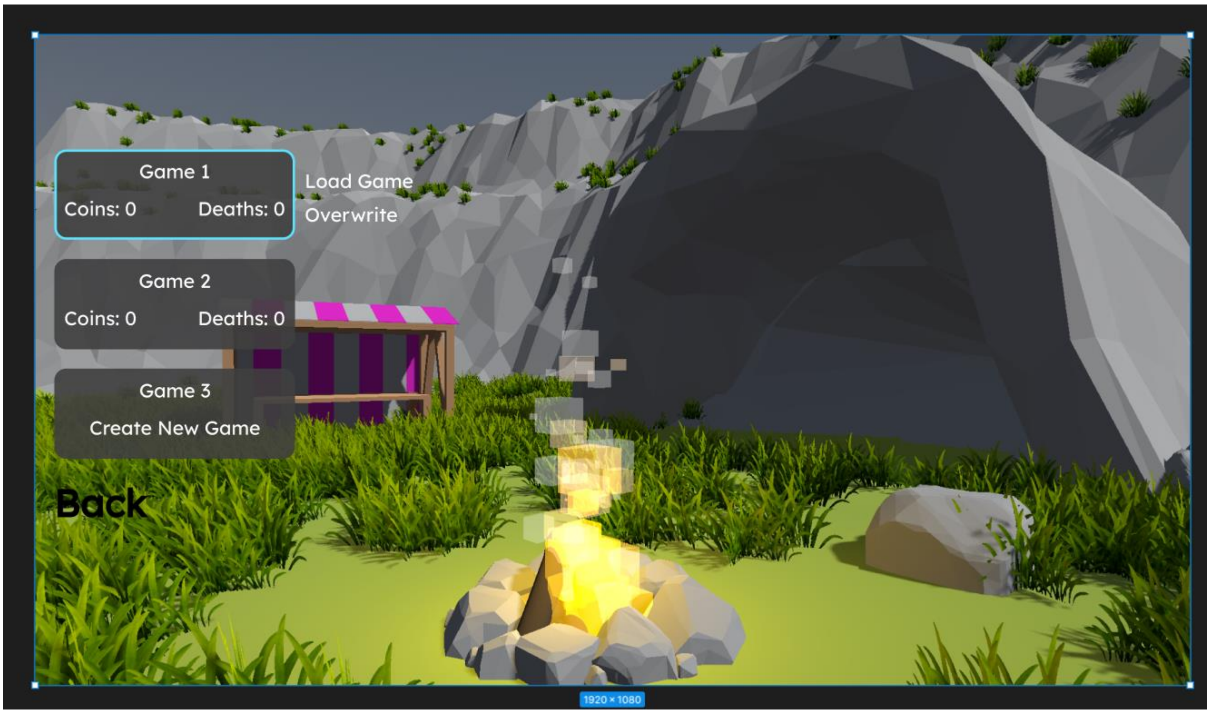


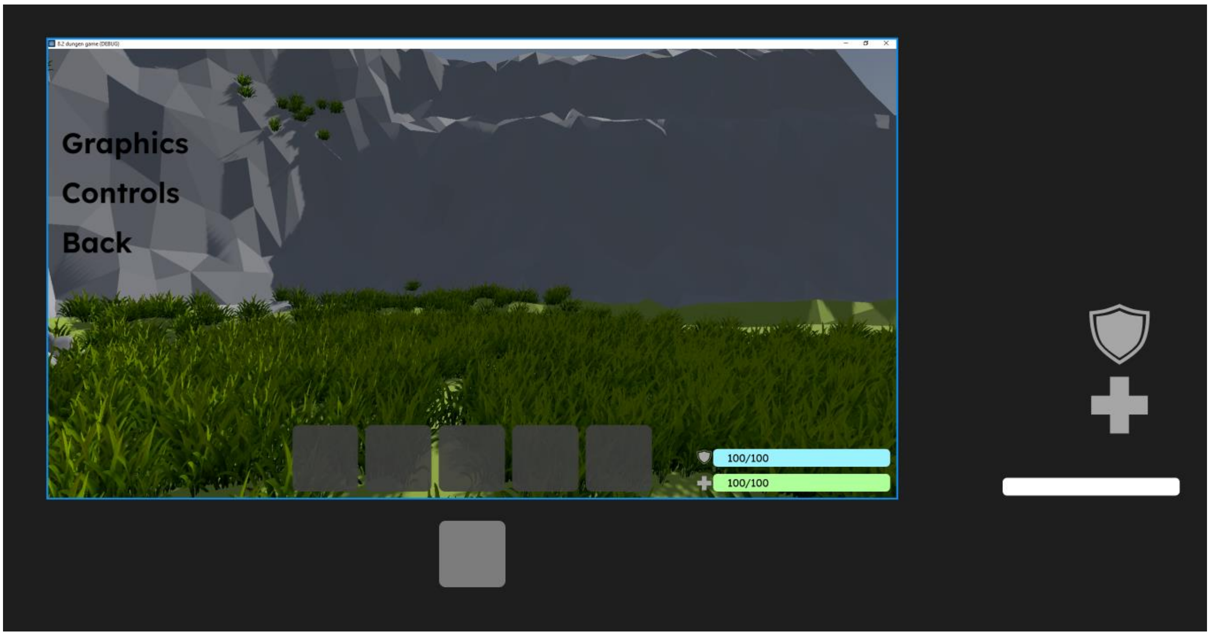
The terrain look good but it was very bare so I added grass, the shop and a firepit.

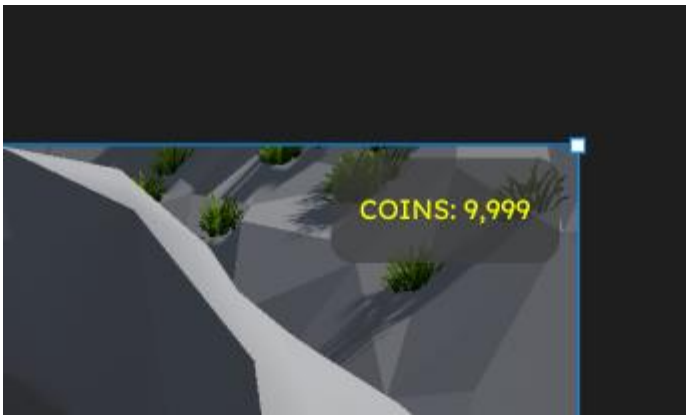


I then exported this to godot 4.03 and did some post-processing to make it look good.

I could now start to think about the GUI. I took this screenshot of the game and imported it into Figma and started to design some of the main menu, starting new game, hotbar, health and coin counter.



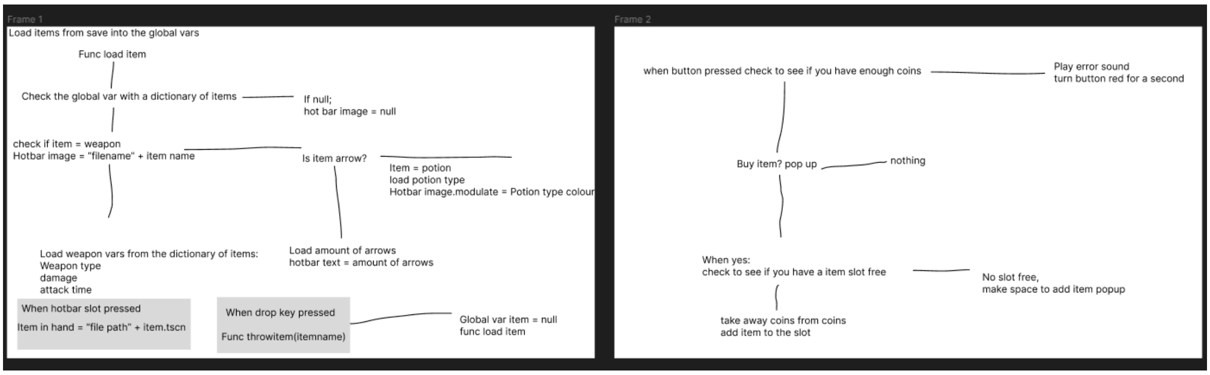




I liked these designs so much that I exported the individual components as pngs so I could use them as the real health bar, coin counter, hot bar etc.

I then started to look for some free to use 3d models to download as use as my player and enemies. I made all the other 3d models such as the rooms, pots, rocks etc.

Most of the code was made with trial and error however I did have to write out a flowchart for some of my more complicated code such as my inventory code so I could understand it better.



Frame 1

Load items from save into the global vars

Func load item

Check the global var with a dictionary of items

If null;
hot bar image = null

check if item = weapon
Hotbar image = "filename" + item name

Is item arrow?

Item = potion
load potion type
Hotbar image.modulate = Potion type colour

Load weapon vars from the dictionary of items:
Weapon type
damage
attack time

Load amount of arrows
hotbar text = amount of arrows

When hotbar slot pressed
Item in hand = "file path" + item.tscn

When drop key pressed
Func throwitem(itemname)

Global var item = null
func load item

Frame 2

when button pressed check to see if you have enough coins

Play error sound
turn button red for a second

Buy item? pop up

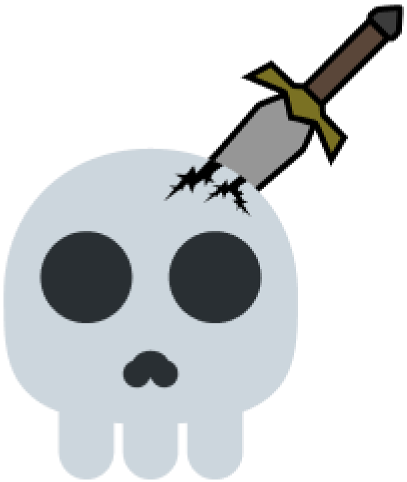
nothing

When yes:
check to see if you have a item slot free

No slot free,
make space to add item popup

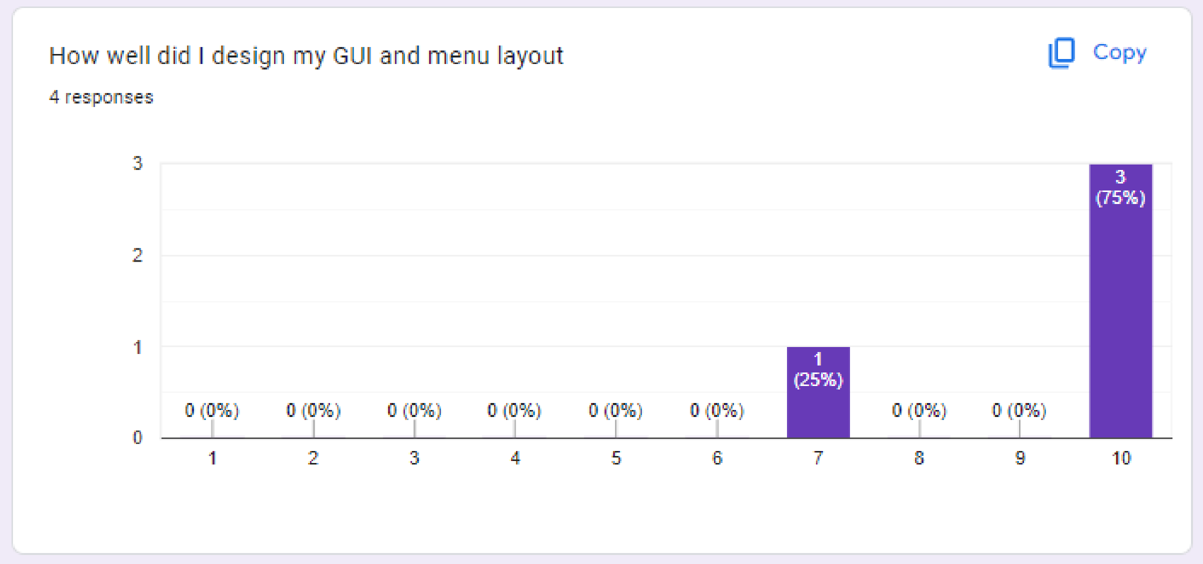
take away coins from coins
add item to the slot

I also made a last-second logo for the game:



Feedback on the designs

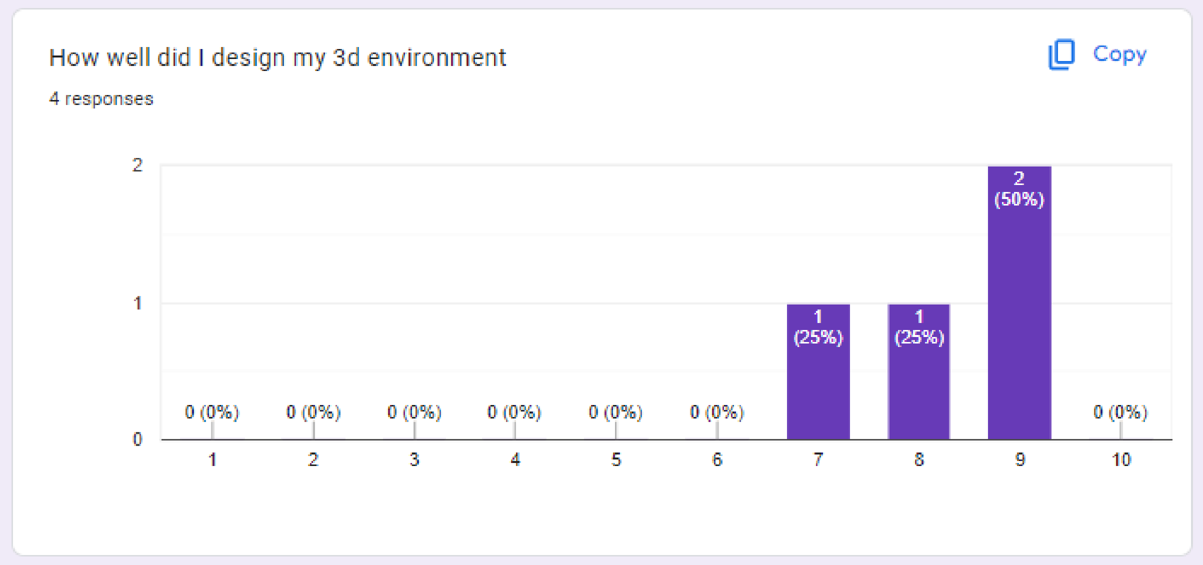
The Feedback on the designs on my GUI and menu layout was received very well and most people voted 10/10.



How well did I design my GUI and menu layout

4 responses

The feedback on for the 3D environment was not as well received and later on people said I should have added more detail to it.

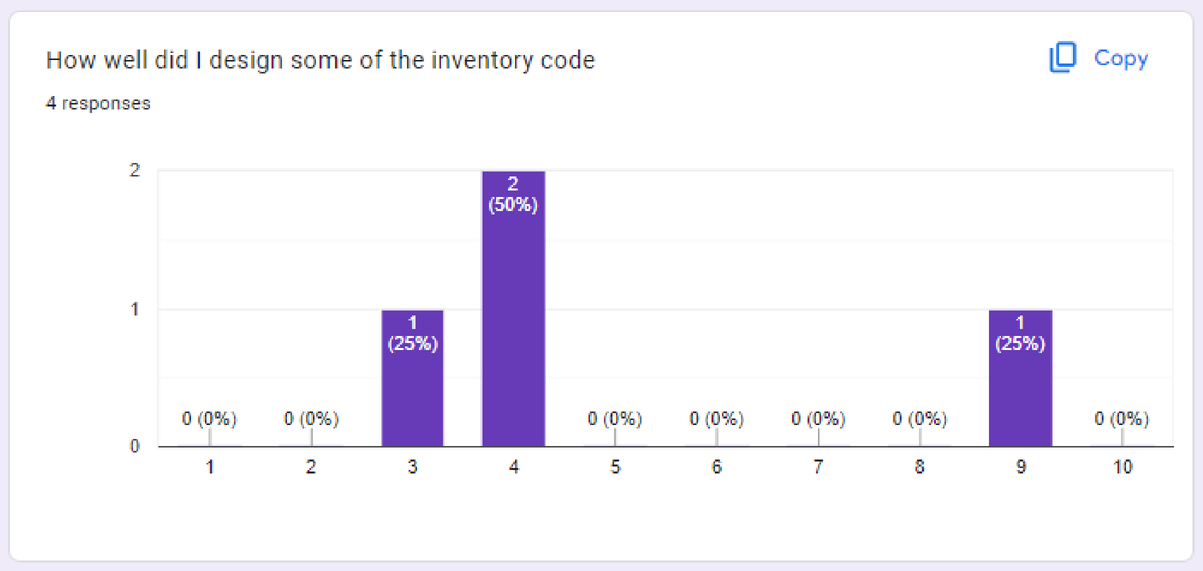


How well did I design my 3d environment

4 responses

The feedback was very negative on the inventory code and got a very low score. And later people said the flowchart was very messy

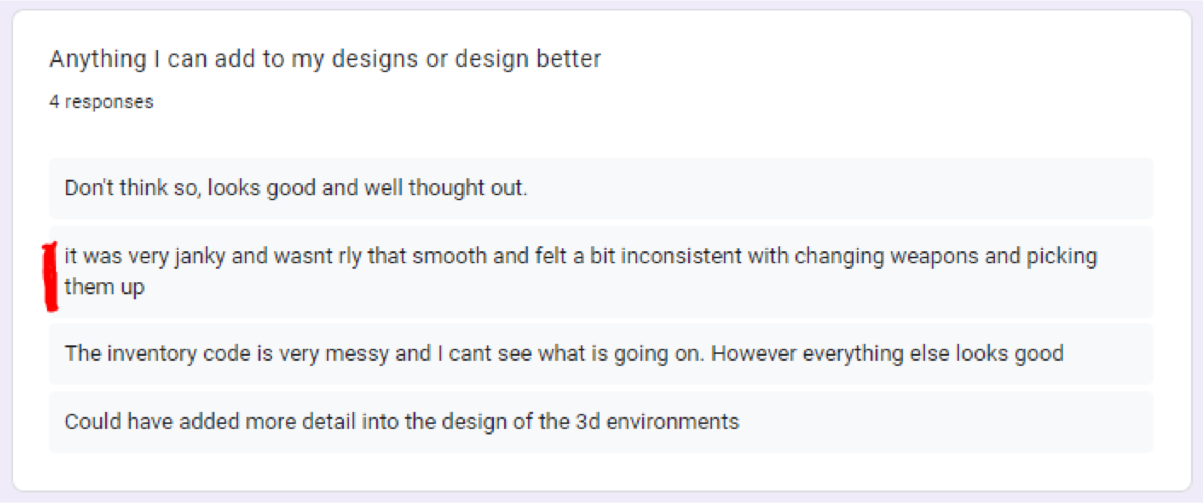
and hard to see.



How well did I design some of the inventory code

4 responses

When asking for overall feedback and anything I can add to my designs or what I could have done better this is what I got.



Anything I can add to my designs or design better

4 responses

Don't think so, looks good and well thought out.

it was very janky and wasnt rly that smooth and felt a bit inconsistent with changing weapons and picking them up

The inventory code is very messy and I cant see what is going on. However everything else looks good

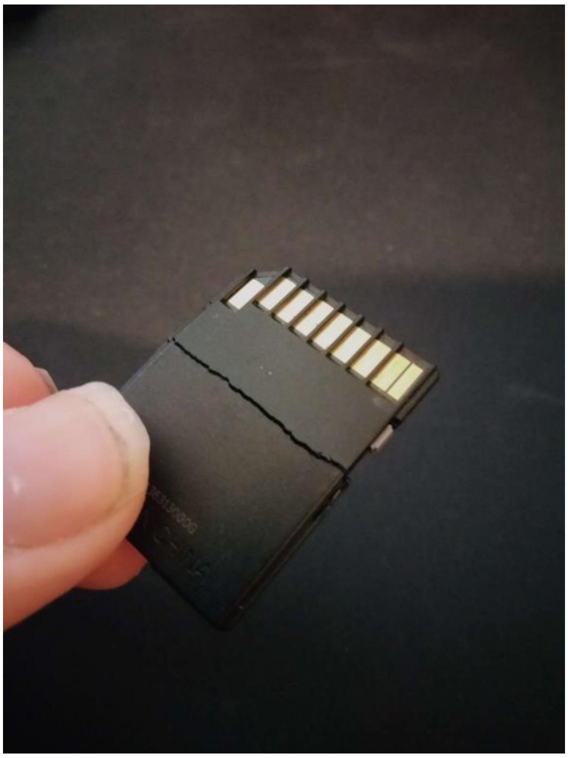
Could have added more detail into the design of the 3d environments

I think the one marked in red was referencing the inventory code I made but didn't actually talk about the design of it, just how it worked in-game.

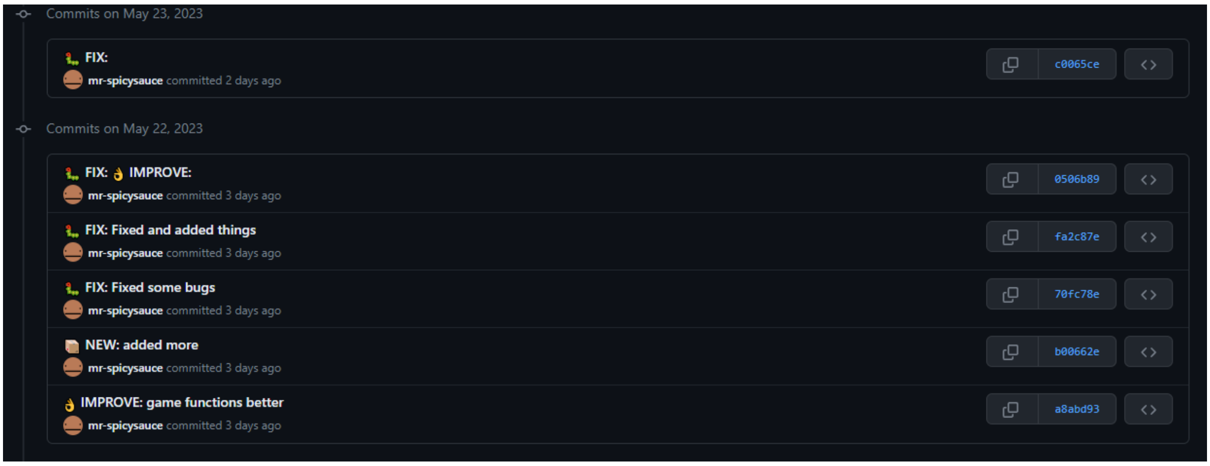
Overall everyone was happy with the designs and how they looked, but I went to go add more details to some of the designs and apply the feedback given.

Development of the game

The development of the game was long and hard and involved a lot of trial and error as I had to learn new things such as AI navigation through a 3D world. In order to help me undertake this massive project I *had* a to-do list, but unfortunately, my SD card got snapped in half while transporting it from college to home and I lost a fair amount of the work I did on the train including my to-do list.



After this, I immediately switched to using GitHub to transfer files and keep a backup so that I always have a copy online in case something like this happens again.



Commits on May 23, 2023

🐛 FIX:
mr-spicysauce committed 2 days ago
c0065ce

Commits on May 22, 2023

🐛 FIX: 👌 IMPROVE:
mr-spicysauce committed 3 days ago
0506b89

🐛 FIX: Fixed and added things
mr-spicysauce committed 3 days ago
fa2c87e

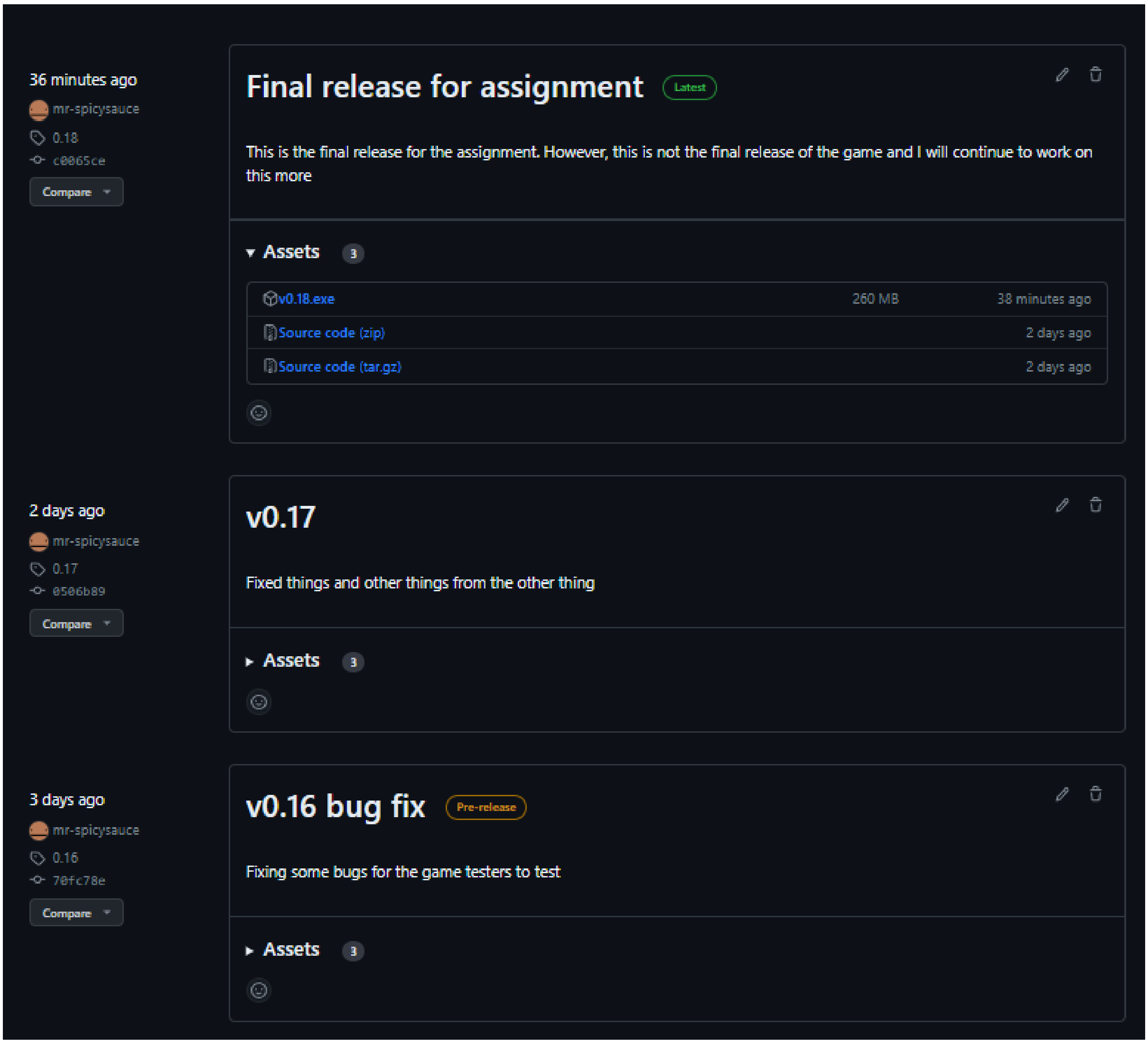
🐛 FIX: Fixed some bugs
mr-spicysauce committed 3 days ago
70fc78e

📦 NEW: added more
mr-spicysauce committed 3 days ago
b00662e

👌 IMPROVE: game functions better
mr-spicysauce committed 3 days ago
a8abd93

This also helped me to quickly publish new versions of the game for me and my friends to try out and give feedback on what needs to be

improved and added to the game.



36 minutes ago
mr-spicysauce
0.18
c0065ce
Compare

Final release for assignment

Latest

This is the final release for the assignment. However, this is not the final release of the game and I will continue to work on this more

Assets 3

v0.18.exe	260 MB	38 minutes ago
Source code (zip)		2 days ago
Source code (tar.gz)		2 days ago

2 days ago
mr-spicysauce
0.17
0506b89
Compare

v0.17

Fixed things and other things from the other thing

Assets 3

3 days ago
mr-spicysauce
0.16
70fc78e
Compare

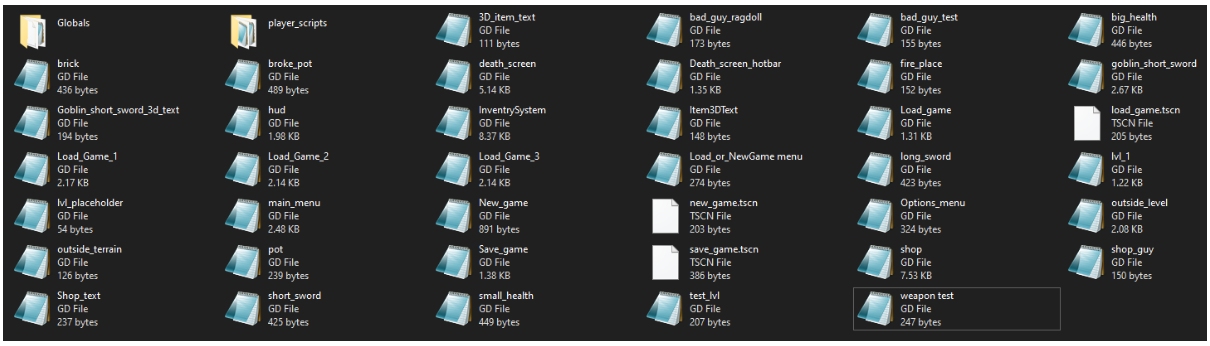
v0.16 bug fix

Pre-release

Fixing some bugs for the game testers to test

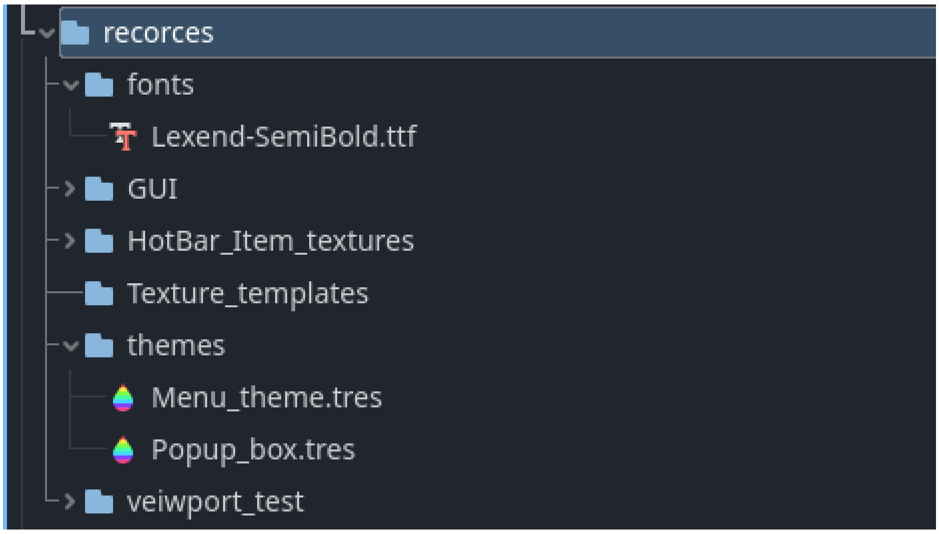
Assets 3

One of the biggest challenges for a game of even this size is the organisation and its right palace. In this game, I have 284 files spread apart in 28 folders. Even with just the code, I have almost 50 separate script files and 100s of lines of code between them.



Globals		
player_scripts		
3D_item_text	GD File	111 bytes
bad_guy_ragdoll	GD File	173 bytes
bad_guy_test	GD File	155 bytes
big_health	GD File	446 bytes
brick	GD File	436 bytes
broke_pot	GD File	489 bytes
death_screen	GD File	5.14 KB
Death_screen_hotbar	GD File	1.35 KB
fire_place	GD File	152 bytes
goblin_short_sword	GD File	2.67 KB
Goblin_short_sword_3d_text	GD File	194 bytes
hud	GD File	1.98 KB
InventrySystem	GD File	8.37 KB
Item3DText	GD File	148 bytes
Load_game	GD File	1.31 KB
load_game.tscn	TSCN File	205 bytes
Load_Game_1	GD File	2.17 KB
Load_Game_2	GD File	2.14 KB
Load_Game_3	GD File	2.14 KB
Load_or_NewGame menu	GD File	274 bytes
long_sword	GD File	423 bytes
lvl_1	GD File	1.22 KB
lvl_placeholder	GD File	54 bytes
main_menu	GD File	2.48 KB
New_game	GD File	891 bytes
new_game.tscn	TSCN File	203 bytes
Options_menu	GD File	324 bytes
outside_level	GD File	2.08 KB
outside_terrain	GD File	126 bytes
pot	GD File	239 bytes
Save_game	GD File	1.38 KB
save_game.tscn	TSCN File	386 bytes
shop	GD File	7.53 KB
shop_guy	GD File	150 bytes
Shop_text	GD File	237 bytes
short_sword	GD File	425 bytes
small_health	GD File	449 bytes
test_lvl	GD File	207 bytes
weapon test	GD File	247 bytes

The thing I did from the very beginning was to make a folder structure so that everything has its place. For example, I have a folder for my resources and then inside that folder, I have more folders for fonts, GUI items, themes etc.

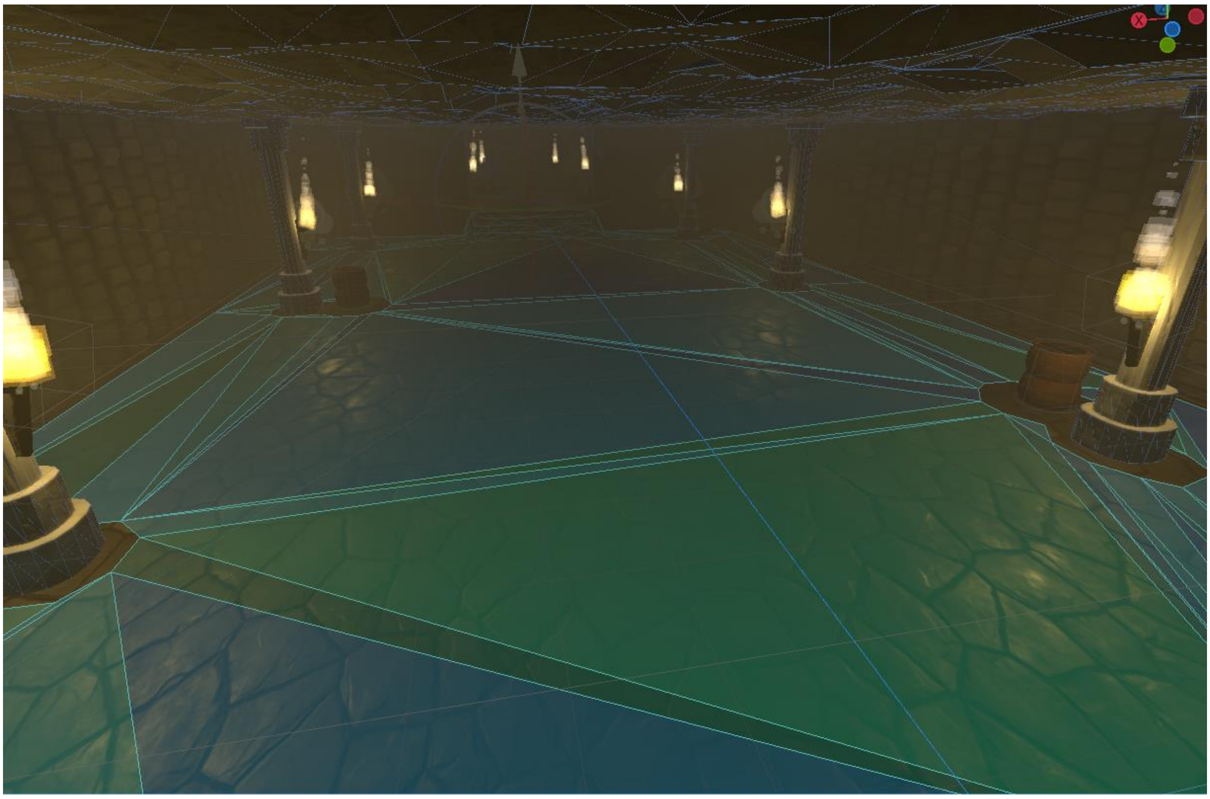


```
└ recorces
  ├ fonts
  │ └ Lexend-SemiBold.ttf
  ├ GUI
  ├ HotBar_Item_textures
  ├ Texture_templates
  ├ themes
  │ ├ Menu_theme.tres
  │ └ Popup_box.tres
  └ veiwport_test
```

If I didn't have this data structure, many different types of files would all be inside 1 area and I would have to spend a lot of time sorting through things to try and find the thing I am looking for.

One thing I did during development to help with the 3d modelling process was I looked up reference images to give me inspiration for how things were going to look. Here are some examples that I looked at:

By far one of the most important things in the game was how the enemies moved and I got to learn about how to generate a navigation mesh so the enemies would not walk into walls when trying to get to the player. You can see the navigation mesh overlay in blue on this screenshot:

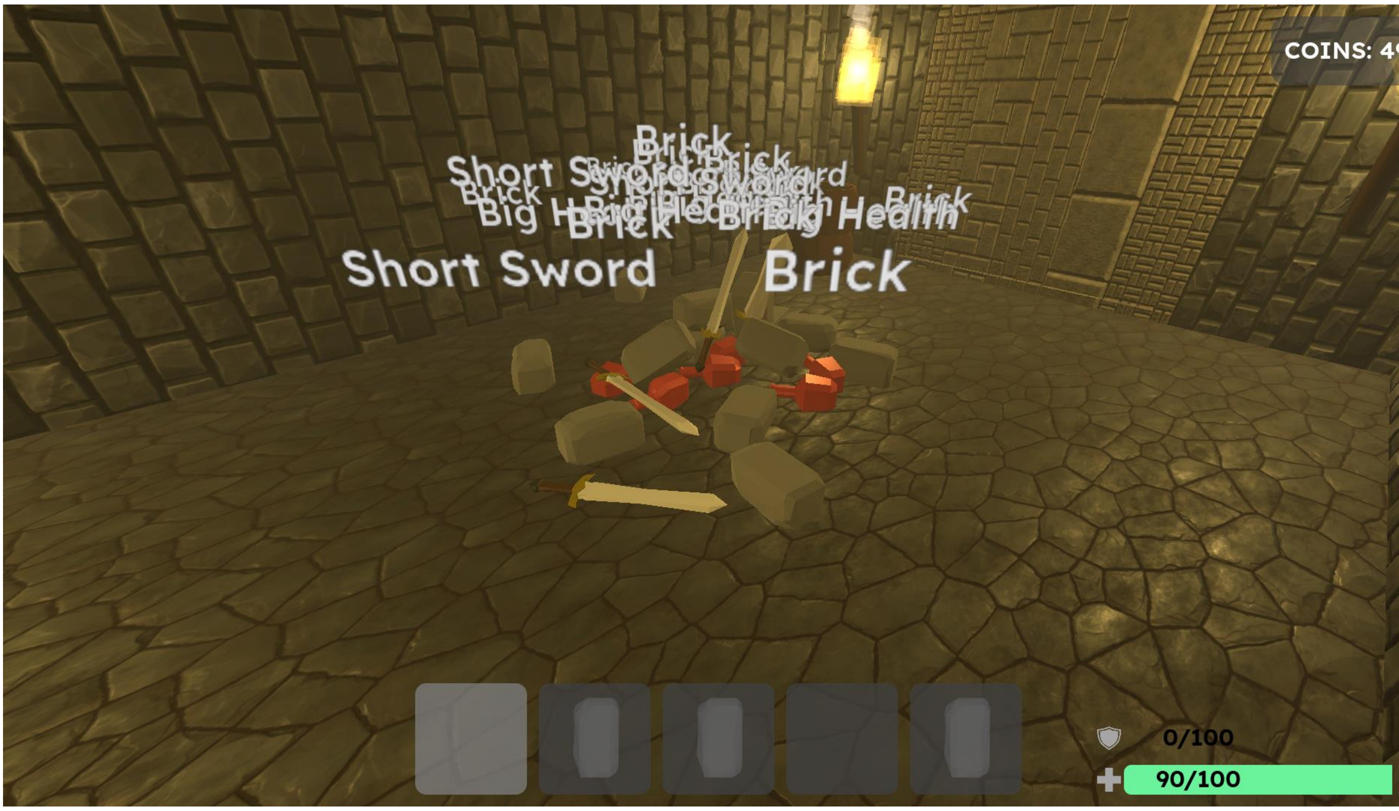


One limitation of this navigation mesh is that it makes it a bit harder to make a procedurally generated world, moving the pots and pillars around randomly etc etc as I would have to generate a new mesh map every time.

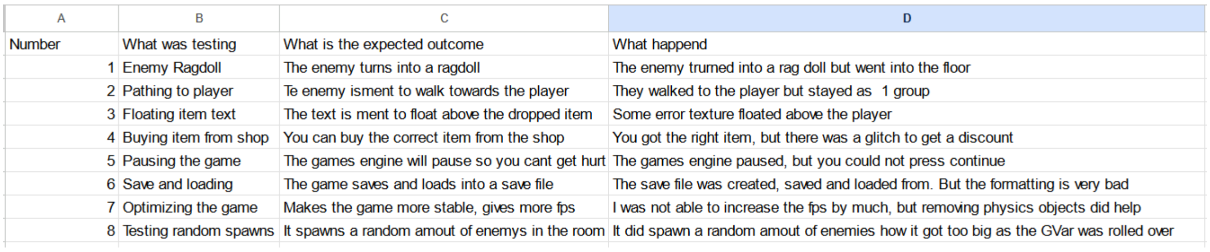
Testing the game

The testing of the game was probably the best part of the process. As the game got more and more complex more things can go wrong such as duplication glitches for items which were very funny to fill

an area with bricks.

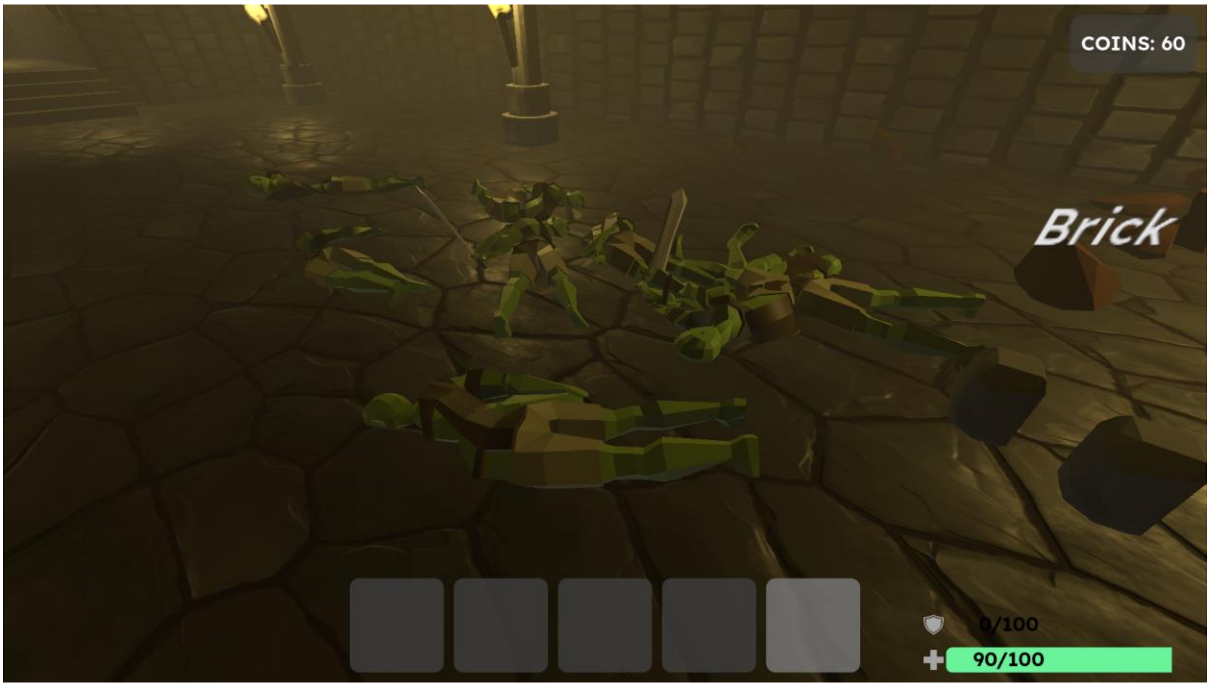


It is very hard to list the number of interactions that had to be tested (there were literally 100s of bugs) but I will list some of the most significant ones that I had trouble with and my test plan for them. Screenshots have been added below for context.

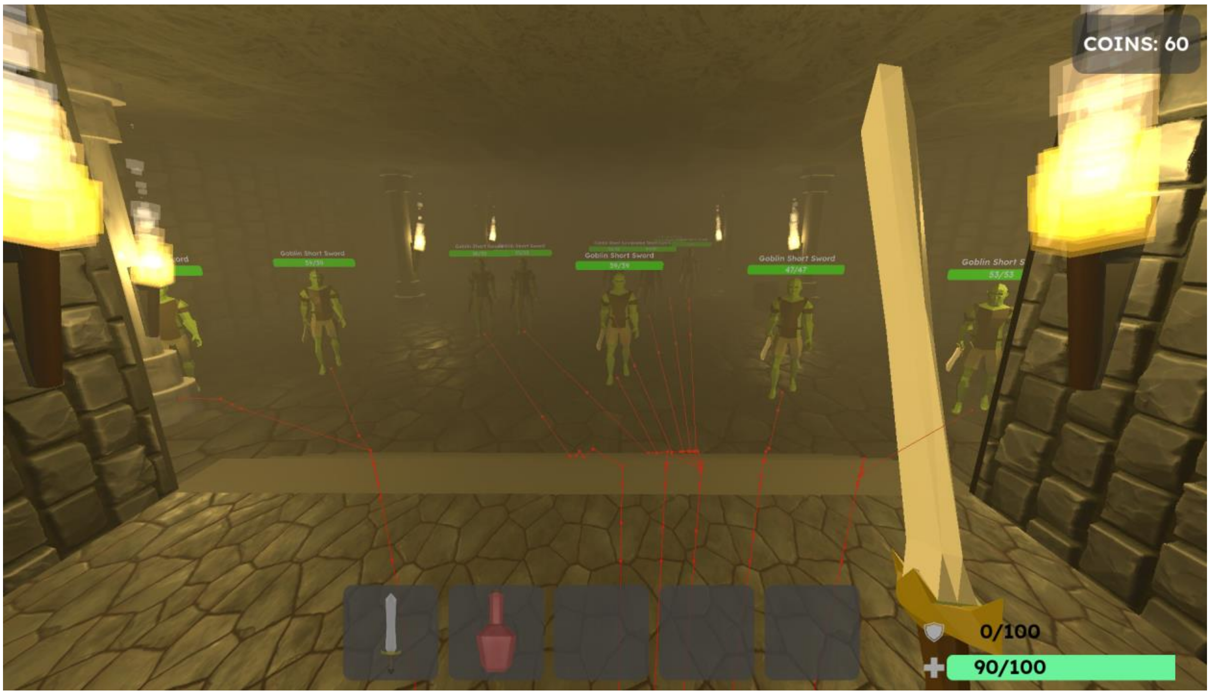


A	B	C	D
Number	What was testing	What is the expected outcome	What happend
1	Enemy Ragdoll	The enemy turns into a ragdoll	The enemy trurned into a rag doll but went into the floor
2	Pathing to player	Te enemy isment to walk towards the player	They walked to the player but stayed as 1 group
3	Floating item text	The text is ment to float above the dropped item	Some error texture floated above the player
4	Buying item from shop	You can buy the correct item from the shop	You got the right item, but there was a glitch to get a discount
5	Pausing the game	The games engine will pause so you cant get hurt	The games engine paused, but you could not press continue
6	Save and loading	The game saves and loads into a save file	The save file was created, saved and loaded from. But the formatting is very bad
7	Optimizing the game	Makes the game more stable, gives more fps	I was not able to increase the fps by much, but removing physics objects did help
8	Testing random spawns	It spawns a random amout of enemys in the room	It did spawn a random amout of enemies how it got too big as the GVar was rolled over

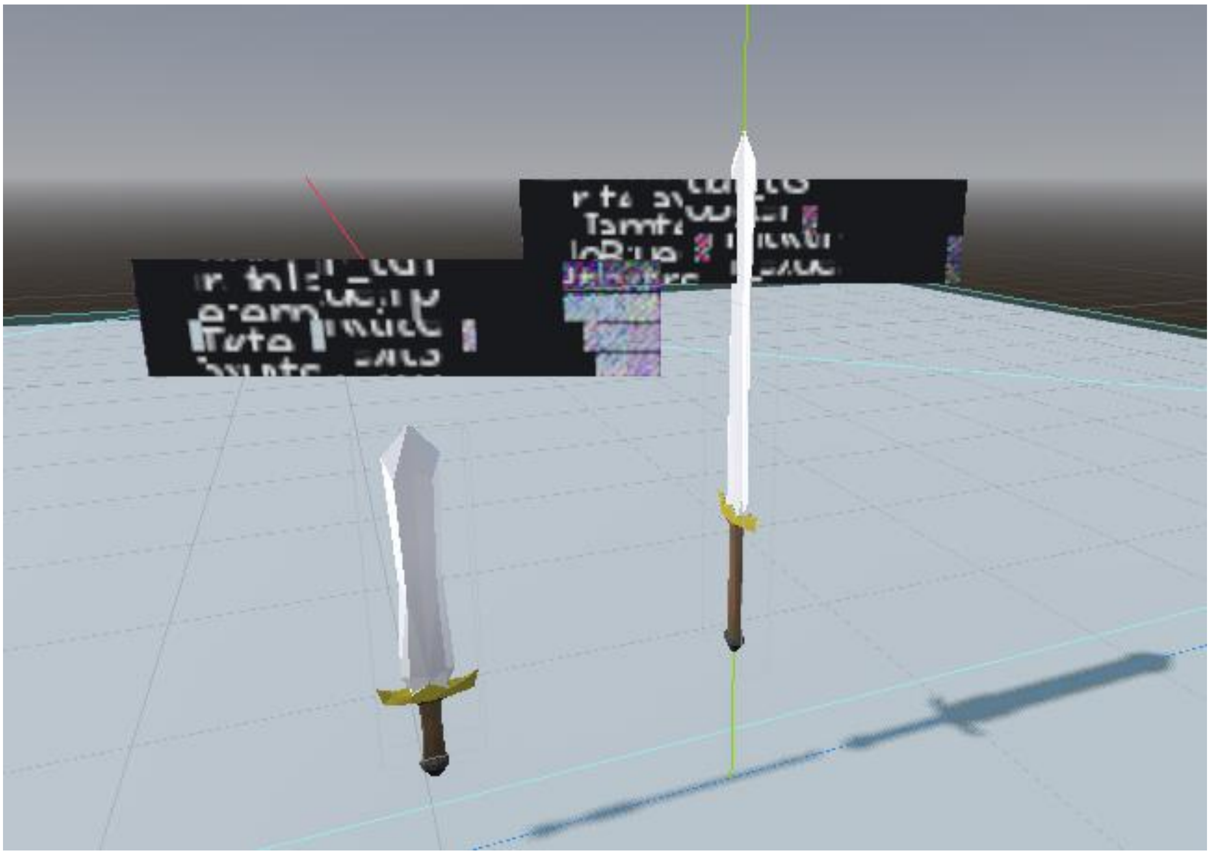
1. Enemy Ragdoll:

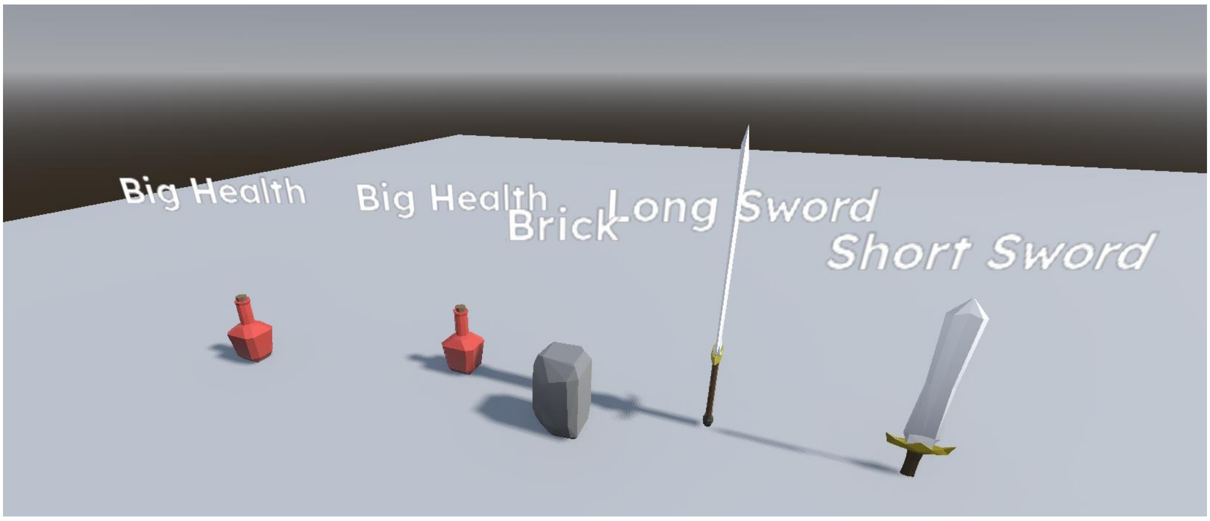


2. **Enemy pathing to player:**

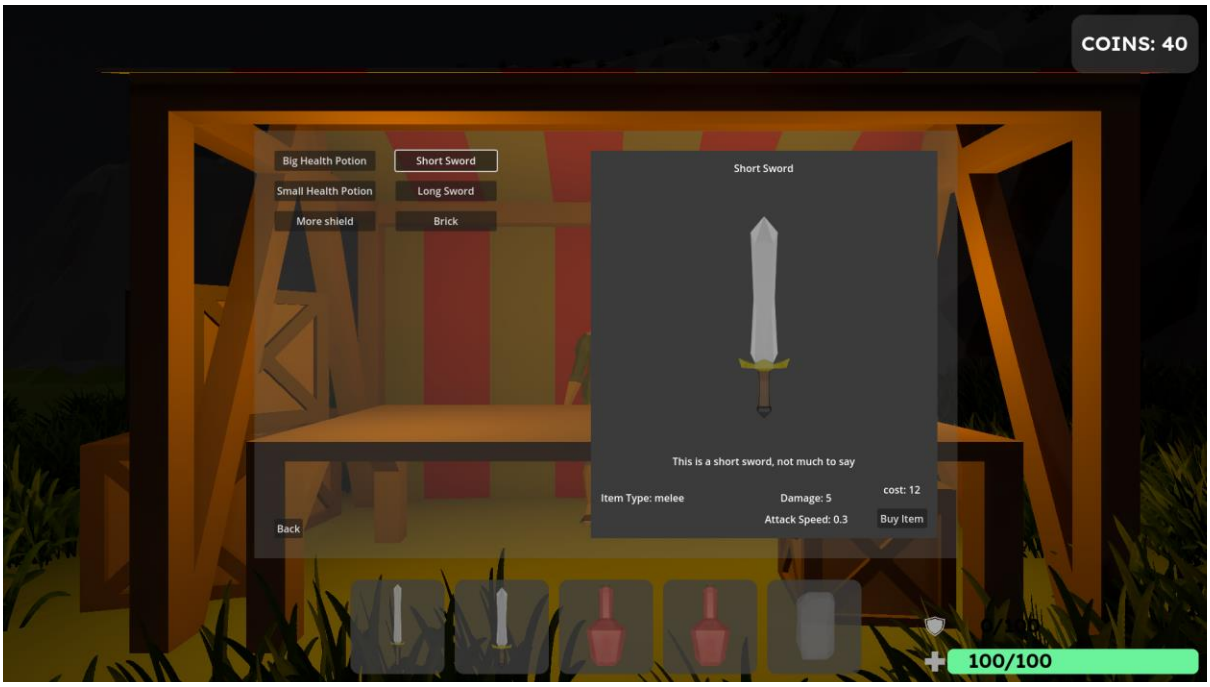


3. **Floating item text:**





4. **Buying the correct items from the shop:**



Short Sword

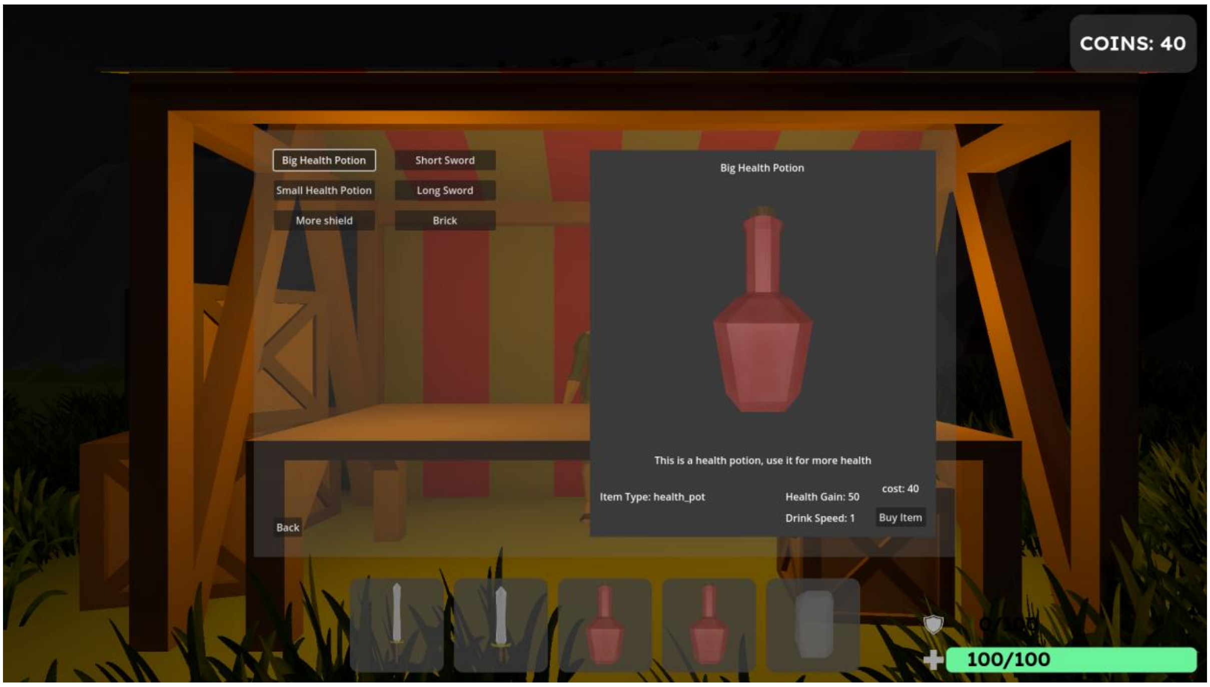
This is a short sword, not much to say

Item Type: melee

Damage: 5

Attack Speed: 0.3

cost: 12



Big Health Potion

This is a health potion, use it for more health

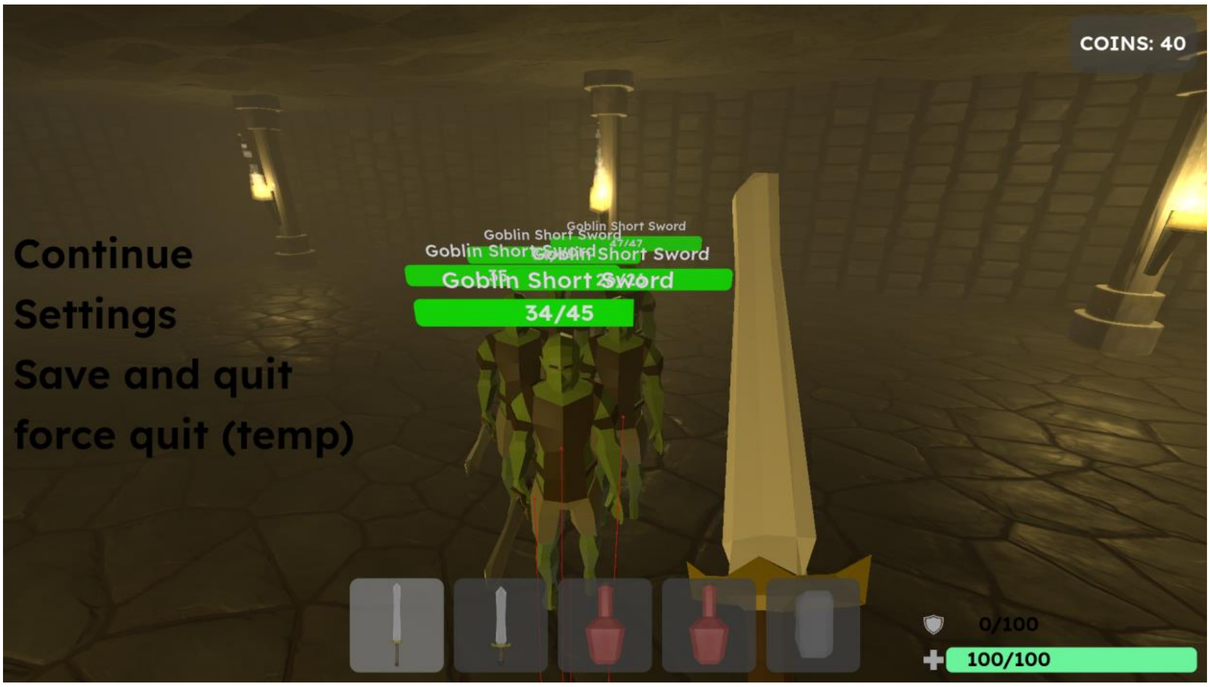
Item Type: health_pot

Health Gain: 50

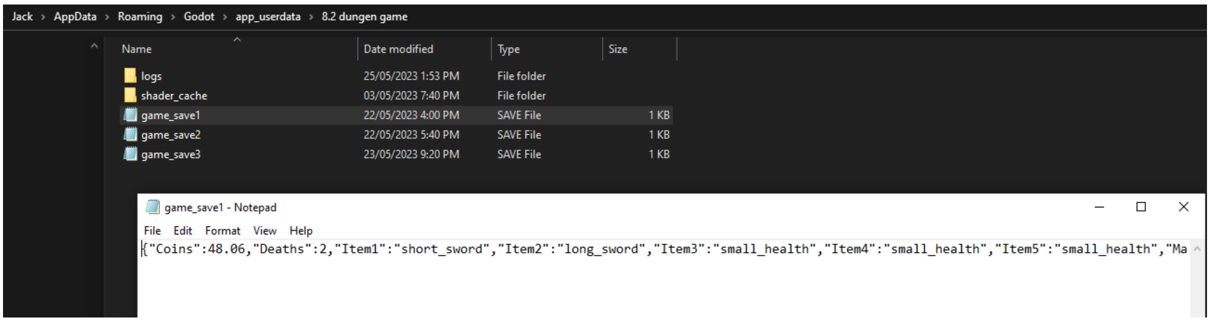
Drink Speed: 1

cost: 40

5. Pausing the game, mid-game and getting the GUI to work:



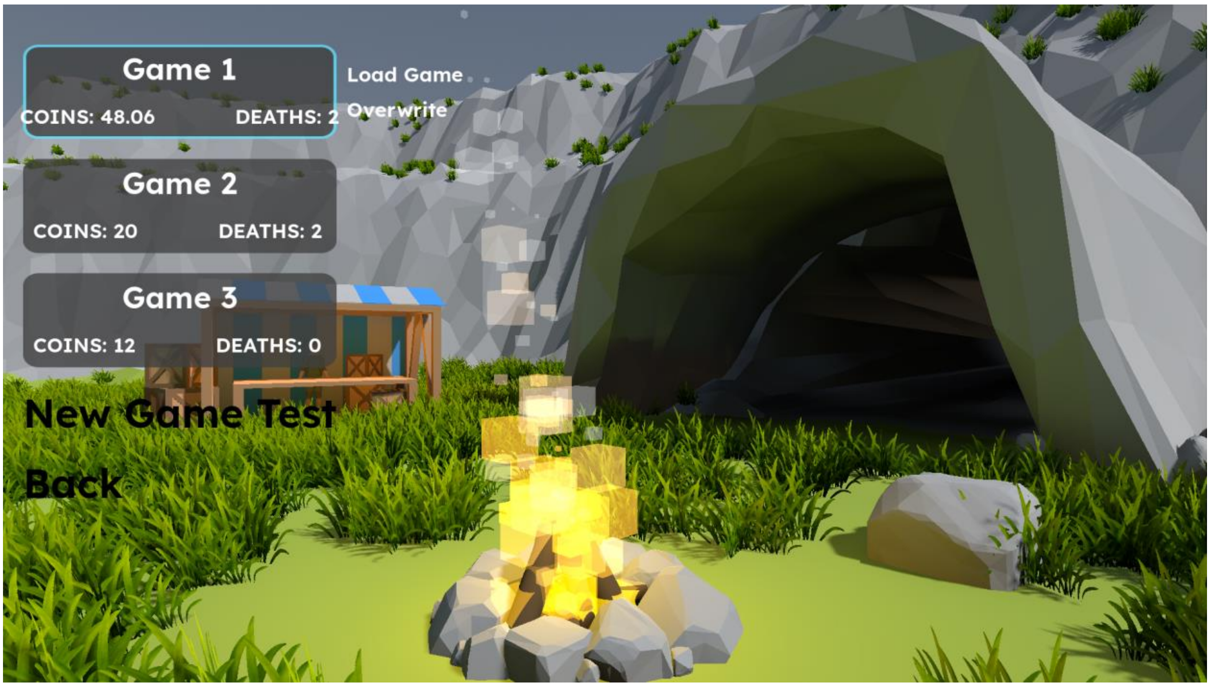
6. Savings and loading of the game:



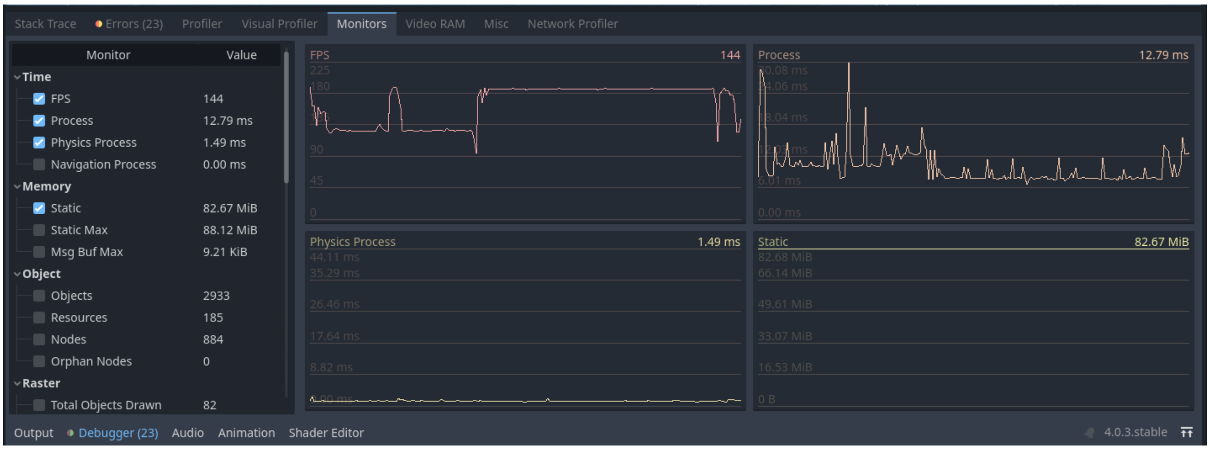
Jack › AppData › Roaming › Godot › app_userdata › 8.2 dungen game

Name	Date modified	Type	Size
logs	25/05/2023 1:53 PM	File folder	
shader_cache	03/05/2023 7:40 PM	File folder	
game_save1	22/05/2023 4:00 PM	SAVE File	1 KB
game_save2	22/05/2023 5:40 PM	SAVE File	1 KB
game_save3	23/05/2023 9:20 PM	SAVE File	1 KB

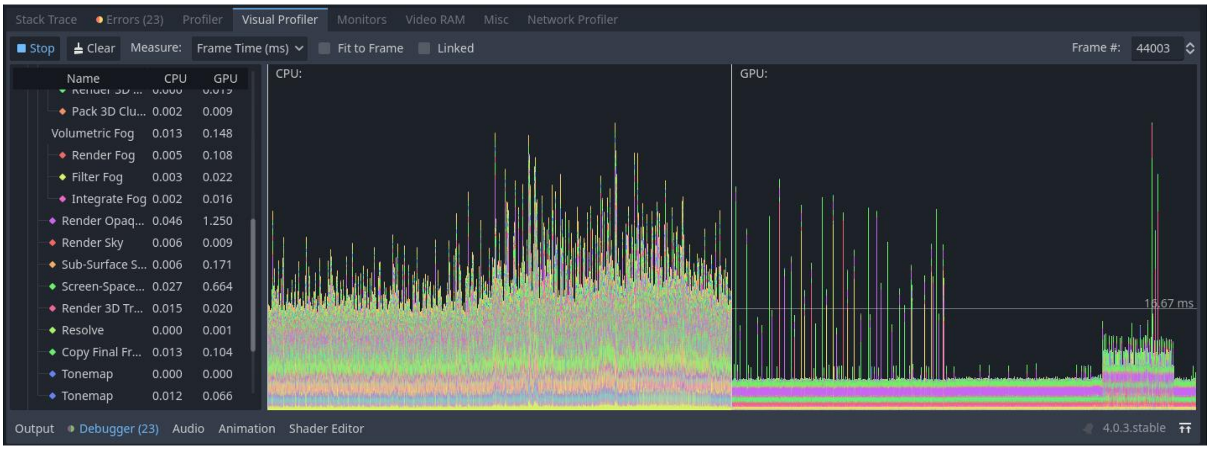
```
{"Coins":48.06,"Deaths":2,"Item1":"short_sword","Item2":"long_sword","Item3":"small_health","Item4":"small_health","Item5":"small_health","Ma
```



7. Optimizing the games performance.



Monitor	Value
Time	
☑ FPS	144
☑ Process	12.79 ms
☑ Physics Process	1.49 ms
□ Navigation Process	0.00 ms
Memory	
☑ Static	82.67 MiB
□ Static Max	88.12 MiB
□ Msg Buf Max	9.21 KiB
Object	
□ Objects	2933
□ Resources	185
□ Nodes	884
□ Orphan Nodes	0
Raster	
□ Total Objects Drawn	82



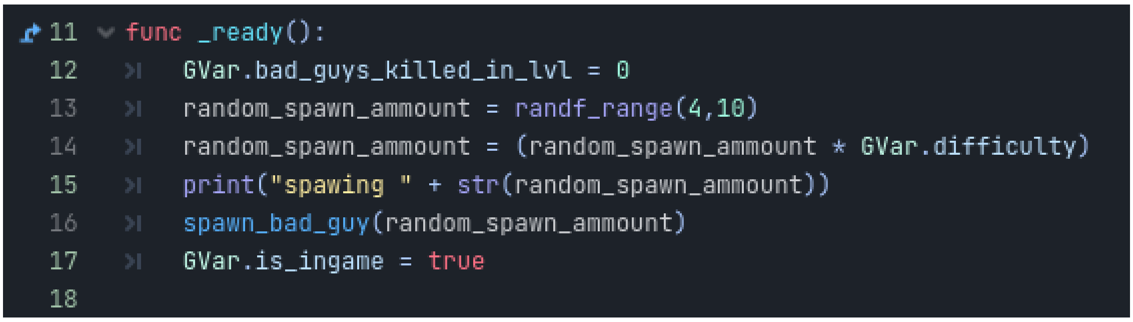
Measure: Frame Time (ms)

□ Fit to Frame □ Linked

Frame #: 44003

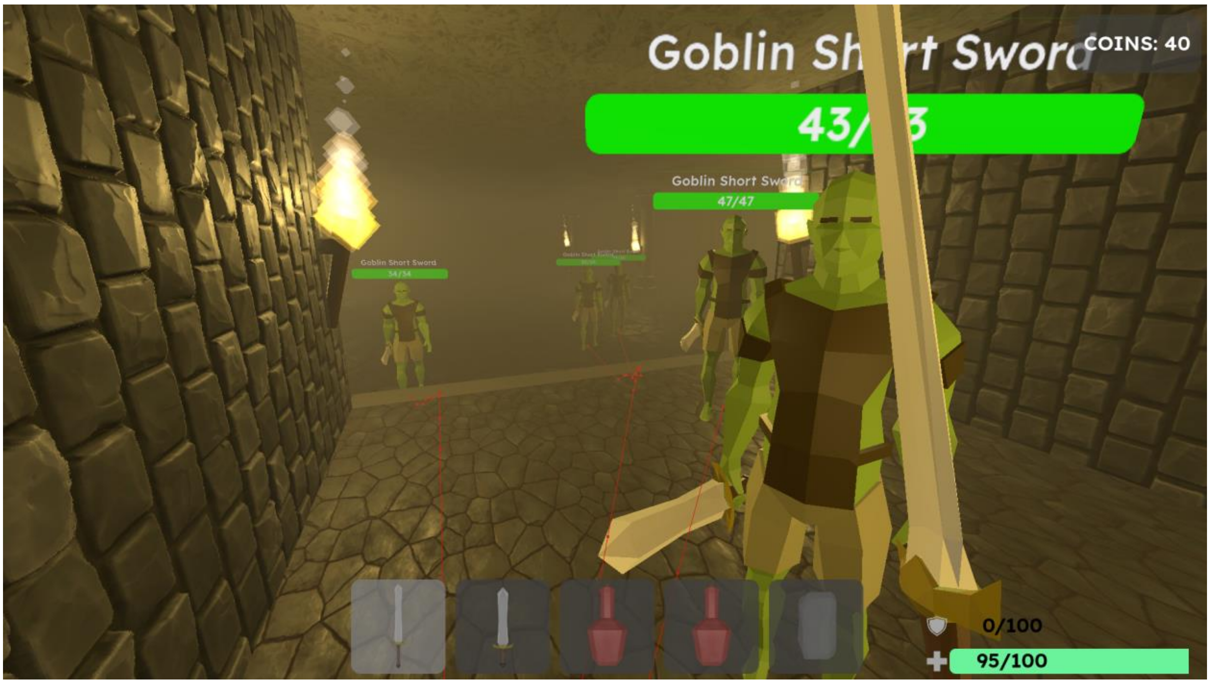
Name	CPU	GPU
Render 3D ...	0.000	0.019
Pack 3D Clu...	0.002	0.009
Volumetric Fog	0.013	0.148
Render Fog	0.005	0.108
Filter Fog	0.003	0.022
Integrate Fog	0.002	0.016
Render Opaq...	0.046	1.250
Render Sky	0.006	0.009
Sub-Surface S...	0.006	0.171
Screen-Space...	0.027	0.664
Render 3D Tr...	0.015	0.020
Resolve	0.000	0.001
Copy Final Fr...	0.013	0.104
Tonemap	0.000	0.000
Tonemap	0.012	0.066

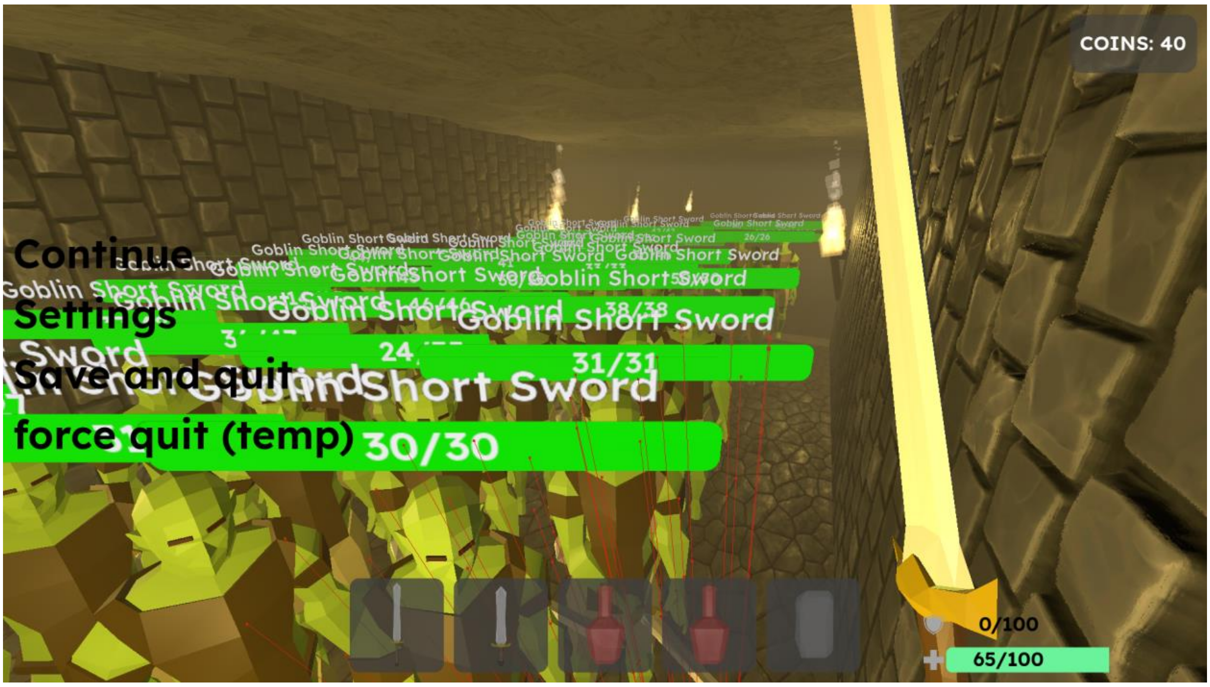
8. Testing the random spawning of enemies:



```
func _ready():
	GVar.bad_guys_killed_in_lvl = 0
	random_spawn_ammount = randf_range(4,10)
	random_spawn_ammount = (random_spawn_ammount * GVar.difficulty)
	print("spawing " + str(random_spawn_ammount))
	spawn_bad_guy(random_spawn_ammount)
	GVar.is_ingame = true

```



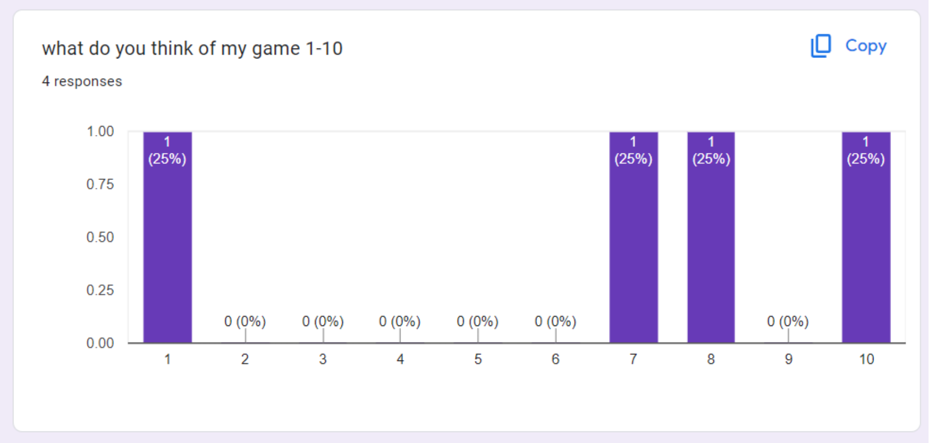


Feedback on the game

After the game was completed and the final version was sent out to the clients I got them to send me my feedback. Unfortunately, one of the clients did not answer the google forms properly and gave an

unfair review of the game saying to "blow up the hard drive" so I will be ignoring his feedback.

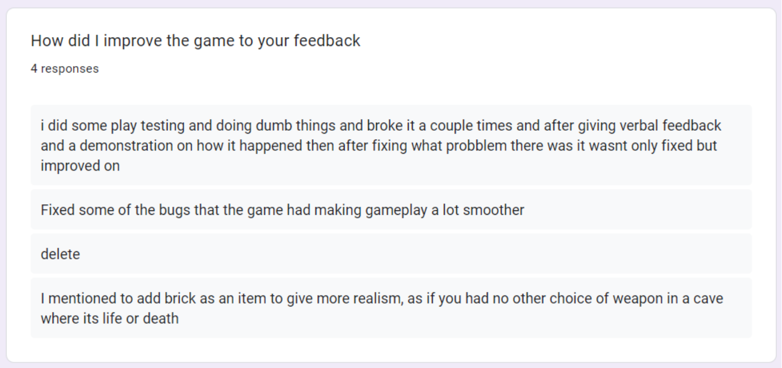
Overall my game was really well received and the clients where actively playing the demos even if they were full of bugs.



what do you think of my game 1-10

4 responses

During the entire game making progress as the clients were testing the game they gave me a lot of feedback and bugs for me to fix and I feel like I did a good job in implementing the feedback. So I asked how well I implemented the feedback they gave me and these are the responses they gave me.



How did I improve the game to your feedback

4 responses

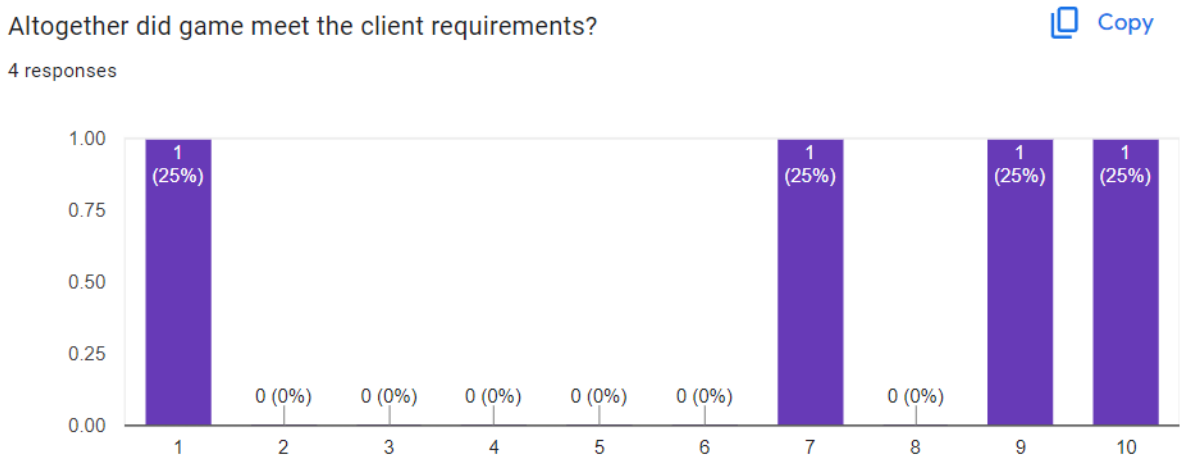
i did some play testing and doing dumb things and broke it a couple times and after giving verbal feedback and a demonstration on how it happened then after fixing what probblem there was it wasnt only fixed but improved on

Fixed some of the bugs that the game had making gameplay a lot smoother

delete

I mentioned to add brick as an item to give more realism, as if you had no other choice of weapon in a cave where its life or death

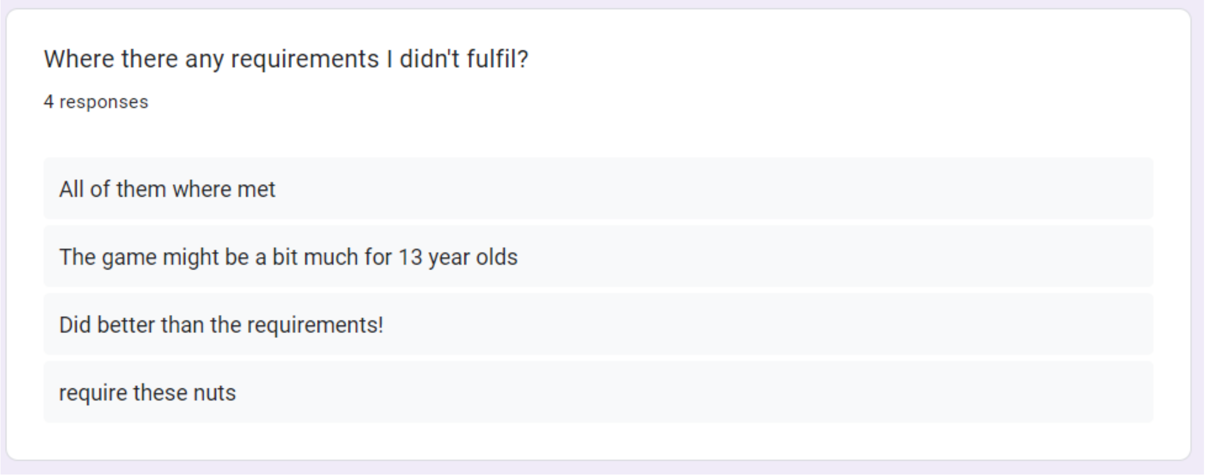
I then asked the clients on a scale of 1-10 how well did I follow their requirements and overall they were very satisfied with how well they were followed.



Altogether did game meet the client requirements?

4 responses

And just to make sure that I didnt miss anything I asked if I didnt fufil any of the requirements and they said that they where all met and one even said I did better than the requirements. However one client did that the game might be a bit much for some 13 year olds and I might have overshot the age requirements. That is something to take into consideration when publishing the game and in the final evaluation of the game.



Where there any requirements I didn't fulfil?

4 responses

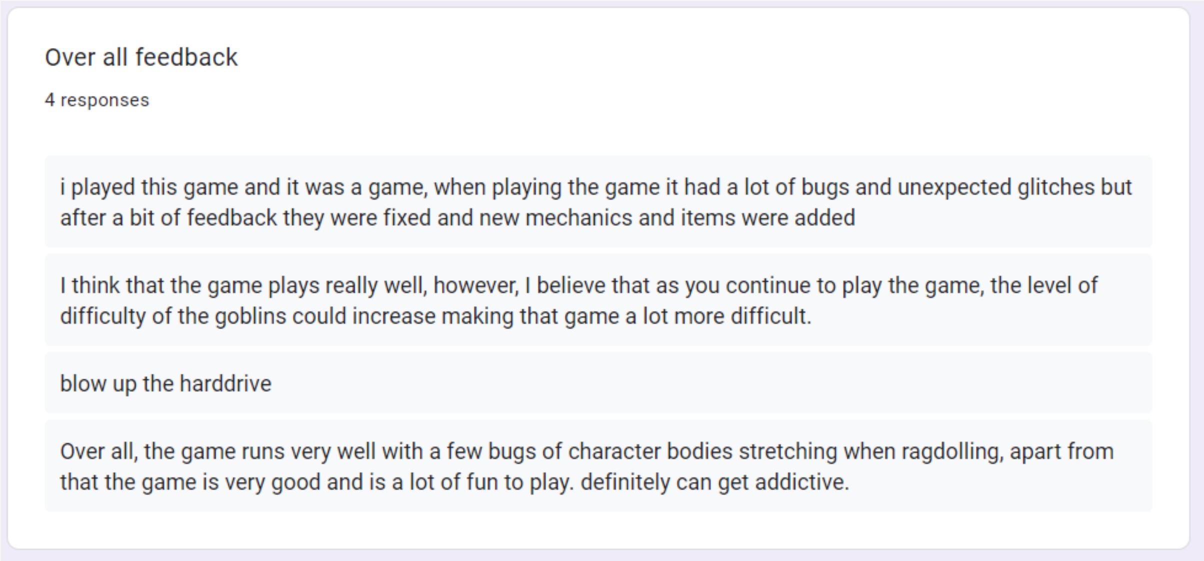
All of them where met

The game might be a bit much for 13 year olds

Did better than the requirements!

require these nuts

Finally I then asked for overall feedback on the game and all the feedback from the clients was very nice and supportive of the game and its overall development.



Over all feedback

4 responses

i played this game and it was a game, when playing the game it had a lot of bugs and unexpected glitches but after a bit of feedback they were fixed and new mechanics and items were added

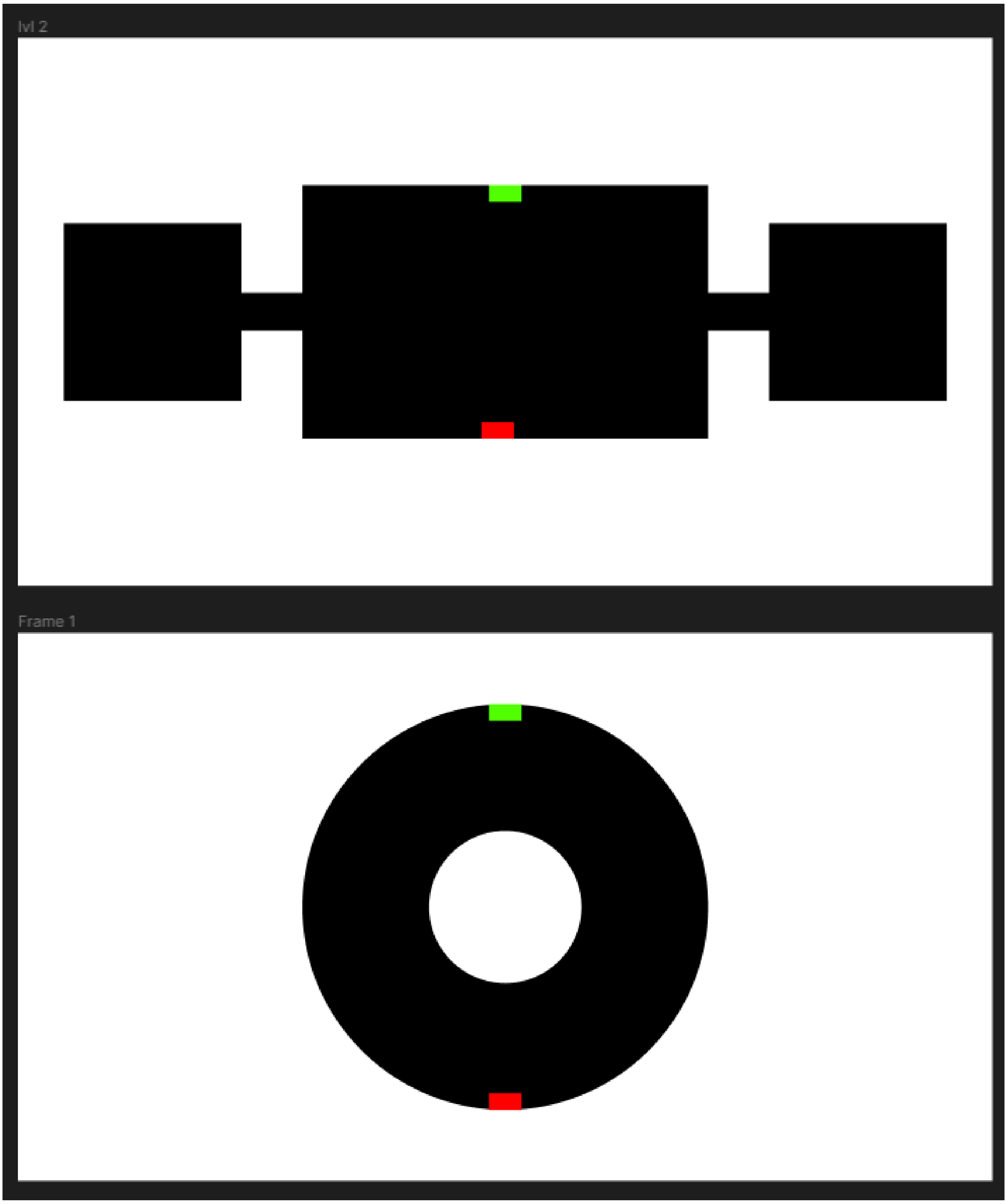
I think that the game plays really well, however, I believe that as you continue to play the game, the level of difficulty of the goblins could increase making that game a lot more difficult.

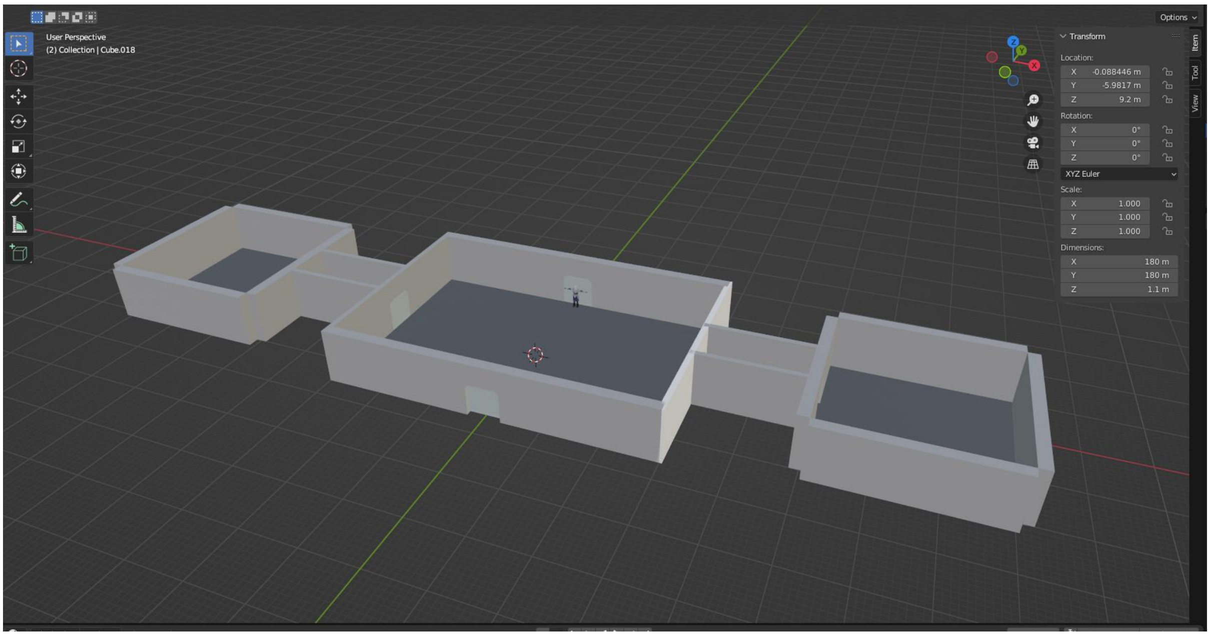
blow up the harddrive

Over all, the game runs very well with a few bugs of character bodies stretching when ragdolling, apart from that the game is very good and is a lot of fun to play. definitely can get addictive.

Evaluating the design and game against client requirements

Looking back at the client's requirements and the game's design I would say they follow the client's requirements very well but there are some improvements that I could have made. One such improvement was to add the designs of the other levels or any levels for that matter. In the design section, I talked about how the 3d modelling was done and how I designed the outside level but now how I was going to make the level randomly generated and what the "shell" of the room would look like. I did however make some basic plans for future levels that could be implemented:


lvl 2
Frame 1



Transform

Location:
X -0.088446 m
Y -5.9817 m
Z 9.2 m

Rotation:
X 0°
Y 0°
Z 0°

XYZ Euler

Scale:
X 1.000
Y 1.000
Z 1.000

Dimensions:
X 180 m
Y 180 m
Z 1.1 m

When looking at the game altogether it follows all the client's requirements however one of the client's requirements might have been broken. Looking at the requirements shows me that the game was meant to be aimed at ages 13-17 and although I did aim it towards these younger ages with the low-poly and simple theme however it does have some aggressive aspects of the game with the fighting and the deaths, and although it's not completely blown the age rating out of the water if the client ever wanted to publish the game they might want to re-think about the age rated and aim it towards 15-18.

Conclusion

In conclusion, the design, development and refinement of the game went really well and it was a lot of fun and I learnt a lot. I got the chance to get better at 3d modelling and animation. I also learnt how to use arrays and dictionaries properly. The client was also very happy with the game and how I was able to use the feedback on the designs and during the development process to optimise the game for their needs and even 1-up their expectations.