COMP 1004 Final Project Plan

☰ Tags	

Project Plan

This section of the document outlines the process undertaken during the planning phase for this COMP1004 project. This document is also under the influence of the initial interim report produced. This first section will highlight the game development life cycle and the sprints taken throughout the project in order to achieve progress within the development of the game. It will also feature the completed UML diagrams prior to development and the backlog of features to be added.

Game Development Life Cycle

As this project is exploring about making a game using HTML, CSS and JavaScript, the type of software development lifecycle it’ll follow is the game development life cycle. And what I followed to be an ideal method for a project like this is agile software engineering. I chose this form of development method as I would constantly be designing a feature within the game and testing it throughout in small chunks until it works as intended which will then be added on to the final product as it builds on.

In order to thoroughly plan this development life cycle further, I will have done a minimum of two sprints for the initial submission of this project – which are chunks of work done incrementally within a set time frame, usually two weeks – where I may follow a rough guideline on what should be completed at what date.

By the end of the project at least 5 sprints would have been finished to show the tasks processed and achieved in the development cycle.

Sprints

From the 2nd of November to the 13th of January of this project, there will have been a minimum of two sprints done for this project, and at least another 5 to be planned later until the project’s final deadline. On the early sprints, I have covered early research on what I would like to develop for this project. This ranged from a very simple text game to a game with full UI elements.

On the first sprint from the 10th of November 2021, I have covered the possible things I could make for the project.

On the second sprint which took place on the 24th of November 2021, I have completed brainstorming ideas for a game to develop for this project. In addition to that, I have also thought of elements to make a game unique.

The most recent sprint on the 8th of December 2021, I have made sure I asked questions to see what I can cover on the marketplace demo with my prototype, this is where I was given feedback where I asked if a tutorial section of a game would suffice and it was given the green light so I started planning for a basic baseline for the game where I could incorporate a tutorial section for the user to follow. Further sprint plan should start to cover parts of the game’s

core
mechanics to make it playable. The initial plan goes from:

- Making the tutorial space
- Making the base overworld environment
- Making the base environment where the user can battle with enemies

Sprint 4 at January 27: After feedback from the initial demo and first report submission, I have decided to decrease the scope of the project because of how taxing it could be to achieve everything I planned initially. **The plan is now to simply implement an environment the player can move around in and interact with NPCs in battle until the last one between different environments.**

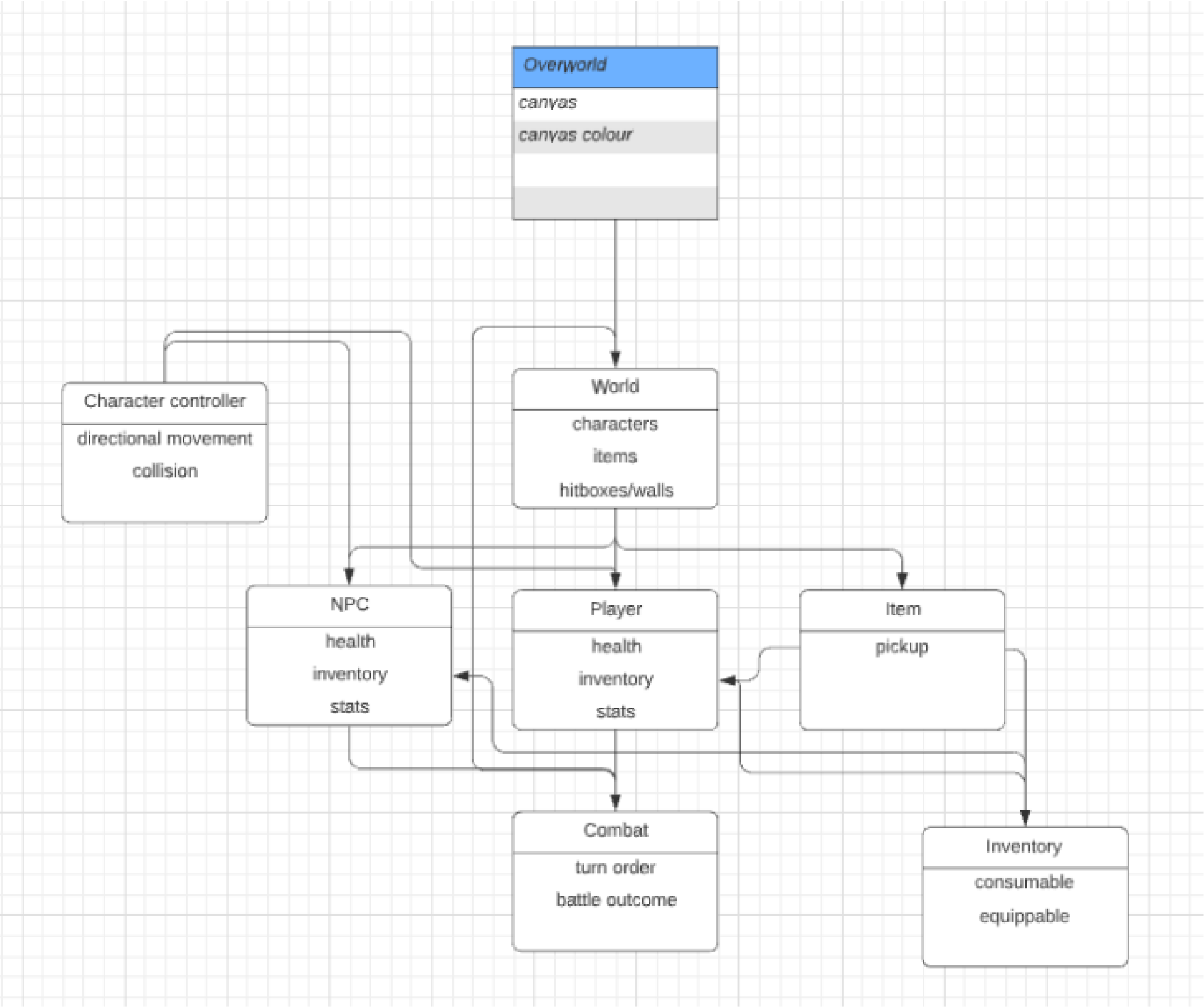
Sprint 5 at February 9: Development continues on the game again, implementing basic NPCs that can move around, shares the same spritesheet as the main character just coloured differently in order to reduce time needed to create sprites.

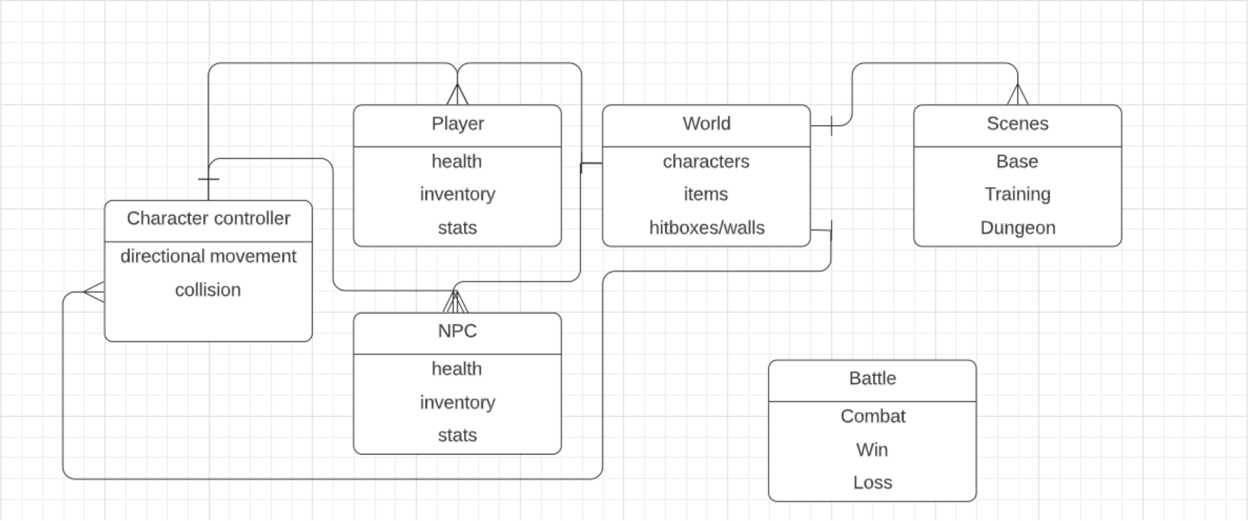
Sprint 6 at March 9: Currently there is a bug with initiating cutscenes with the player that needs to be resolved in order to proceed with the game. The cutscene system implemented is responsible for being able to interact with the NPCs and move them around when wanted as a cutscene and be able to do battles.

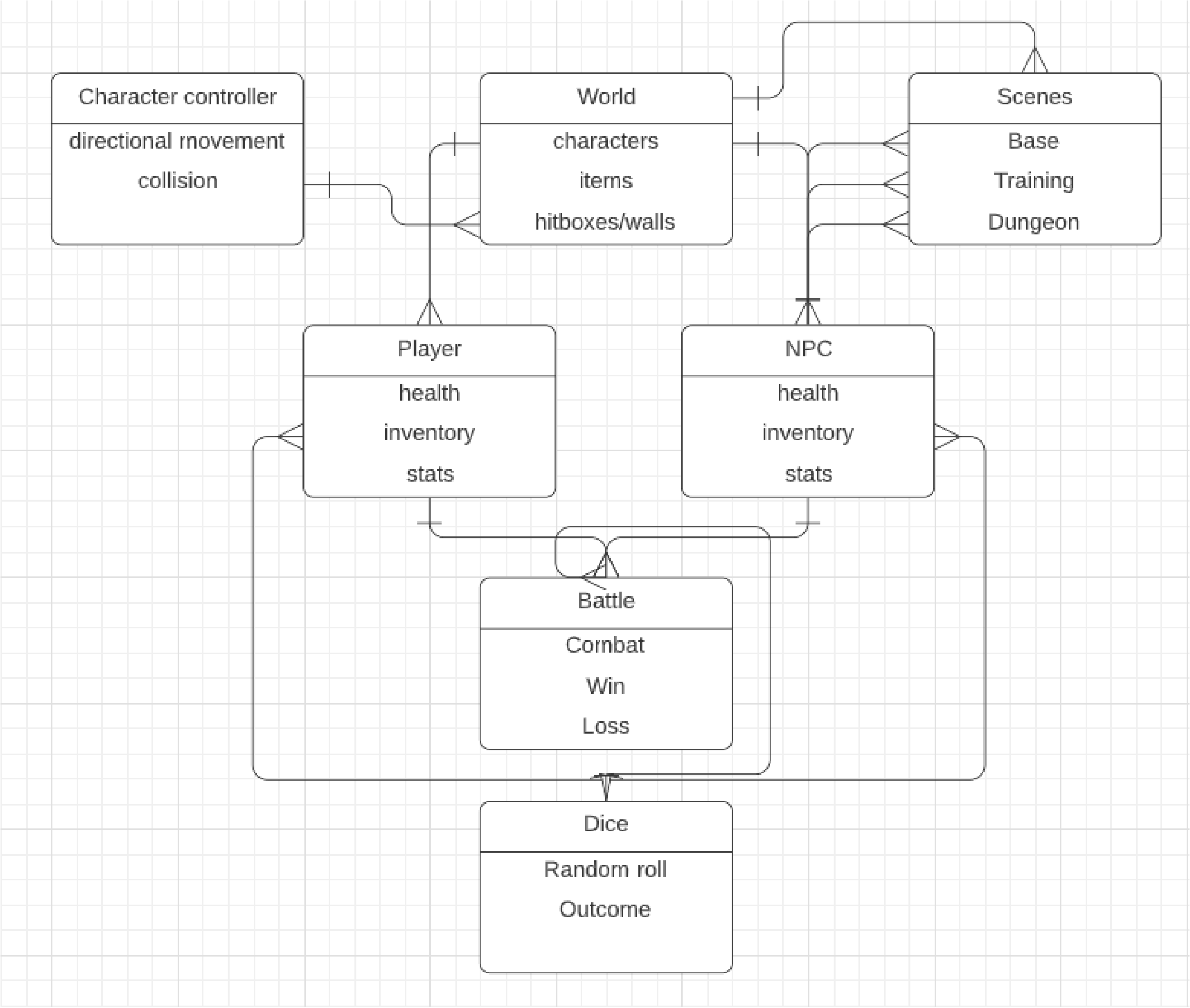
Sprint 7 at March 30: Created other assets for future use

Sprint 8 at April 6:

UML Diagrams







Backlog

The product backlog for this document will just list out features that I want to be implemented/need to be implemented.

Backlog

Name	Tags
Starting game environment	Priority
Training game environment	Priority
Battle system - Priority	Incomplete

Name	Tags
Cutscenes - Priority	Incomplete
JSON file implementation - Priority	Incomplete
Refined level design	Non priority
Refined NPCs	Non priority
Menu	Non priority
Pause system	Non priority
Save system	Non priority

Game Design Document

This next main section of the document covers the headers that makes up the Game Design Document. This includes the Project Vision, Background, User Stories and Use Case Scenarios, Games structure, Mechanics, Characters, Menu System, Level Desgign, Wireframes, Issues and Constrains and the final link to the GitHub repo.

Project Vision

In this project, I will be exploring leisure time and what might enrich this through developing a game. Developing this project will provide me with a wide range of knowledge and experience as I will be improving my game development skills further. During my A Level Computer Science 2019/2021 I was assigned a task to produce a game as my NEA project which gave me a foundation to game development using Unity. In this project, I will use my experience and this time I will be developing a game using HTML, CSS and JavaScript. This is an unfamiliar but new environment I will be touching upon but that is why undertaking this project is important to me.

My main vision for this project is to create an enjoyable and worthwhile game in order to encourage the user to spend their leisure time with good quality which will provide them a good experience which is what a good game should deliver. In order to achieve this, I have thoughtfully written out different ideas, including common and existing game concepts and adding a bit more to them or a gimmick which will make it unique. This is because developing a game of a concept that already exists may not be worthwhile to the user as they could just play another game of the same genre but better developed. By adding extra flair to it or combining two different concepts to make them unique, I am providing something quite new to the user.

Background

For the planning of this project, I decided to think of three different game concepts and decide which one I would like to develop to either learn a new area of game development or to simply improve my skills of one area/genre. I can already rule out the game being 3D as it would become extremely difficult for me to develop one especially in JavaScript as I am not too familiar with the language yet. So, I was left with making the game 2D. From this I then decide if I wanted the game to be a plain 2D where the user faces the game flat, top-down or isometric.

A plain 2D game would be the one of the easiest options as the whole canvas is set in a flat plane which means that putting assets on the canvas would not cause too many problems. I could then add further features in later development where such as a parallax effect making the game “pop” out more.

Super Mario World (Nintendo, 1990)

A top-down 2D game is the other easy option, as I my NEA project was based on this 2D perspective, so I have experience with developing in this style. This perspective is also popular within the 2D gaming genre, it features games such as the Zelda series, the Pokémon series, Stardew Valley and more.

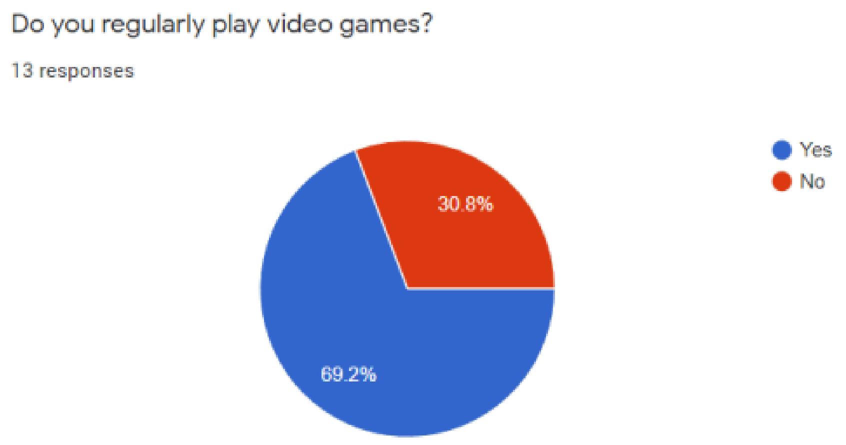
Pokémon Diamond and Pearl (GameFreak/The Pokémon Company, 2006)

A 2D isometric game is an interesting option as I think it’s hard to pull off a good game with this perspective. It was a revolutionary perspective when 3D games were still in a very early stage and the isometric plane gave a good balance between the two. However as of December 2018, Hades, a roguelike dungeon crawler is set as an isometric game became a very highly rated game on different platforms. It defines roguelike very well along with the fast-paced action it has which I think integrated very well on the isometric world.

Hades (Supergiant Games, 2018)

User Stories and associated Use Case Scenarios

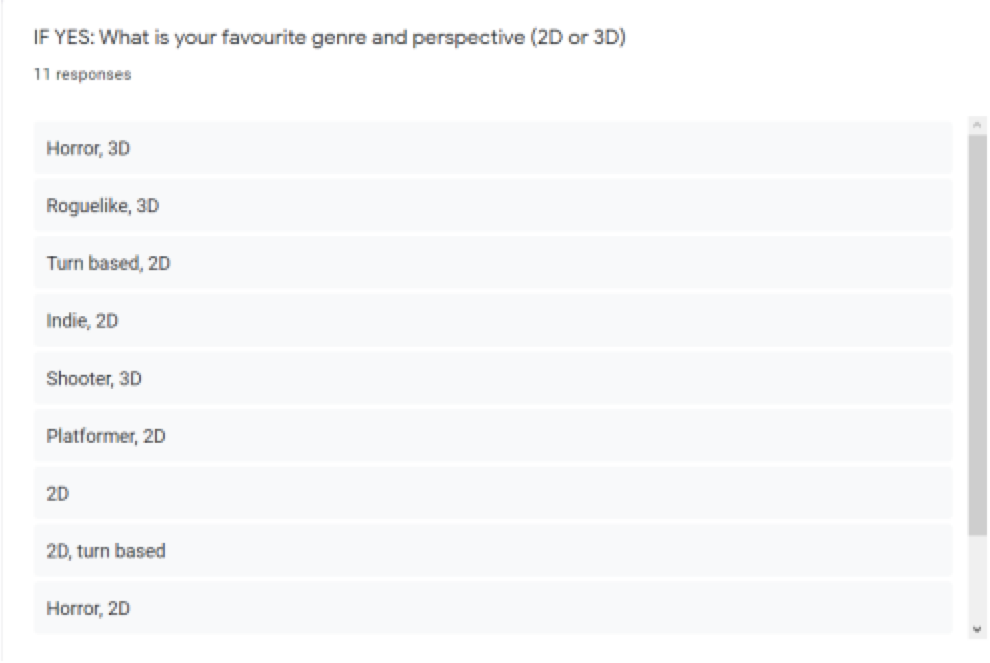
For this section this is where I gather data from other users who regularly play games but also to those who don’t regularly play games.
This split is important as there is a considerable experience gap between the two categories which could give me varying results which I do not want to mix up.
For the method of data gathering, I have chosen a simple questionnaire with a few questions where I will outline the results in a visual pie chart where appropriate.



Do you regularly play video games?

13 responses

This first question just gives me a rough idea of a percentage of people who regularly play games. I tried to make this fair by handing this questionnaire to not just those who are within a computer science subject background as it is quite likely for them to be playing games regularly. This itself is an assumption to the research.



IF YES: What is your favourite genre and perspective (2D or 3D)

11 responses

Horror, 3D

Roguelike, 3D

Turn based, 2D

Indie, 2D

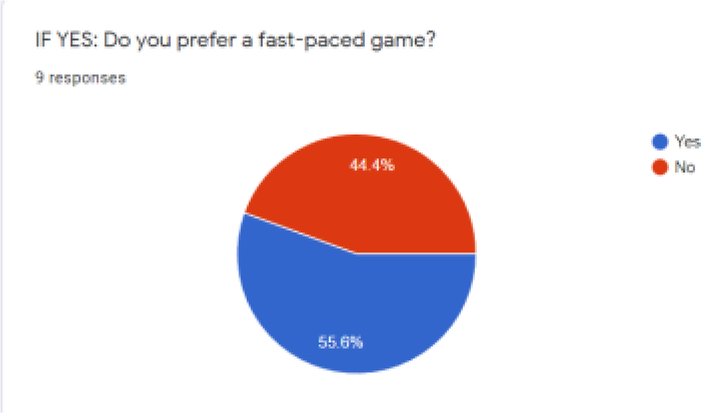
Shooter, 3D

Platformer, 2D

2D

2D, turn based

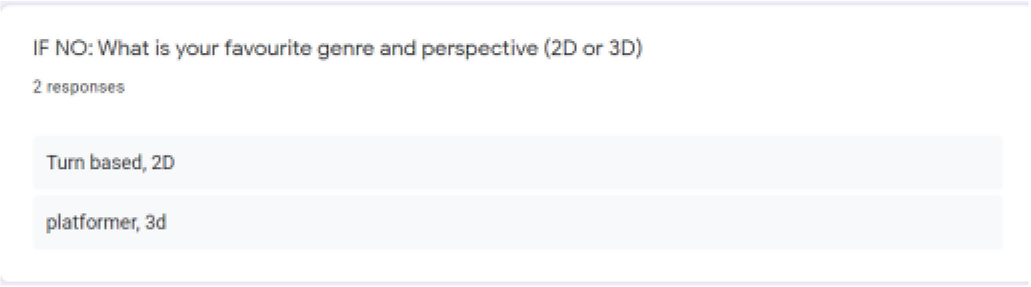
Horror, 2D



IF YES: Do you prefer a fast-paced game?

9 responses

For the people who play games regularly, I have a near 50/50 to the question of “do you prefer a fast-paced game?”. The reason why I considered this a question to be in the form is that it crosses out another big chunk of possible games I could make, however in this case I am still left with a 50/50 to either a fast-paced game or not.

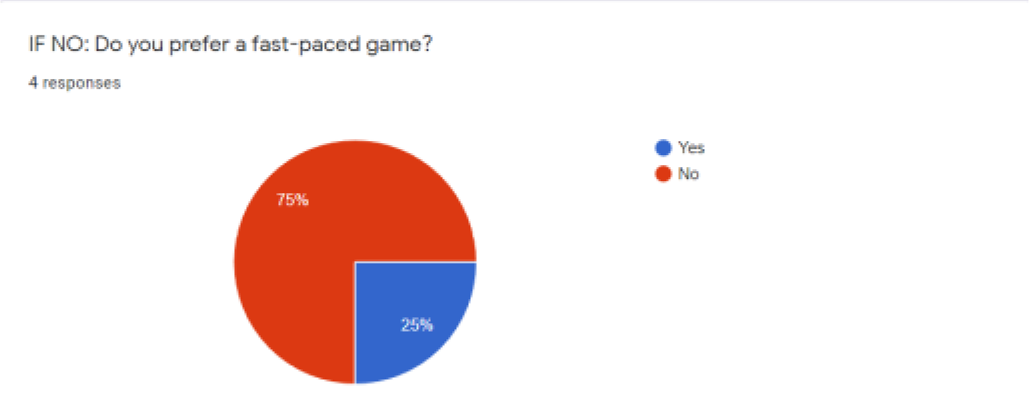


IF NO: What is your favourite genre and perspective (2D or 3D)

2 responses

Turn based, 2D

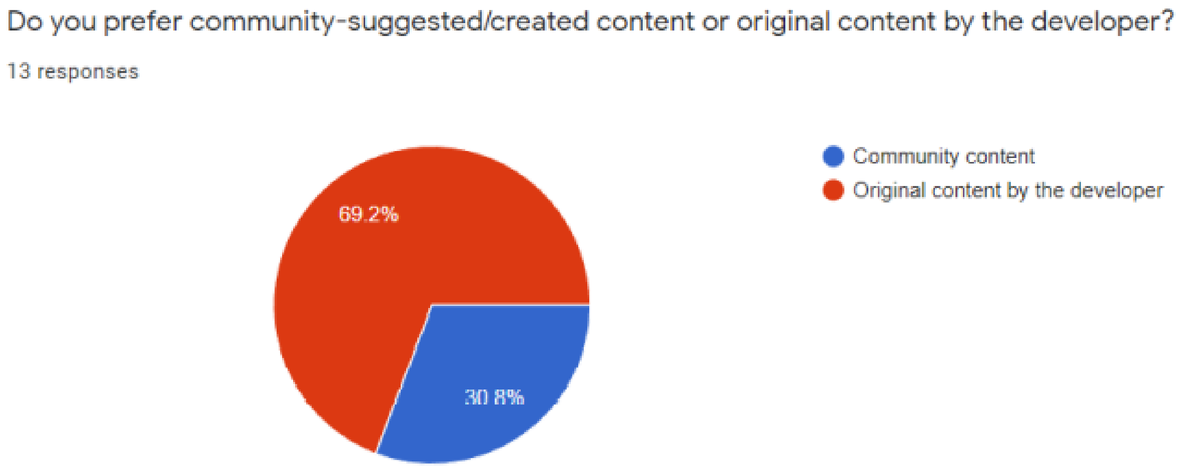
platformer, 3d



IF NO: Do you prefer a fast-paced game?

4 responses

For the people who don’t regularly play games, I seem to be missing a couple of responses on the genre and perspective of the game. The safe assumption here is that they are not too sure with what to enter in the form, but I do have a few responses for the fast-paced question where the majority said no. Which should give me an idea of the type of other games they could be playing.


Do you prefer community-suggested/created content or original content by the developer?

13 responses

This is a question I also considered back in my NEA project. I decided to ask this question in the form because having a community to give feedback on what they would like to see or changed in a game is integral to keeping a game’s lifespan up and keep users content with how the game is developed.

After evaluating these responses to my questionnaire, I have slimmed down my options to what game I could make. The chosen genre is a simple adventure game that has turn-based elements and will be set on a topdown perspective. On the base, the game is simple however a constraint I will encounter is making the combat of the game as I am unfamilliar with turn-based combat. However, to further challenge myself and the one element I have thought of to make this game unique, is to add elements of Dungeons and Dragons into the combat. This means that the combat stays turn-based, but there will be a lot of factors that can change the flow of the game such as abilities, stats, equipment and more. I chose this element to add as it fits well into an adventure RPG game and makes it more interesting.

Games Structure

This section will highlight the core game structure integral to the whole game itself. This will include the main mechanics of the game, the layout and other features.

The base structure of the game will be the overworld itself where it lays out the scene of the game the user character can walk and explore in. There will be multiple scenes in the overworld where it will take the user into a different room depending on the building/entry they go in. Different characters will also be placed around the world, and they could be passive NPCs (Non-player characters) that the user character can interact with.

Classification

One of the core features of the game to be covered is the combat. The combat will be turn-based set as a Dungeons and Dragons style. Meaning when the user initiates combat, they will be determining the majority of their actions through a throwing dice, the die involve is just a simple D6.

The next feature of the game will be the characters themselves. This includes the user character itself and NPCs. The NPCs are passive characters around the overworld that the user character can interact with. They could either guide the user character, provide dialogue or be an integral part to progressing the game and they could also be included in cutscenes (More about that later).

There is also another core feature that connects to almost every part of the game and that is the stats. This includes the HP of the user character, their abilities and stats which the same also applies to enemies. These change throughout the game as Dungeons and Dragons is dynamic in terms of the state of the characters playing

Responsibilities

Each of these features connect to form the base of the game, such as how the combat is widely affected by the enemies and the stats of each character on the game. The environment is responsible for setting up the scene for the user character to explore in and it will also spawn in the enemies and friendly NPCs at set positions. Items will also be spawned around the world that the user character can collect and use. Enemies can respawn on their relative position when they have been defeated and are not present, however items are a one-time spawn, but the user may be able to obtain them in other methods. During combat, the stats, user character and enemies come into play. The user character and enemies have a set HP and during each of their turns their stats could change until an outcome of the battle is reached.

Constraints

The main constraints I have with the game structure is learning how to implement them. Seeing as coding a game in JavaScript is completely new to me, it will take me considerably longer to create what I have planned than if I developed the game on an engine like Unity and using the C# language for the scripts. Another constraint is learning how to code scripts to work on particular characters. Unlike the Unity engine where you can simply make a script and attach them to an entity that can accept such script, the script can run for that entity without having any extra code. I may encounter more constraints not described in this section, however with the experience I have gained from making my NEA project previously I can look back on that and see how I can find a solution to any possible problems I will encounter. Another big constraint I have already come across is planning on how I will implement Dungeons and Dragons elements to the game especially in combat. A turn-based game can be simple or complex but adding Dungeons or Dragons into the mix means there are many factors that could be included to change how the game’s combat flows. This could make the game have a very interesting and unique combat; however, I can see that it will be very difficult for me as a solo student to implement.

Processing

The main processes the game will handle is the user character on the open overworld. The user can control their character through directional keys (WASD or the arrow keys), go through the base tutorial where they will then be sent out to the main overworld and progress throughy. The user can enter and exit different rooms and buildings.

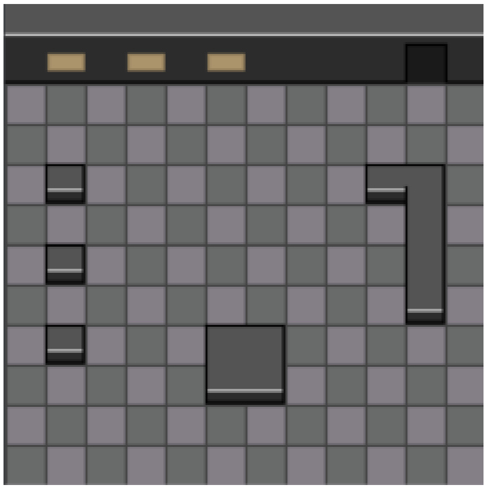
When the user initiates combat, normal Dungeon and Dragons rules will take place where the user rolls a D6 to determine the turn order then usual RPG combat takes place where each character in battle can use a move against the other character, use an item or a certain interaction; to which I will limit as it could be increasingly difficult to program multiple interactions the user character could do.

Menu System

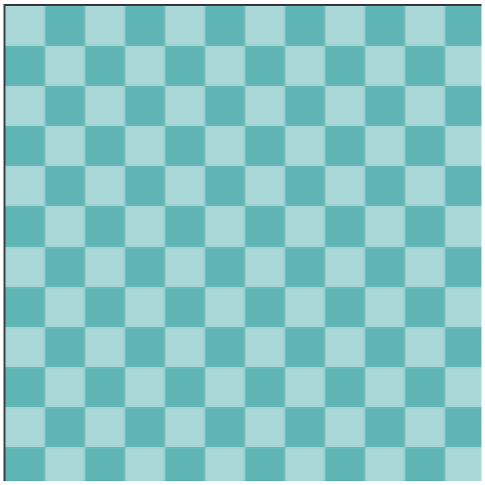
As the majority of the development will be focused into implementing more important features, the menu is most likely left out unless there is enough time to develop it properly. For now I have decided on leaving the user in a base world they can explore and roam around in and proceed to progress out.

Level Design

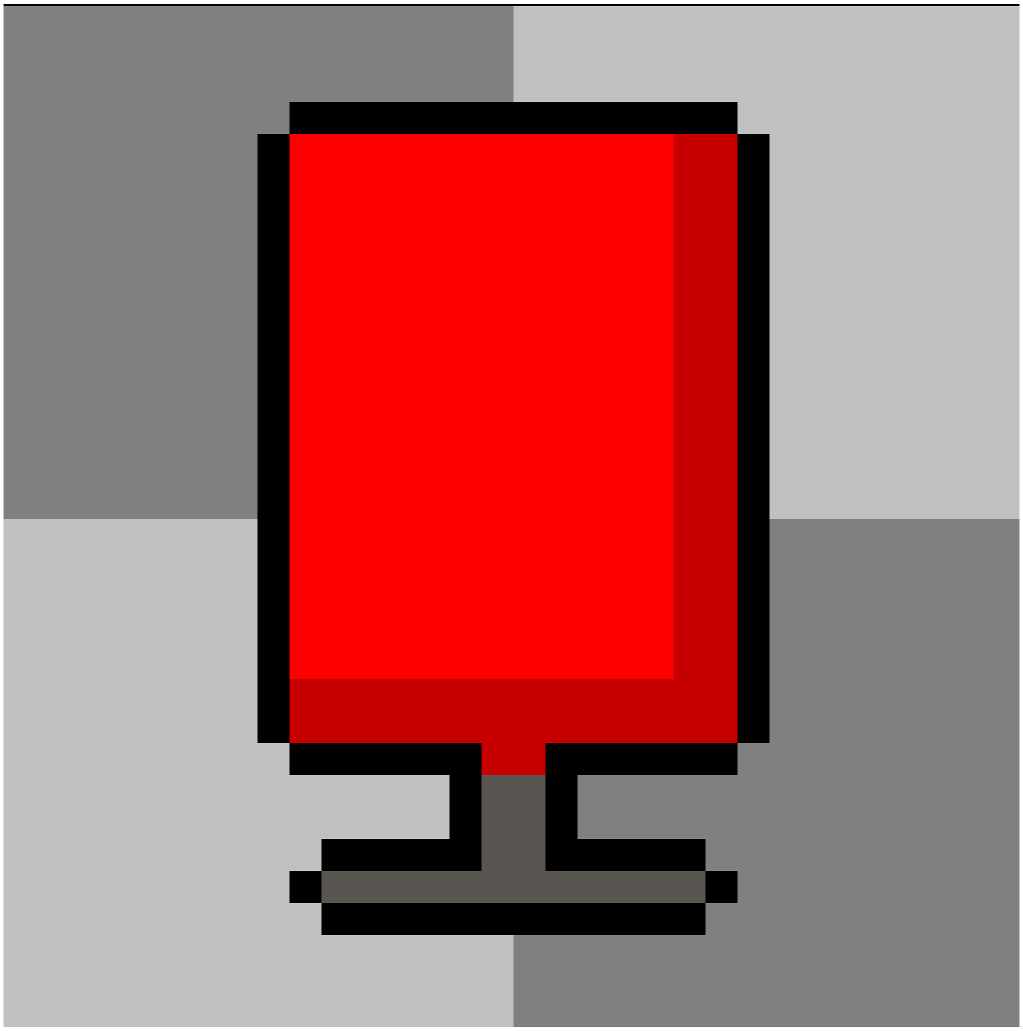
Currently the game will feature a base environment, a training environment and the battle environments along with the NPCs.



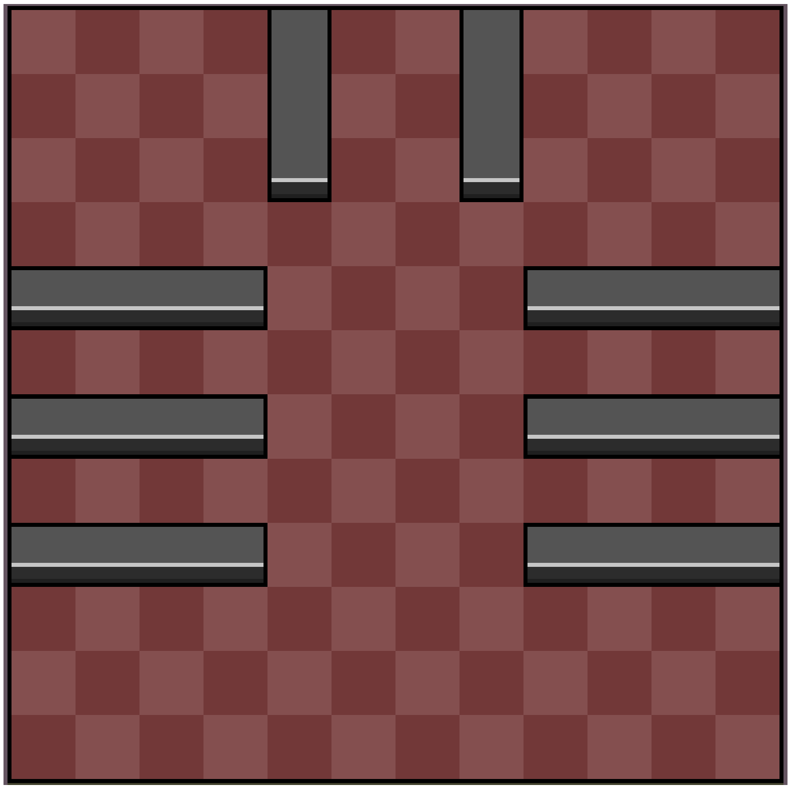
This is the base environment the player is thrown in.



This is the tutorial environment the player enters, note that it is plain as there are dummy NPCs here that can be interacted.



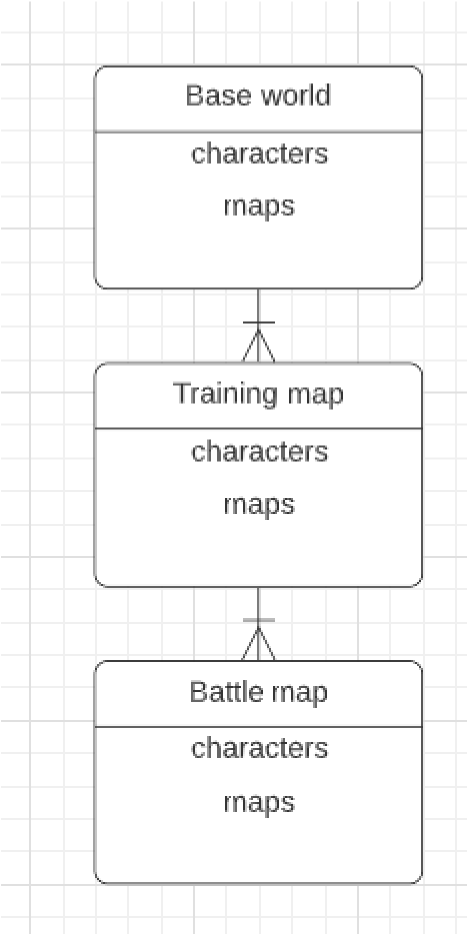
Which is just a punching bag.



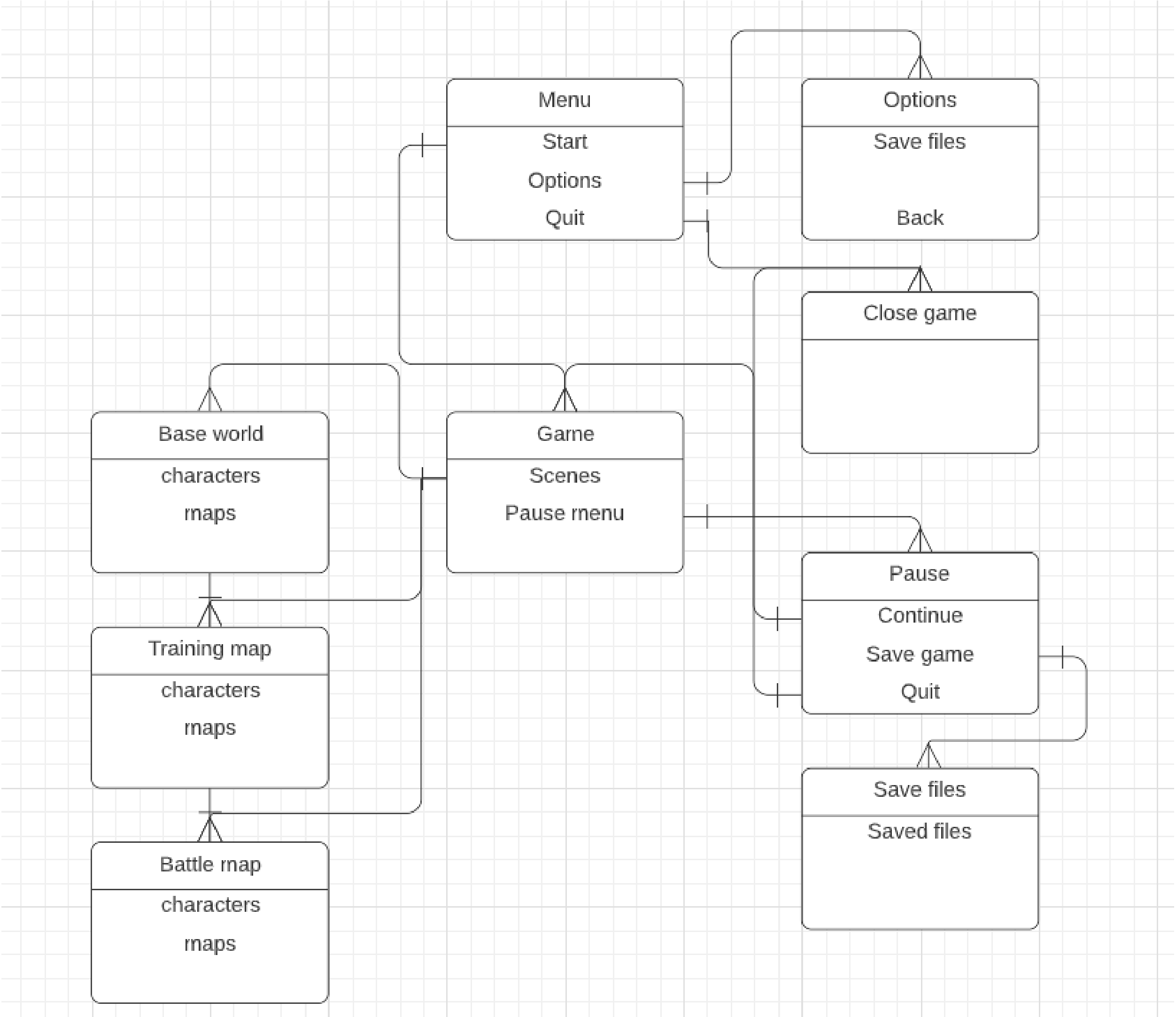
And this is the prepared battle environment, where there will be NPCs situated behind the walls and a final boss NPC at the back.

Wireframes

The wireframes for this final project will just show how the site currently works with all the connections between them.



This is what it should look like:



Noted Issues and Constraints

Throughout planning the development for this game, I have encountered
numerous constraints that may hinder my development methods and managing the time spent programming the game.

The main constraint I have is that I am programming a game in a language that I am unfamiliar with, meaning I must take up valuable
time to learn the language and try code with it. In the meantime, there
are various YouTube channels online that cover basic JavaScript

game programming with HTML and CSS which I can follow through and adapt for my own game.

Another constraint is that my experience of developing a game was using the Unity engine, where most things are convenient to code. This time, coding in HTML, CSS and JavaScript for this project has no engine to use as a baseline. Meaning I would have to hard code features like collisions manually, and sprite animations.

The biggest constraint I have in this project is definitely time, all of these constraints costs me valuable time to work out how I would develop this game. On top of that, the game is currently in a very unfinished state due to a fatal bug tied to the cutscene system which is heavily relied on. As such, NPC interactions and battle scenes are not implemented. Because of time, I have also failed to implement the JSON script needed.

GitHub Repo

```
https://github.com/ZekKen243/comp1004---main-assignment-ZekKen243
```

Reflection

I definitely had challenges with undertaking this project. The biggest take away with the project for me is that I learned the basics of setting up a HTML site with CSS and JavaScript files and utilising them to connect to each other and create a base of a game. I overestimated myself with how well I could manage the

project, especially with other trivial things in the way such as family, work and mental issues that got in the way.
From all this I learned a great deal of experience with another form of game development with JavaScript and seeing how much work manually implementing features we take for granted now that we see conveniently in game engines such as Unity. An example of this was sprite slicing, where in JavaScript I had to manually slice the sprites myself in the code whilst on an engine like Unity I would just put the spritesheet in the files and edit from there into the game scene simply.

I would have definitely gotten further if I managed my time well along with other courseworks in the way using the sprints better, and I would have otherwise seen the fatal bug sooner and found a fix for it with enough time to implement other important parts of the code.

I also couldn’t have done it without a great deal of research and resources to find to help me. Drew Conley’s YouTube channel especially helped me as the project he worked on his channel was very close to what I also wanted to make, so I mostly just followed his tutorials and learned the as he explained what parts of the code and scripts do. His YouTube link is in the Project Plan document at the bottom to credit him.

Overall I would know now how to properly plan a feasible project scope with the allocated time given, and how to especially follow a proper development method in order to achieve proper progress and know to manage time well, which is my responsibility.

Bibliography

1. Nintendo, 1990. *Super Mario World*. [image].
2. GameFreak, 2006. *Pokémon Diamond*. [image].
3. Supergiant Games, 2018. *Hades*. [image].

Special thanks to Drew Conley for providing valuable learning resources with game development on JavaScript, HTML and CSS.

https://www.youtube.com/channel/UCvQwAK9oAYXM0RMucLy2-BA