Dungeon of Death

Fadi Gorges

Table of Contents

Statement of Intent

I intend to create an entertaining fantasy video game titled “Dungeon of Death,” in which the player explores unique underground dungeons, overcomes enemies and collects treasure along the way. Dungeon of Death is set during medieval times, where monsters are prominent throughout the world, and the player must make use of weapons to defeat them. My decision to build this video game is based on my own love for playing similar games growing up; ones that defined my childhood. They sparked my passion for gaming and ultimately game development, as I hope to bring others the same enjoyment from my project as they've brought me. I aim to design and code my video game Dungeon of Death with the purpose of entertaining players and possibly taking game development as a career in the future.

My project will be completed over the course of a year in multiple stages such as research, planning and development. In order to create and advertise a playable video game, I will need to research and build many technical skills including graphic and sound design, coding, animation and video editing. These skills are difficult and time consuming to refine, but I am confident in my ability to develop and hone them. I will study similar games in the genre, and with my own ideas, plan and develop a video game with a story and art style that best matches the concept of Dungeon of Death. I will also create a video trailer to promote my game and convey to potential players what they can expect. To do this, I require a drawing tablet and a fast computer with a high storage capacity, capable of running intensive software such as Unity, Rider, Adobe Photoshop and Adobe Premiere Pro. Using industry level programs will also ensure that I can produce a quality project.

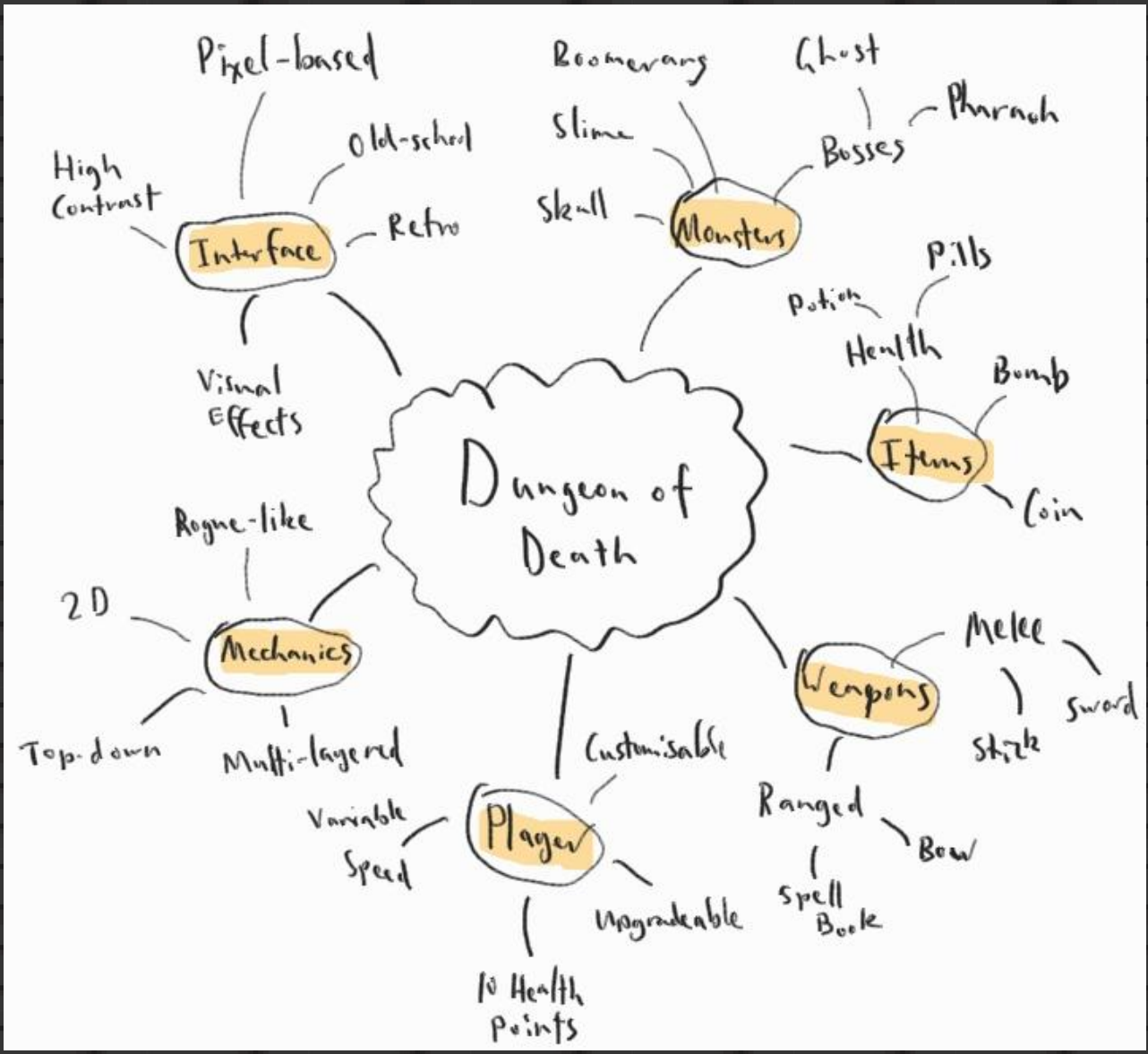
The general target audience of my video game is anyone who enjoys challenging skill-based gaming. The main audience; however are teenagers, specifically teenage boys, as they are most likely to be interested in the style and gameplay Dungeon of Death provides. Players of this game will receive an entertaining and enjoyable way to spend time and compete for high scores against their friends.

My final project will be composed of parts that are presented in their own individual ways. The Dungeon of Death video game will be available as an executable file, which can be run and played interactively on Windows PCs, but can also be adapted to run on almost any operating system. The game trailer will be presented as soon as the player begins the video game, right before the action. I plan to utilise this Major Project in the future as part of my portfolio and build my own freelancing brand.

There may be a few factors that will limit my progress and project development. My current hard drive does not have enough storage space to store the project files required for my video game, such as images, code and sound effects. To overcome this problem, I will invest in a larger hard drive with greatly increased storage space. Software such as Unity and Rider that I need for my project may require purchasing, which can also halt my progress, but I plan to acquire a student license, as I am still in school, to overcome this. Another limitation I will encounter is the learning curve associated with mastering relevant and appropriate coding languages such as C# and Python, but my prior knowledge of the fundamentals of coding and my determination will help me enhance my knowledge and skills in this area, while also allowing me to meet my intentions.

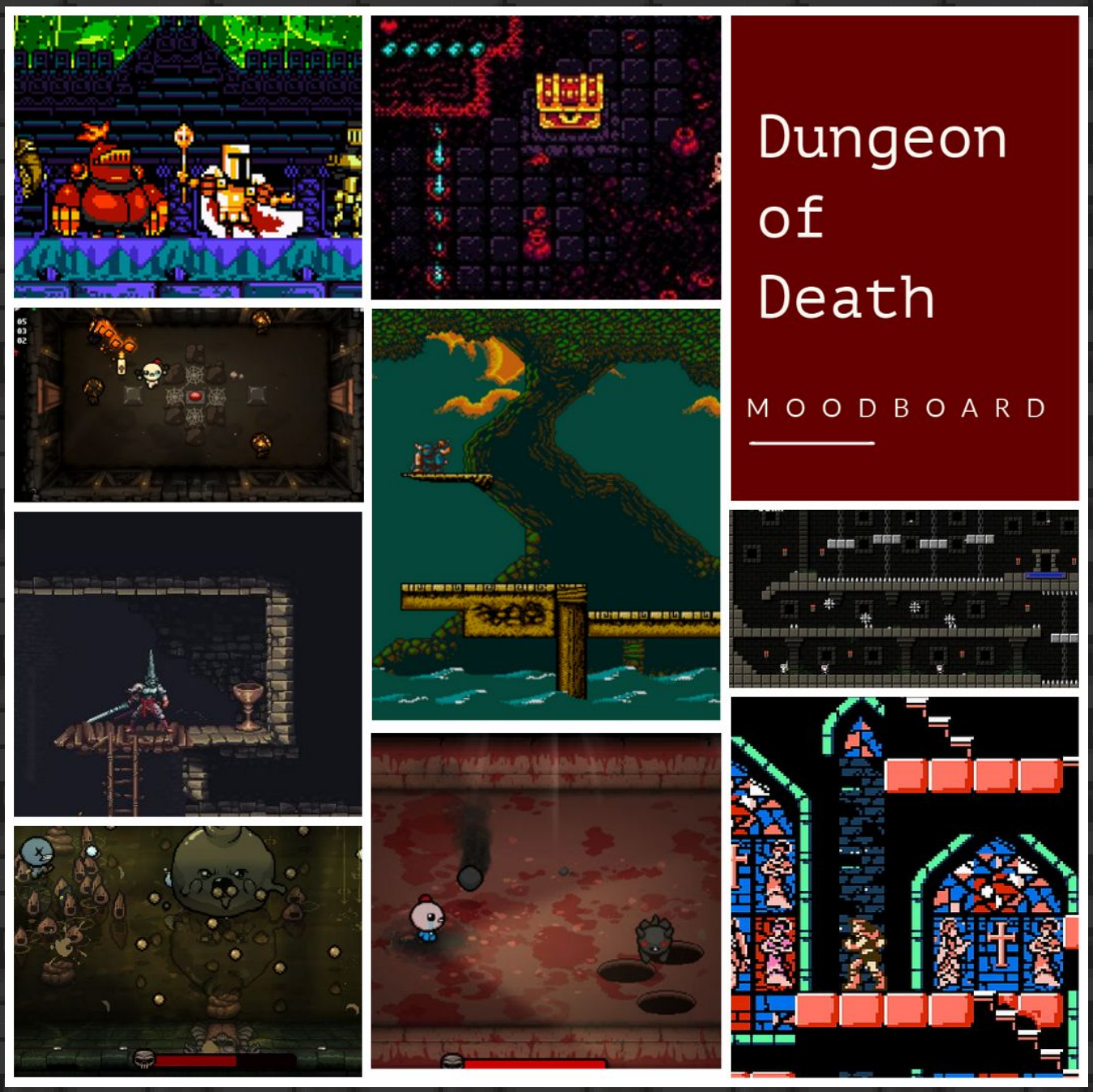
Idea Generation

Based on inspiration and my own ideas, I have a somewhat clear vision of the characters, weapons and items that will be incorporated in my video game. I also have a general idea of how the story will play out, and how gameplay will be integrated to match the pacing of the story.



Idea Generation

Moodboard



Idea Generation

User Interface

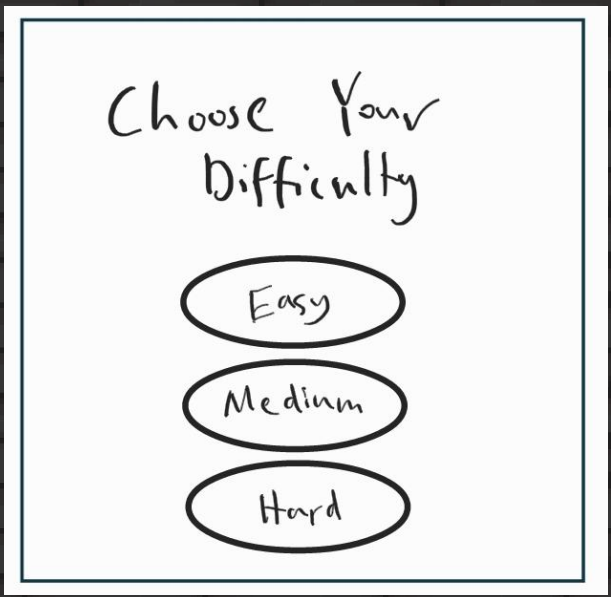
The user interface (UI) is an extremely important aspect of a video game that determines how the player will interact with elements such as menus and buttons. A badly designed user interface can make a game extremely confusing to learn and unpleasant to play. It can also render a game unappealing, with terribly placed buttons and text, making it an eyesore.

I chose to make my UI very simple and minimalistic, cutting down the amount of buttons and text that aren't necessary. The reason why I chose to make a simple menu design was because it's the first thing that the player encounters when playing the game. If there are too many options, the player will get overwhelmed, but if the design is too simple, the player will gain a bad first impression.

This was the first concept of a main menu that I created, consisting of a large title and three buttons: Play, Settings and Exit. These are the most basic buttons required in the main menu for game functionality and customisation. It would work well and be simple enough to understand without overwhelming the player from the get-go.

Idea Generation

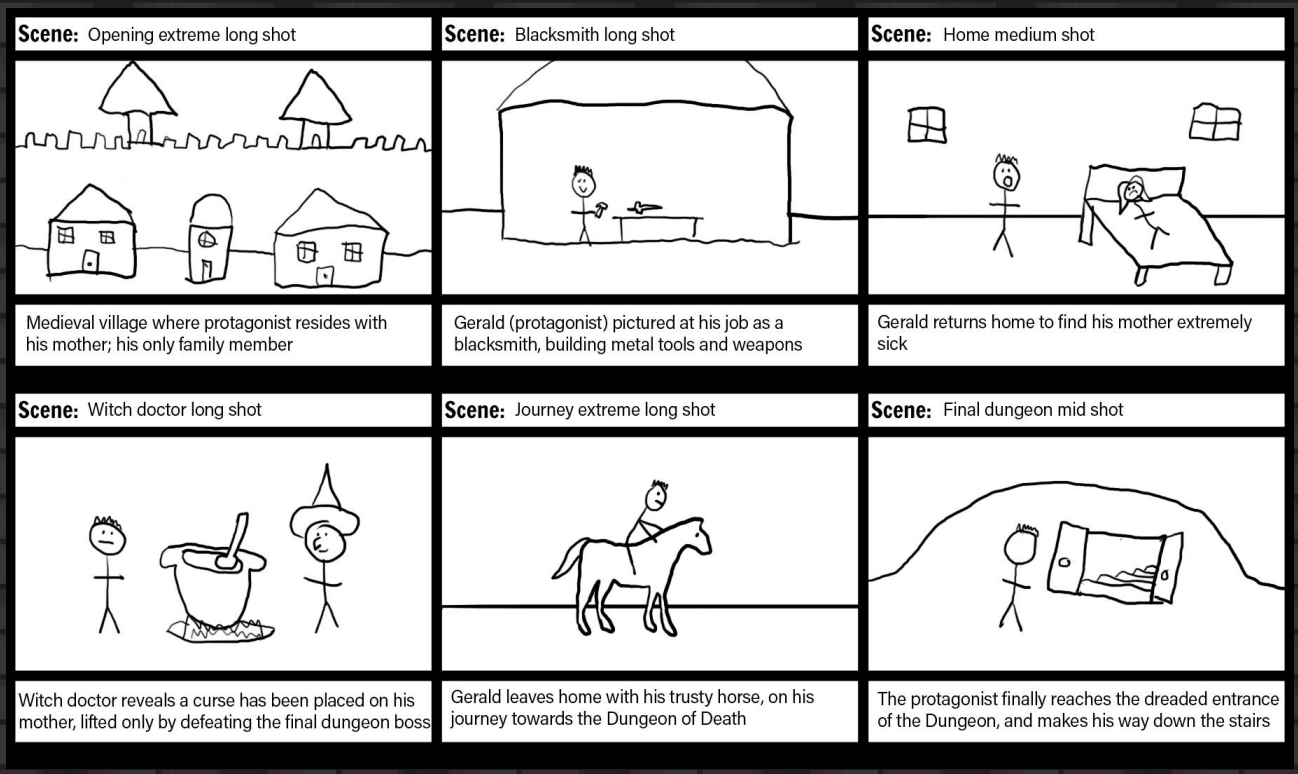
However, I decided to redesign this main menu as a static title with static buttons looks too simple and unappealing to the player. This time, I took into consideration other aspects of interface design such as fonts, images and especially animation. I decided to make a black sliding animation that plays as soon as the game is launched, revealing the title of the game alongside the logo. This title then shrinks and moves slightly upwards, allowing room for a textbox instructing the player on how to start the game. This main menu concept was much more appealing to me as it incorporated animation, shaders and different styles of text which grab the players' attention much more easily, while adding to the retro arcade vibe. I scrapped the Options and Exit buttons as they were unnecessary in regards to gameplay, and replaced them with a single keypress that is needed to begin the game.



Next, I sketched two more screens that would be displayed before the player begins playing the game. Firstly, a difficulty selection screen would be presented with choices consisting of "Easy, Medium and Hard." These options give the player a choice on their level of gameplay and the skill required to beat the game. Customisation like this can greatly improve players' experience as some tend to be more skilled than others, and building skill often requires playing at an easier difficulty to begin with. The final screen displayed before the game begins will be a menu explaining all the controls of the game such as how to move and attack, so that the player can easily learn how to play the game. It only has to be very simple, so I outlined the game controls in text alongside their corresponding mouse/keyboard controls.

Idea Generation

Opening Animation



I decided to create this sketch of a short animation that plays when the user clicks on the "Play" button. This clip will be drawn and animated inside Adobe Animate. I'm not a very good animator or proficient in Adobe Animate, but I am determined to learn the ins and outs of the process.

Initially, I wasn't going to create or include any long animations like this because I thought it wouldn't turn out very well. I also doubted that I could learn Animate, in conjunction with all the other programs, in the limited time frame I have. However, I like a challenge, and this animation would not only be a great learning experience but also beneficial to the structure of my video game. It would be fun to watch, while also providing background information on the life of the protagonist, Gerald, and the reason why he decided to venture into the depths of the dungeon. Right after the animation ends, the player is spawned in and the gameplay begins.

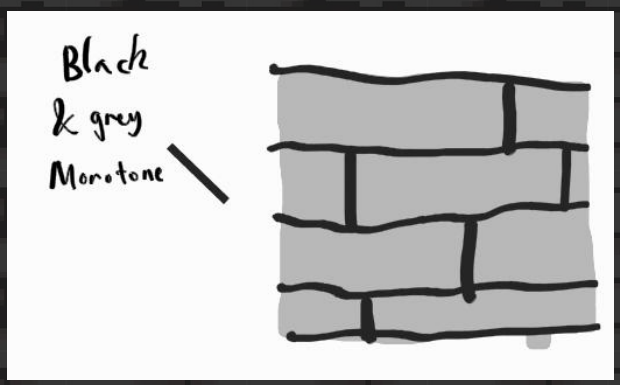
In the final version of my game, I did not add this animation as I could not get it to look as good as I wanted. I anticipated that this idea of a long animation would be challenging, and although I tried my hardest to create it, the result did not turn out as well as I expected. Instead, I decided to leave this part out of my game and focus on creating an amazing game trailer and improve the game experience even further.

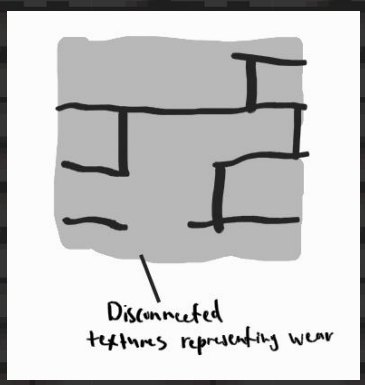
Idea Generation

Texture Design

The main colour scheme of my video game is made up of a variety of dark contrasting shades and warm hues. Examples of recurring colours I will use throughout my art style are blacks, whites and greys, along with colours such as red, orange and yellow to represent torches or fire.

As my video game, Dungeon of Death, is based on old-school arcade classics such as Super Mario Bros, I wanted to create the same vibe through a retro art style that brings back nostalgia among players. To design the levels throughout the dungeons of my project, I required several textures for walls, floors and UI elements.

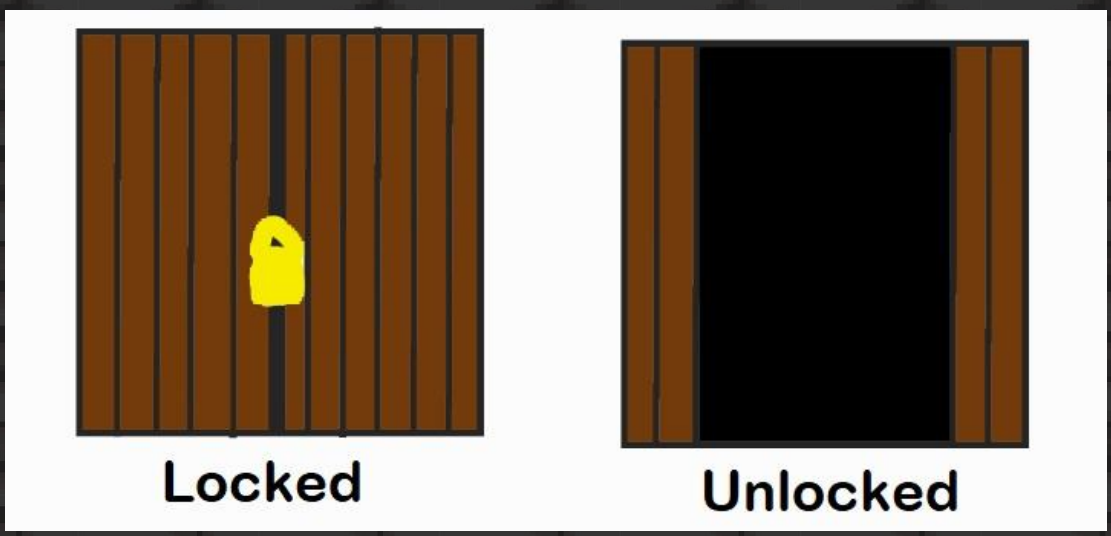




I created these simple sketches of the first dungeon layer in my video game. The surrounding walls of the levels will be designed as a monotone cobblestone brick texture with contrasting shades of black and light grey. The textures will be created in a program such as Photoshop or Piskel by drawing individual pixels on a small template of 16x16 or 32x32 resolution. Although they look plain in these sketches, I plan to add post processing and illumination in-game through the use of components in the game engine Unity.

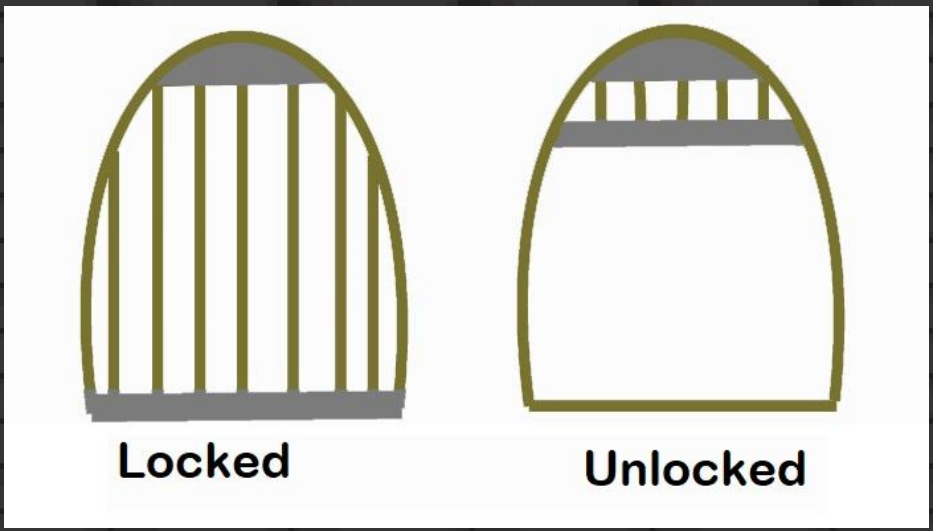
Idea Generation

For the second layer of dungeons, I sketched an underground sandstone type of texture. This follows the progression of the game's story as the player explores dungeons deeper and deeper underground, defeating monsters and claiming dungeon rooms. The textures resemble light and dark cracked desert sandstone, with the walls being a lighter colour than the floor. Also, the tiling of the floor pattern is more dense and compact than the walls, which is a style choice that I personally love.



The doors that the player travels through will have 2 phases that animate once the player survives the level; a locked and unlocked phase. These doors will be used for the first layer of dungeons, and when the player reaches the 2nd layer of sandstone, the doors become gates:

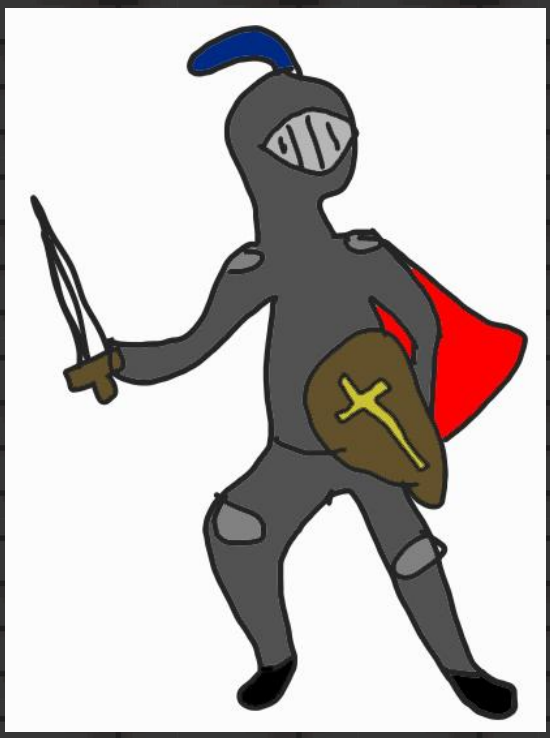
Idea Generation



Player

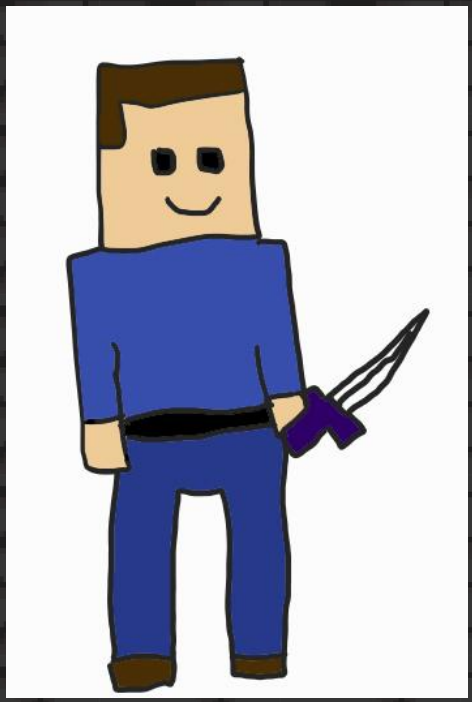
In my mind, the main protagonist of my video game is just a simple man who ventures down a dungeon in the hopes of lifting a curse placed on his mother. He's not a knight or a trained fighter, just a generic unrecognisable blacksmith who works in a village. This idea of a player character is relatable to the player of the video game, as they have never played or witnessed the dungeons before either. It adds another subconscious experience to the gameplay, instead of playing a knight figure as in most other dungeon crawler type games. Although I chose this route, I still experimented with sketches of a knight.

Pros:
• Good design, nice colours • Follows same character design of many dungeon crawler characters
Cons:
• The design is too complex for a low-res retro 8-bit art style. With the limitation of only a 32x32 player image size, I could not fit a sufficiently detailed and animated pixel art representation of this knight into my video game. • Doesn't fit my storyline of a blacksmith and the curse



Idea Generation

Pros:
• Fits into the storyline of Dungeon of Death • Simple player model that can easily be translated into pixel art form through programs such as Photoshop or Piskel
Cons:
• Very simple design



Although the knight design seems like the more appealing choice, it would require my storyline to change and it wouldn't bring the added benefits of having the player relate to the main protagonist. Also, the design was simply too complex for an 8-bit retro style video game. To combat the simplicity of the chosen blacksmith player model, I decided that I would add in player armour and accessories that can be picked up during the play through of the video game.

The idea of the player armour and accessories wasn't added into the final version of the game as I experienced quite a few difficulties in the process of creating it. The armour would not track the player's body correctly, and misplaced every time the player moved on the screen. Also, as the player animation rotates when changing direction, I was required to draw 4 separate animations for each piece of armour and accessory.

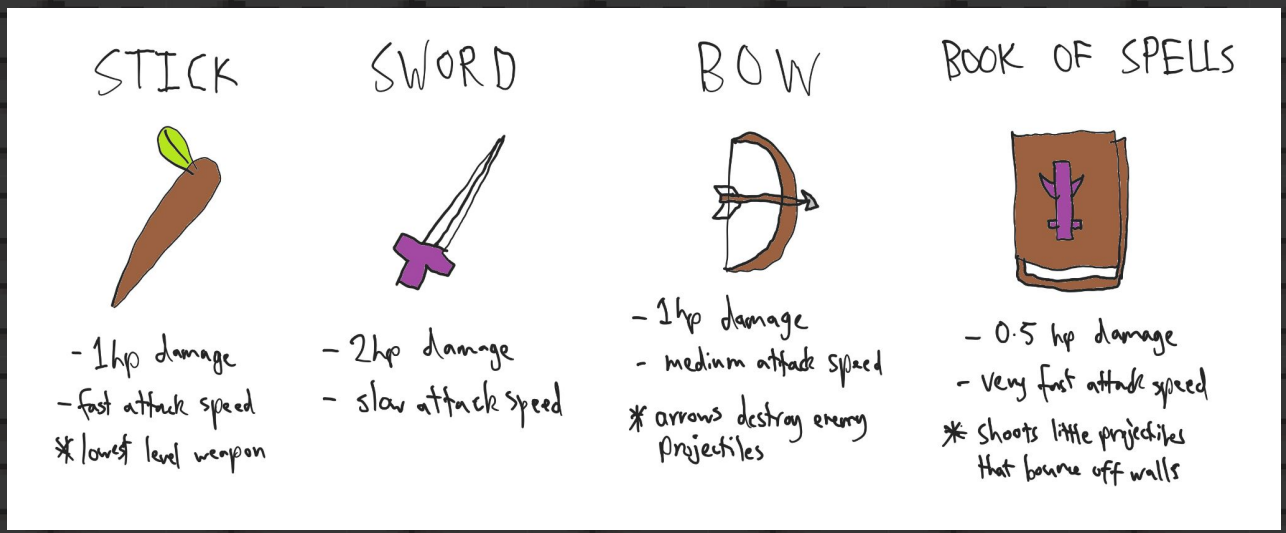
Idea Generation

Monsters

Dungeon of Death will have these basic monsters. They will do anything they can to end the player's life and stop them from reaching the final Boss. Each monster has their own unique properties such as health, speed and damage dealt. Some can also slow the player down and fly through obstacles. These are listed in the sketch.

I designed the enemies as monsters because it makes the game more unique, adding to the fantasy and story behind the player exploring the dungeons. It's very similar to the old retro games that I grew up with as well, such as PacMan and Super Mario Bros, with their ghosts and fantastical enemies.

Weapons

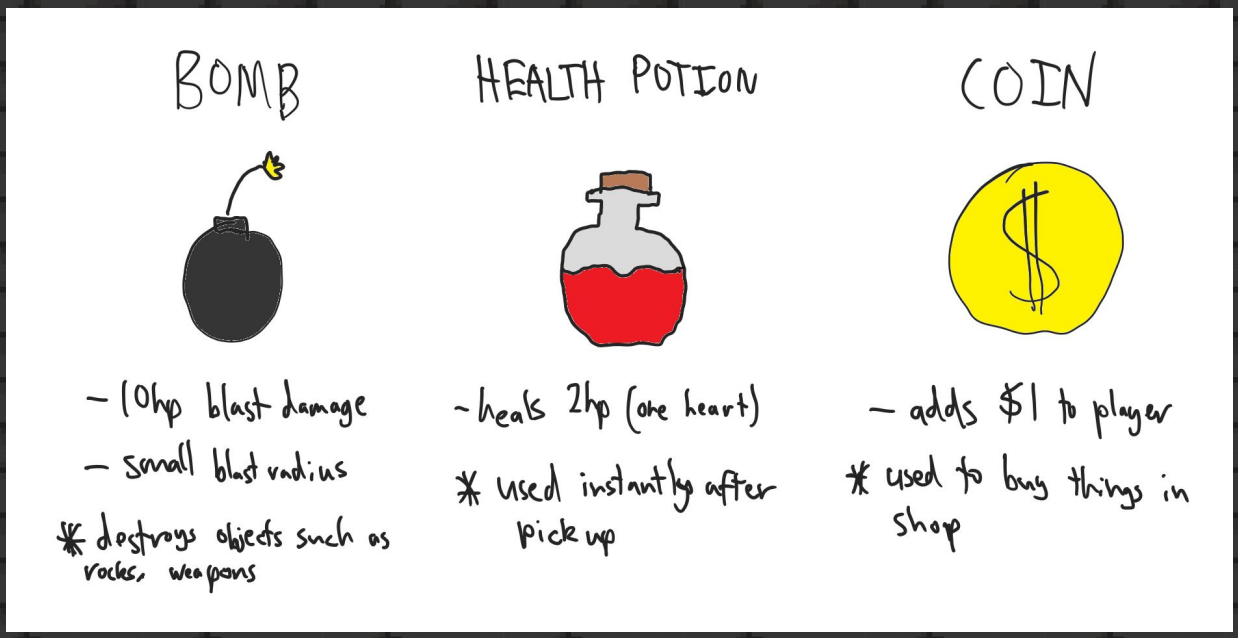


Idea Generation

I came up with 4 main weapons so far: the Bow, Stick, Sword and Book of Spells. Each weapon has its own properties such as damage, cooldown and range that determine how good it is at defeating enemies. They also utilise different methods of dealing damage. The Sword and Stick deal melee damage while the Bow and Book of Spells fire projectiles.

These weapon ideas, apart from the Book of Spells, should be very familiar to any person playing Dungeon of Death as they are very common objects, especially in other video games. Most people know that sticks and swords are used as melee weapons, and that bows are used to shoot arrows. This familiarity is why I chose to add these weapons, as both experienced and non-experienced gamers would be able to correctly identify and utilise them. The Book of Spells is not so obvious however, but as it is found later in the game, it can be picked up and experimented with by players as they gain experience throughout the game.

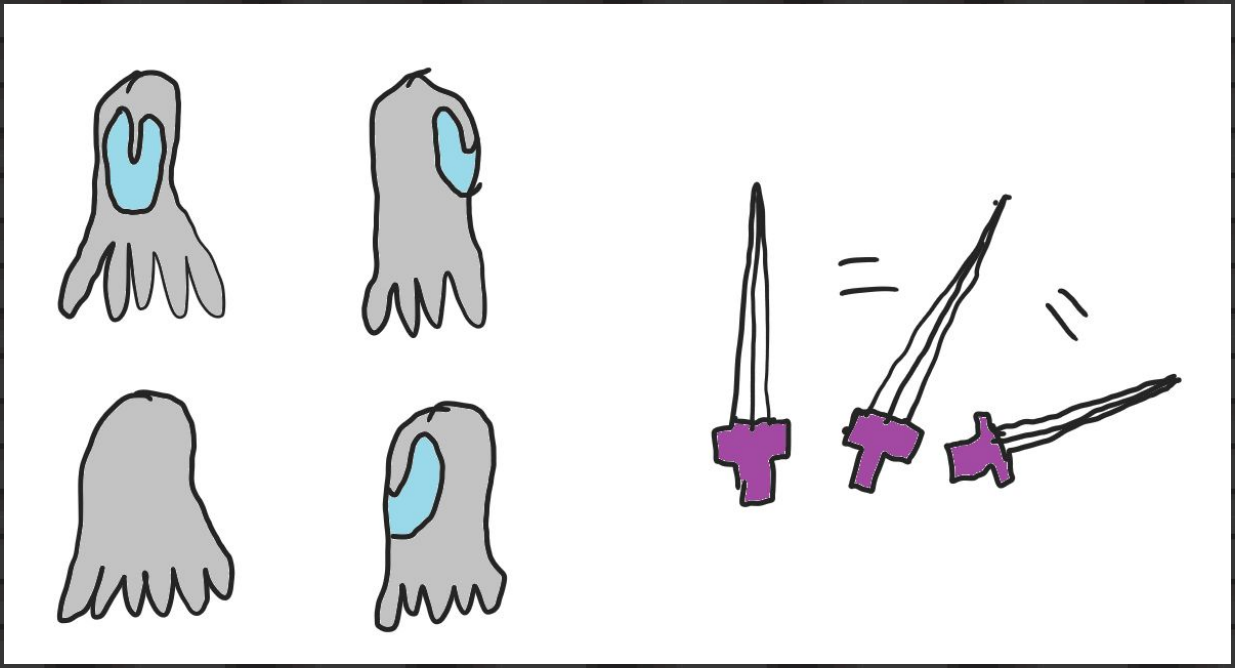
Items



The items/pickups that I introduced are Bombs, Health Potions and Coins. Originally, I was going spawn weapons and items, but then I had an idea of creating a shop after the boss is defeated. So instead, I decided to make items randomly generate inside rare chest placements in certain rooms, as well as be purchasable in shops. Bombs differ from regular weapons as they can only be used a limited amount of times (3 to begin with). The player can pick up Bombs and use them as they wish until they run out. When a bomb detonates, it deals damage to any creature in its small radius. The Health Potion is pretty self explanatory; it heals the player, providing 2 hearts instantly. The Coin is simple too, used as a currency to trade for special items and weapons inside Shops.

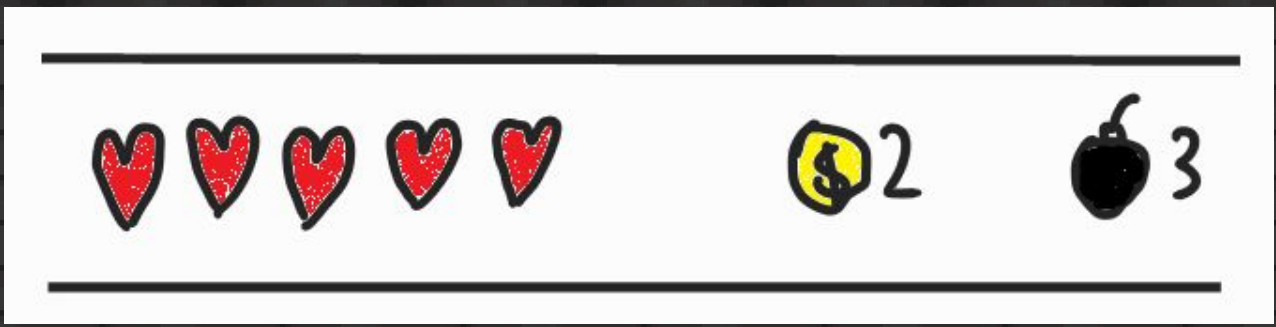
Idea Generation

Sprite Animation



Every mob such as the player and monster has 4 unique animations for moving up, down, left and right throughout each level. My thoughts behind this was that it'll provide greater visual feedback on where the monster is heading, as well as be more visually and aesthetically pleasing, compared to one single animation. Weapons also have their own animations when fired, such as the sword pictured above.

Info Bar

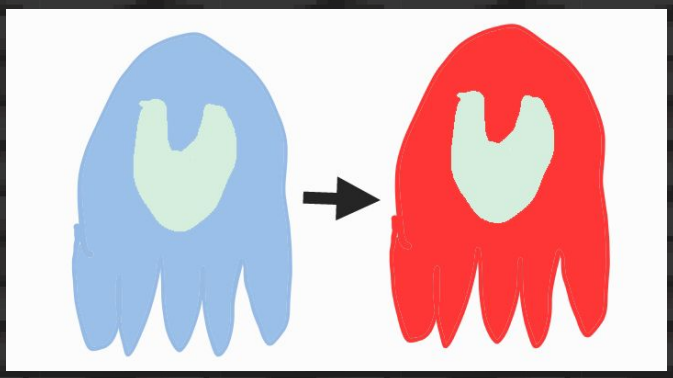


The health, bomb and coin system are managed at the top of the game window. Players have 10 health points, with each heart of the 5 hearts representing 2 health points. These hearts can be either full, half full or empty, all depending on the player's health. I used a 5 heart system instead of 10 full hearts to make it easier to fit onto the UI and easier for the player to keep track of their health. The bomb and coin counter is also displayed at the top of the window.

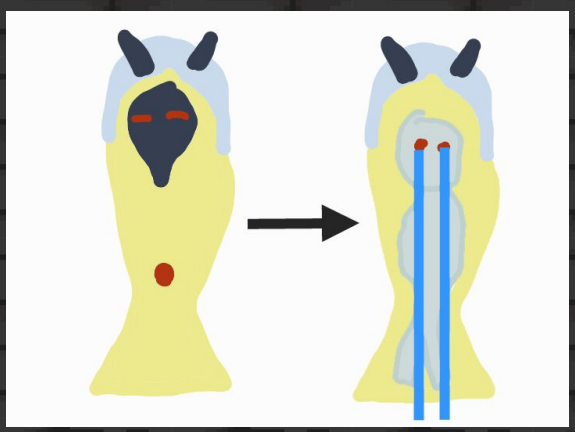
Idea Generation

Bosses

For the two layers of my dungeon, brick and sandstone, I decided to create two individual boss fights after the first 5 ordinary rooms have been defeated. The Bosses present a real challenge to the player, greatly testing the players' skill and determining whether or not the player is good enough to pass. I decided to make the boss of the brick layer the Gelare Boss. It's basically the mother of all Gelares; supersized, unique dashing pattern and extremely high health points. It has two phases, a Passive and Attack phase. While Passive, the Gelare Boss is coloured blue and can be attacked and does not harm the player. However, while in the Attack phase, it turns red and dashes towards the player, dealing great damage while also being invulnerable.



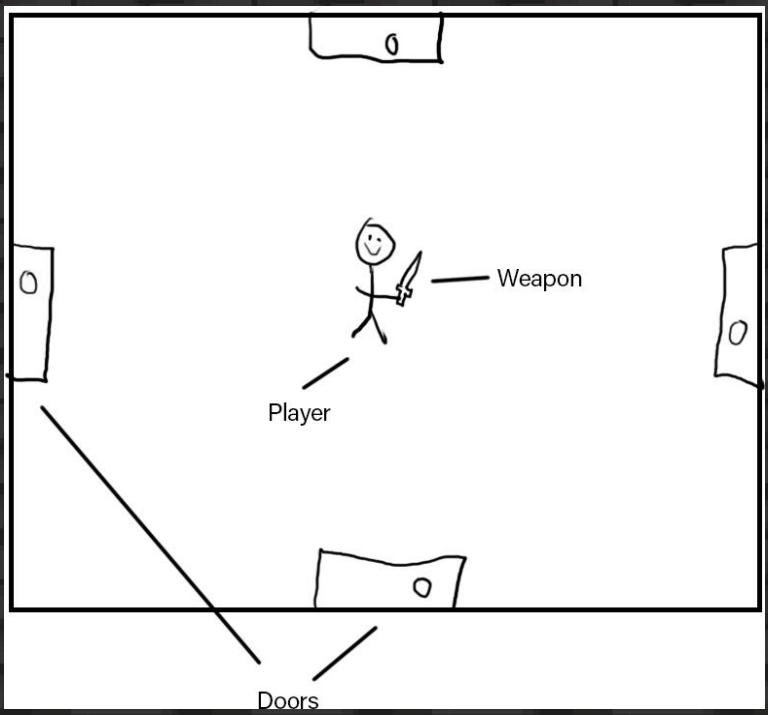
The Sarco is the second and final Boss that the player has to defeat before winning the game. The Sarco is modelled after a Sarcophagus which is a stone coffin associated with ancient Egyptian civilisations. It has 3 phases; Passive, Spawn and Laser. The Passive phase is similar to the Gelare Boss, as the Sarco remains stationary in the centre of the level while able to be damaged by the player. During the Laser phase, the Boss opens up and reveals a human figure inside, that fires a laser downwards as it travels left to right across the level. Finally, the Spawn phase enables the Sarco to open up and teleport to all corners of the dungeon, summoning common Enemies in the hopes of stopping the Player.



Idea Generation

Level Design

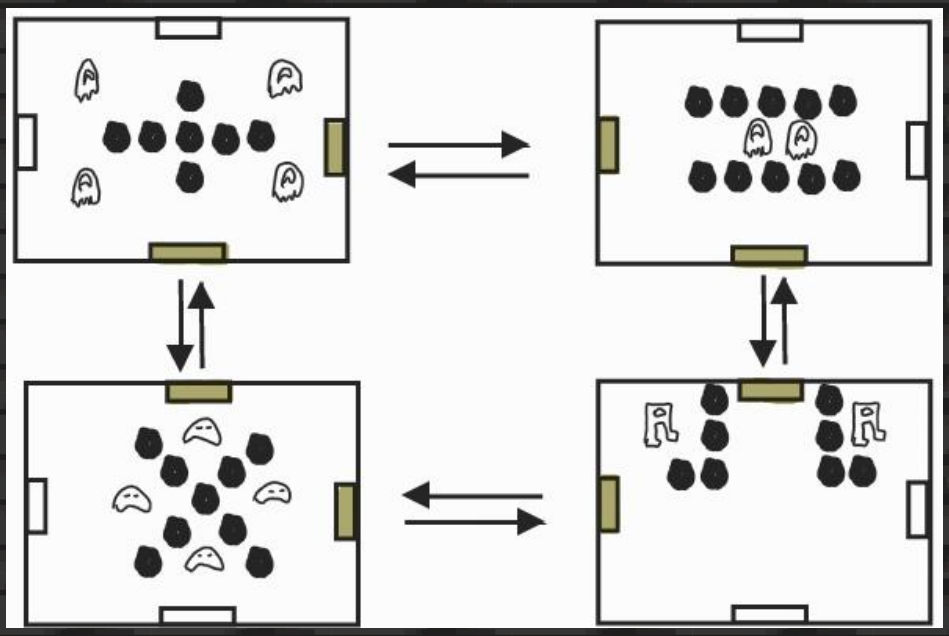
The most basic level design is a rectangular room with up to 4 doors that lead to other rooms throughout the dungeon. These rooms are saved and players can backtrack into these rooms in order to pick up items and weapons if they choose to. I originally planned to keep these doors unlocked the entire game, but realised players would just sprint straight towards the boss, without any challenge. Instead, I decided that rooms would stay locked until every monster is defeated, forcing the player to participate, while also challenging their gaming skills.



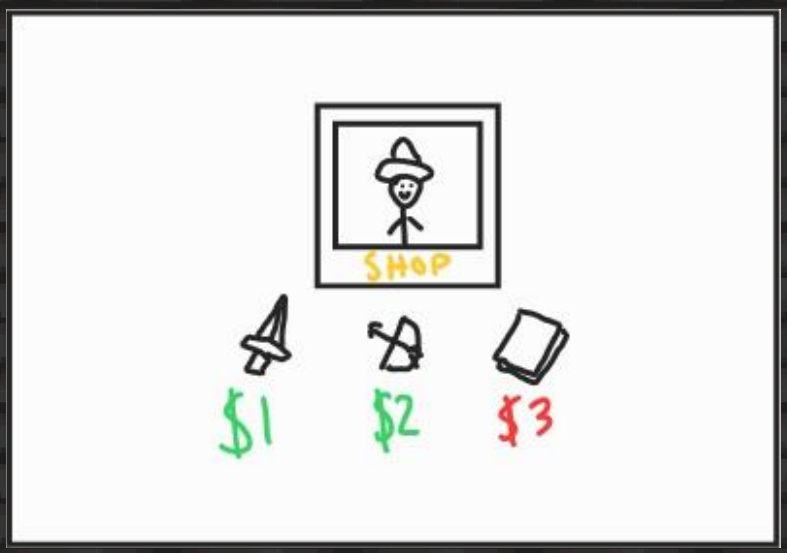
The first room of the game has 4 unlocked doors, allowing the player to choose any direction they wish. They must first pick up the lowest level item of the game, the Stick, to aid the player in defeating enemies.

The doors will remain locked until the player defeats all the monsters in the level. I chose to have the doors locked at the beginning so that players cannot speedrun through the levels without fighting any enemies, defeating the entire purpose of the game. It keeps the game balanced and fun to play.

Idea Generation



The dungeon layout will be procedurally generated from a series of premade level designs. Players will explore each level facing a new level layout, different sets of enemies and using different weapons. Rooms will be memorised in the game system and players will be able to backtrack through to them in order to gather old loot and switch weapons.

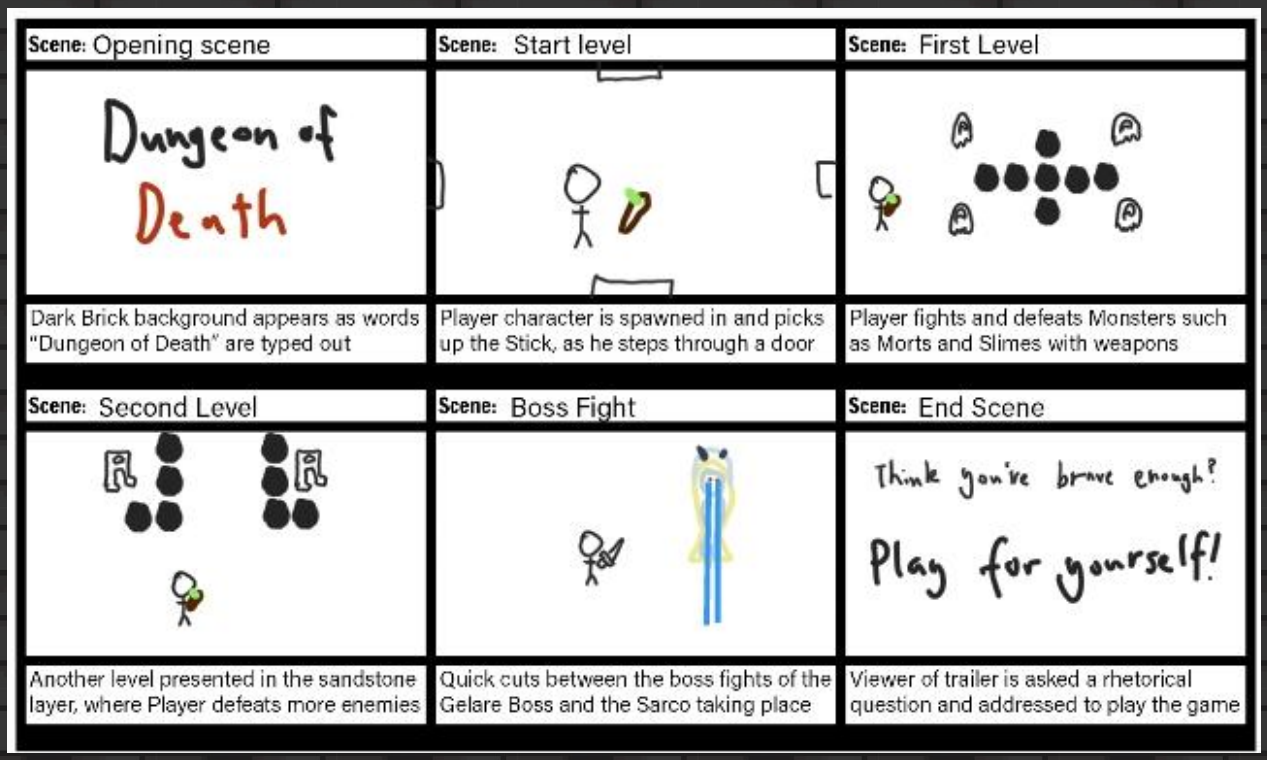


After all levels, including the Boss Level, are defeated in the first layer of the dungeon, the player will stumble upon a shop to purchase items before they make their descent towards the 2nd sandstone layer. Players will be able to spend coins they gather throughout their playthrough to purchase health, weapons or bombs to aid them in survival and conquering the dungeon.

Idea Generation

Video Trailer

The second element of my Major Project is the video game trailer for Dungeon of Death. The purpose of the trailer is to give the player/viewer insight on the gameplay they can expect, as well as hooking their interest and making them more likely to try the game out. My idea of a game trailer is a cinematic, movie-like experience with tense music and fast paced gameplay clips. It will outline the basic mechanics of the game as well as the enemies, items and weapons the player can expect to meet.



This storyboard idea is intriguing and attention grabbing, portraying many aspects of the game without revealing every part of the game, so as to not leave the viewer with nothing to gain from playing. The video ends by addressing the viewer and leaving them with the final thoughts, “Think you’re brave enough? Play for yourself!” These lines almost dare the viewer into playing the game themselves, greatly increasing interest and the likelihood of the viewer engaging.

Research, Selection & Justification

Inspirations and Motivations

TITLE
The Binding of Isaac

CATEGORY
Video game

LINK
https://store.steampowered.com/app/113200/The_Binding_of_Isaac/
Date: T4W2 2020

Implications and justifications:

The Binding of Isaac was one of my favourite games to play growing up. I chose this video game as an inspiration because I love the conceptual aspects such as the story, gameplay and the similar dungeon-crawler-esque style that I had in mind for my own video game.

The 2D, almost 8-bit art style will most likely be incorporated into my own video game, to provide players with the old 90's and 2000's video game feel, along with my own visual effects and graphics. I will also utilise the general pacing of the rooms and dungeons, with my own original characters, weapons and monsters.

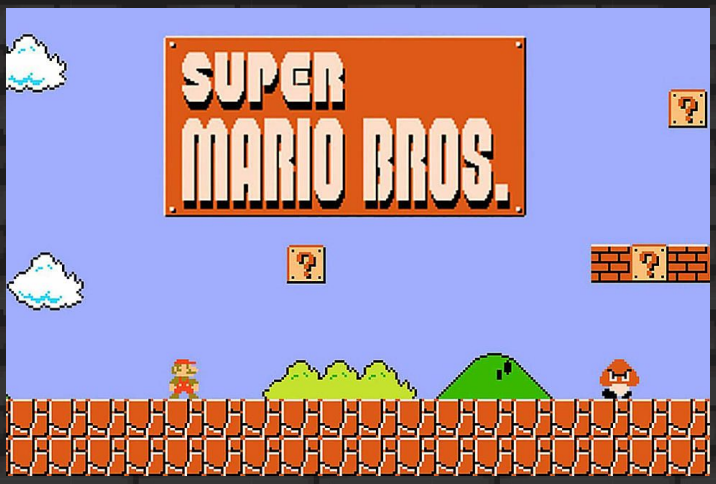
Research, Selection & Justification

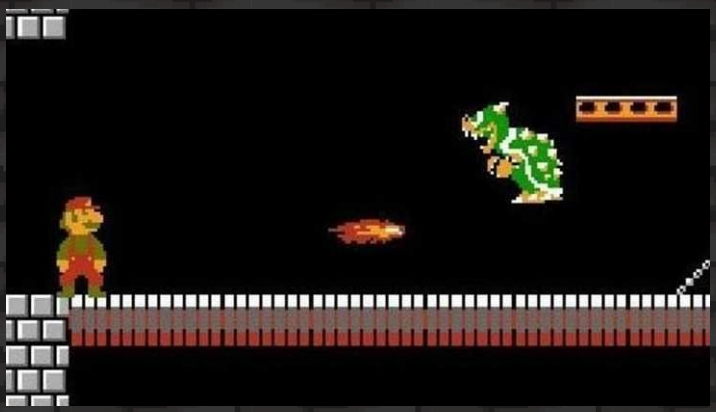
Inspirations and Motivations

TITLE
Super Mario Bros

CATEGORY
Video game

LINK
https://mario.nintendo.com/





Implications and justifications:

Super Mario Bros was one of the first video games I've ever played. It takes me back to fond memories at a younger age, totally disconnected from reality and immersed in its timeless 8-bit visuals.

My video game will be heavily inspired by the simplistic 2D 8-bit visuals that vary greatly from modern, hyper realistic graphics that make it difficult to differentiate gaming from reality. This is perfect, as I aim to provide players with the sense of old, nostalgic gameplay.

Research, Selection & Justification

Inspirations and Motivations

TITLE
Undertale

CATEGORY
Video game

LINK
https://undertale.com/

Implications and justifications:

Undertale is a newer game released in 2015, but the way it utilises technical and conceptual design features to create an immersive, nostalgic sense of gameplay makes it a huge inspiration for me.

The old fashioned art style and unique story hints at early 90's video games, which is exactly what I aim to achieve with my game. The beautiful music and soundtrack of Undertale will also significantly influence the music I produce and incorporate into my project.

These video games greatly defined my childhood and gave me a great sense of nostalgia. They made me fall in love with gaming as a whole and I know for sure that it provided millions of others with the same enjoyment and experience that I had when I was a kid. They got me into game design, in the hopes that I can bring back this enjoyment and nostalgia for others.

Research, Selection & Justification

Technologies - Hardware

Hardware	Description	What does the tech offer?
Home Desktop 	Prebuilt personal computer from MSY Technology. Specifications: • Intel Core i5-9400F 2.90GHz • Asus PRIME H310M-K Motherboard • Kingston 8GB DDR4 2400 • Asus Nvidia 6GB GTX 1660 Ti Dual Graphics Card • Kingston 128gb SSD and 1tb Hard Drive	This computer is extremely good at providing a powerful home workstation to design and develop my Major Project. The mid to high-end components of this desktop make it fast and efficient in tasks necessary for completing my Major Project, such as graphic design, video editing, game development and coding.
Monitor 	ViewSonic 27” VX2757 Monitor Specifications: • Low input lag • 75Hz refresh rate • AMD FreeSync™ technology • Black stabilisation • Full HD 1080p resolution	This ViewSonic monitor is used in conjunction with my home desktop to provide a smooth, lag-free user experience. The large 27” display makes for great multitasking and enables me to operate software at their full potential, while still maintaining a sharp, colourful image. This will help to ensure the quality of my project.
Mouse & Keyboard Bundle 	Razer Gaming DeathAdder 2000 Mouse Razer Cynosa Pro Keyboard	This Razer Mouse and Keyboard bundle offers great practical and ergonomic design for long periods of use. The low input lag, sturdy design and backlit keys of the mouse and keyboard are perfect for everyday use and are designed to last through years of general wear and tear.

Research, Selection & Justification

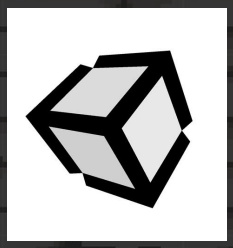
Hardware	Description	What does the tech offer?
School iMac 	iMac provided by the school to work on my project in class Specifications: • 2.7GHz quad-core Intel Core i5 processor • 8GB (two 4GB) of 1600MHz DDR3 memory • 1TB (5400-rpm) hard drive • NVIDIA GeForce GT 750M graphics processor • 21.5-inch (diagonal) LED-backlit display	The school iMac is a decent all-in-one desktop capable of running necessary software for my Major Project while I am at school. The iMac is several years outdated however, and tends to lag while running intensive tasks such as developing and executing code, although it is good enough for my at-school needs.
Mobile Phone 	Samsung Galaxy Note 9 Specifications: • 6.4-inch quad HD OLED display • Snapdragon 845 processor with 6 / 8GB RAM • 128 / 512GB storage with microSD support • 4,000mAh battery with wired and wireless fast charging • Dual 12-megapixel rear camera with OIS and switchable aperture • 8-megapixel front camera with autofocus	The Samsung Galaxy Note 9 is a fast, large-sized phone that's perfect for researching and jotting down notes on-the-go. It allows me to watch tutorials and look for inspiration during transit, car rides and in my free time. The Note 9 also comes with an amazing pressure sensitive S-Pen that is perfect for drawing and note taking, so that I can jot ideas down to refer to at later stages.
Drawing Tablet 	Wacom Intuos Small Drawing tablet Specifications: 4,096 levels of pressure sensitivity 4 customisable ExpressKeys Active area of 152 x 95 mm.	The Wacom Intuos Small Drawing Tablet is my go-to for amazing precision artistry, anything from sketches to final designs. It offers a great, lag-free connection and amazing support with Adobe owned products such as Photoshop

Research, Selection & Justification

Technologies - Software

Software	Description	How will it be used?
Adobe CC Photoshop 	Adobe Photoshop is a raster (pixel based) photo editing/manipulation software that enables users to design and create aesthetically pleasing graphics and art. Users can create art ranging from modern logos and graphic design to complex drawings and paintings, and just about anything you can think of.	I will use Adobe Photoshop throughout my Major Project to design the 8-bit sprites of my video game, such as the player, monsters, items and dungeons. Sprite animations will be created in Photoshop too, making use of its animation timeline. I will also use Photoshop to create the game's user interface elements such as buttons, health indicators and menus.
JUSTIFICATION I chose to use **Adobe Photoshop** over other similar photo editing programs such as **Paint.NET** and **Affinity Photo** because of the complex features it has to offer and my level of familiarity with the program. Photoshop is an industry leading program that outdoes competitors by a long margin. The timeline feature of the program is also extremely useful in my pixel art animations as I do not need to constantly switch to another program such as Adobe Animate to create my video game art. This greatly improves my efficiency and helps me work more productively on my project. Also, Photoshop is provided free to access through my educational license.		
Adobe CC Premiere Pro 	Adobe Premiere Pro is a video editing software that allows users to cut and order video clips on a timeline, adding transitions, music and sound effects. Premiere Pro can be used to stitch together anything from short video clips to full-length movies, colour correcting every shot and adding atmospheric sound to the final film.	I will use Adobe Premiere Pro to cut, trim and edit together an informational video on my game. The video will showcase the controls, features and gameplay in a thorough but easy to understand manner. It also promotes and showcases the intricacies of the game, garnering interest while also enticing experienced players wanting, not only a nostalgic feel, but a challenge as well.
JUSTIFICATION **Adobe Premiere Pro** was my video editing suite of choice over **Sony Vegas Pro** and **Final Cut Pro** as it is by far the easiest for me to access. Sony Vegas requires me to pay a hefty sum of money for a video editing program, while Final Cut is only able to be run on Macintosh operating systems. As my home desktop computer runs Windows and I can obtain Premiere Pro through a school license, it became my application of choice.		

Research, Selection & Justification

Software	Description	How will it be used?
Unity 	Unity is a cross-platform game engine that is used to build 2D and 3D games from scratch, that are able to be run on a majority of operating systems such as Mac, Windows and Android. Unity provides support for simple 2D and 3D game formats, as well as the Unity Render Pipeline and High Definition Render Pipeline which can create amazing, hyper realistic graphics in video games.	My video game will be designed and developed inside the Unity game engine. All the models, animations, components and scripts will be stitched together to create a fun, playable game. The game will also be exported through unity on either the Mac, Windows or Android platforms, becoming fully playable without the Unity editor.
JUSTIFICATION I chose to use the **Unity** game engine over others such as **Unreal Engine 4** and **Godot** as the premium version is free through an educational license, and it is extremely easy to use. Unlike Unreal Engine, Unity has an extremely well set-up interface for 2D game design, aiding efficient and productive game design. Also, the code that Unity relies on is the programming language C# which I have had previous experience with and will be able to continue to hone through my Major Project.		
Jetbrains Rider 	Jetbrains Rider is a professional cross-platform code editor that supports languages such as C#, HTML, CSS and Javascript. Rider provides real time code analysis with thousands of code inspections and automated quick fixes. Its refactoring, navigation and search tools make it extremely useful to keep track of large projects and script files. Rider also fully integrates with the Unity game engine, making game development and debugging extremely intuitive.	I will use Jetbrains Rider to code all the logic and back end that makes the game function. Everything from movement to weapon functionality to health systems will be programmed inside Rider's integrated development environment. I will also debug any errors or game-breaking bugs that arise from coding errors, all inside Rider's built in debugger and breakpoint system.
JUSTIFICATION Jetbrains Rider might not be the most popular code editor available on the market, but it has by far the most features and plugins available that make coding a breeze. Unlike Visual Studio, Rider supports the Unity game engine directly, adding tons of debugging information and shortcuts that cut the time taken to code in half. It's certainly a market leader and a game changer in the field of code editing software. The founding company Jetbrains also provides an educational license, making it readily available to use for my Major Project.		

Planning & Progress

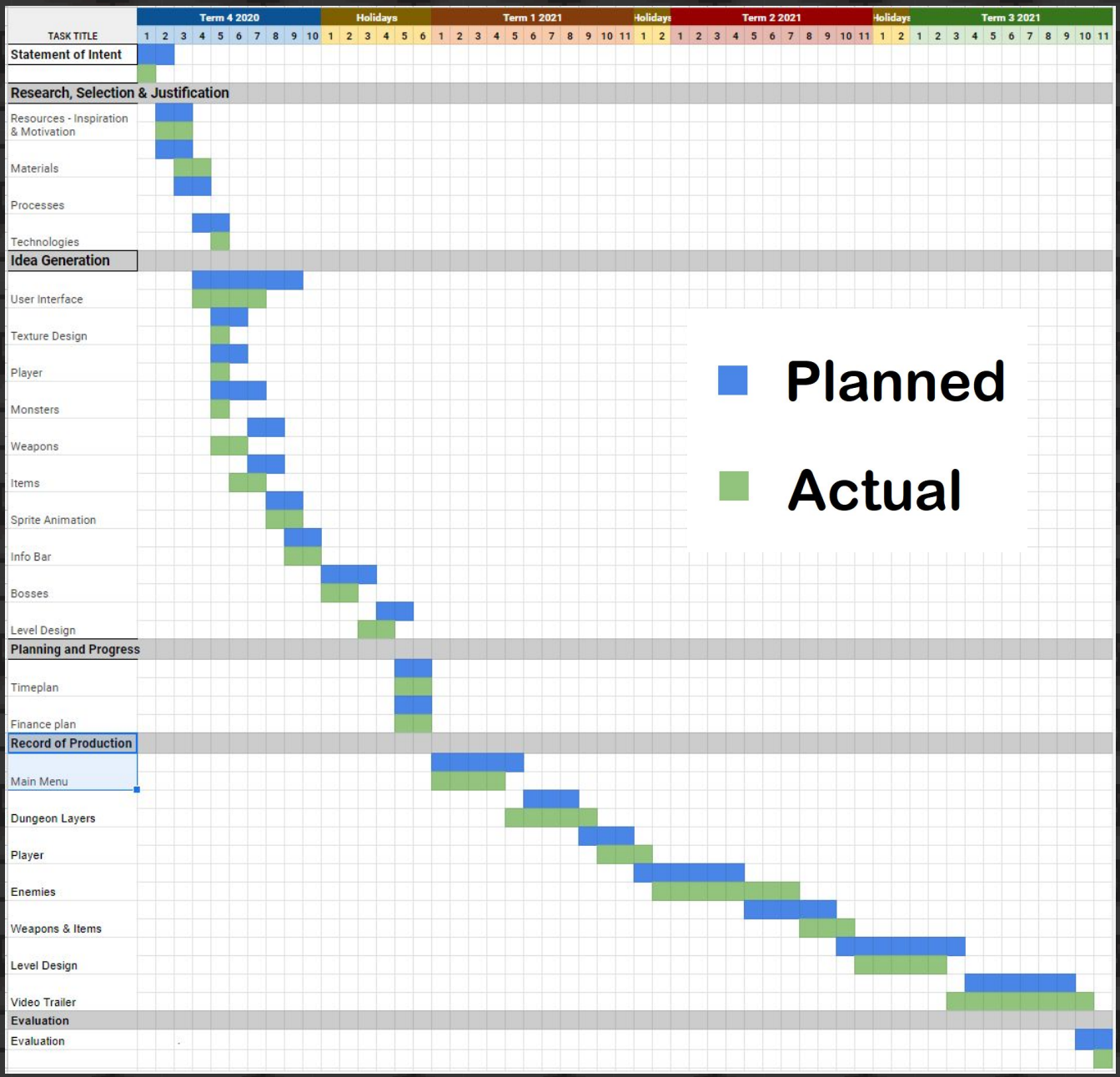
Financial Plan

FINANCE PLAN 2020-2021 | HSC MAJOR PROJECT MULTIMEDIA				
ITEMS	SUPPLIER	EST. COST	ACT. COST	PURPOSE
HARDWARE				
Home Desktop	MSY	$1,050.00	$1,049.00	The main device for all production
School iMac	School	$1,179.00	$0.00 (school supply)	The secondary device for all major project production.
Mouse & Keyboard Bundle	JB Hi-Fi	$100.00	$50.00	Operating my home desktop
JBL Charge 3 Speaker	Harvey Norman	$150.00	$150.00	General sound and music playback device
1TB Toshiba Portable Hard Drive	Harvey Norman	$60.00	$58.00	Extra storage required for game assets
WiFi USB Adapter	MSY	$50.00	$48.00	Connect home desktop to wifi connection
Wacom Intuos Drawing Tablet	Amazon	$130.00	$107.00	Main device for sketching and creating art
Sennheiser HD 4.50 Headphones	JB Hi-Fi	$200.00	$149.00	High quality sound playback for audio design
Akai MPK Midi Keyboard	DJ City	$200.00	$150.00	Midi controller for music production
SOFTWARE				
Adobe Premiere Pro CC	Adobe	$696/yr (Adobe CC)	$0.00 (student)	Video editing and manipulation
Adobe Photoshop CC	Adobe	$696/yr (Adobe CC)	$0.00 (student)	Drawing and photo manipulation
Adobe InDesign CC	Adobe	$696/yr (Adobe CC)	$0.00 (student)	Portfolio layout and design
Adobe Animate CC	Adobe	$696/yr (Adobe CC)	$0.00 (student)	Creating animations
Adobe Dreamweaver CC	Adobe	$696/yr (Adobe CC)	$0.00 (student)	Web Design
Jetbrains Rider	Jetbrains	$139.00/yr	$0.00 (student)	Code editor and debugger
Unity	Unity Technologies	$0.00	$0.00	Game engine
ADDITIONAL				
Mobile Data	Kogan	$200/yr	$204/yr	Internet access
Total		**$4,154.00**	**$1,965.00**	

The costs of my project turned out to be significantly lower than I estimated. The main purchases that I made were on my home desktop computer and the peripherals that I required for design. Most of these products however were ones I already owned, and so I did not need to make any more significant purchases solely for my major. Also, I anticipated that a lot of money would be spent on licensing software, however I found out that most of these applications offer a free educational license for highschool students. This helped save me a lot of money and make my project extremely cost effective.

Planning & Progress

Gantt Chart

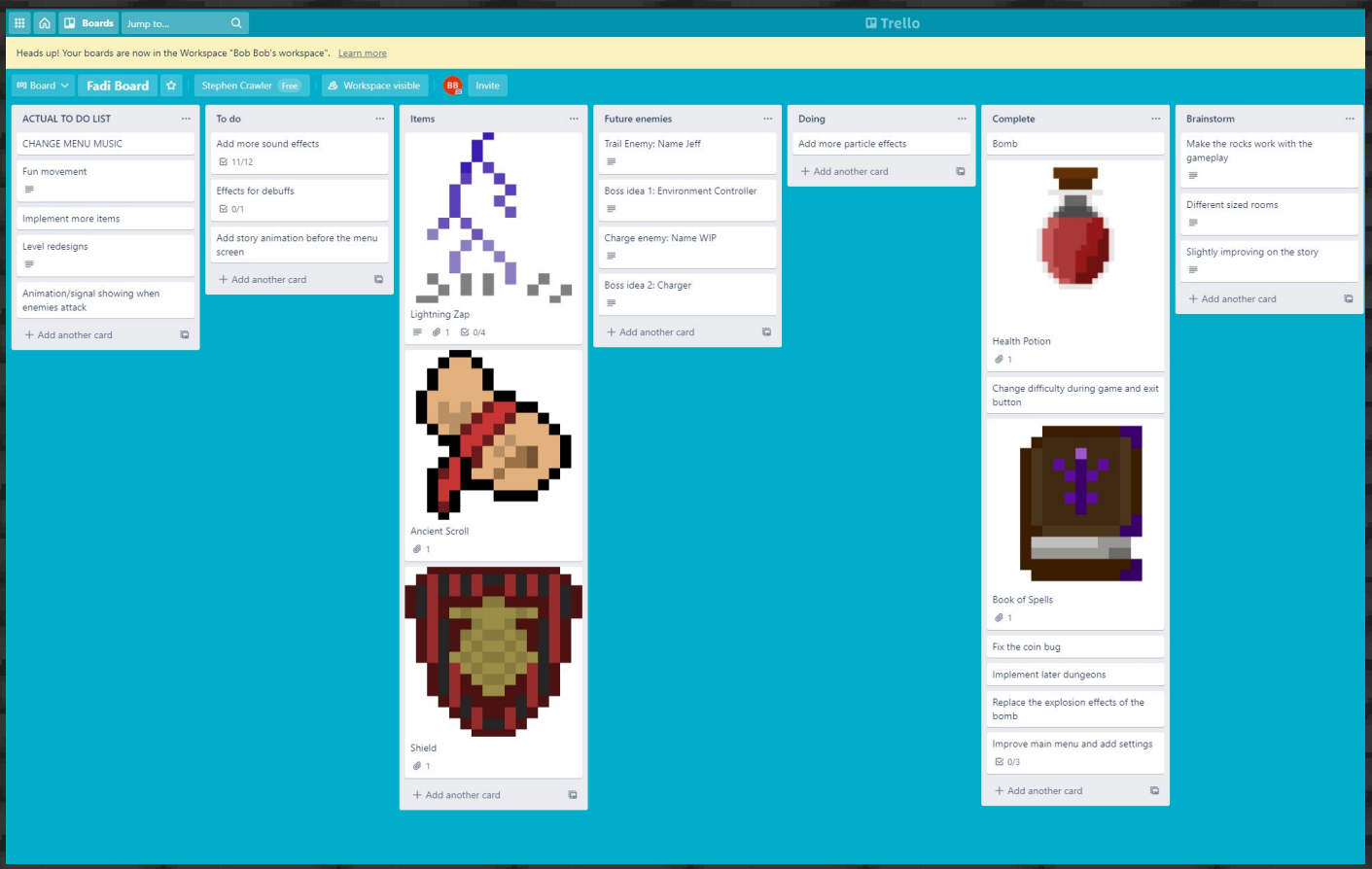


I feel like my time management throughout the whole year went extremely well and I managed to stay on track even through hurdles such as other ongoing exams/obstacles that came up during that time. I was extremely enthusiastic about making this project as it was a project that I really wanted to make and enjoyed working on. I began my planning and portfolio very quickly and even completed sections ahead of schedule.

Planning & Progress

Of course, the lockdown that the COVID-19 pandemic created did interrupt the communication and feedback opportunities with my teacher and took a slight toll on my motivation. I couldn't work in a classroom environment with support from my peers as well, or have my friends run any more play tests on my game, but I still managed to push myself through as I had all the tools I required at home.

A tool I utilised that greatly improved my time management and productivity was the website Trello.com. It allowed me to keep track of everything I wanted to add to my project in a ToDo list, with extreme detail including pictures, check boxes and links. I used this website as I normally have a hard time keeping track or remembering ideas, so as soon as I came up with something I wrote it down on Trello.



Record of Production

Main Menu

To create a landing page for the player once the game is launched, I required an aesthetically pleasing but functional Main Menu design. It had to follow the style of the old school 8-bit genre I was aiming for; meaning it had to look, sound and feel like a retro arcade game. My menu of choice had to hook the player's attention from the get-go, giving them an insight into the game while not being too over the top.

Originally, the design of my menu was composed of a simple image, title and button. The image was sourced from a free-to-use image database www.unsplash.com. I used a medieval-type font for the title "Dungeon of Death" and incorporated the same font for the "Play" button.

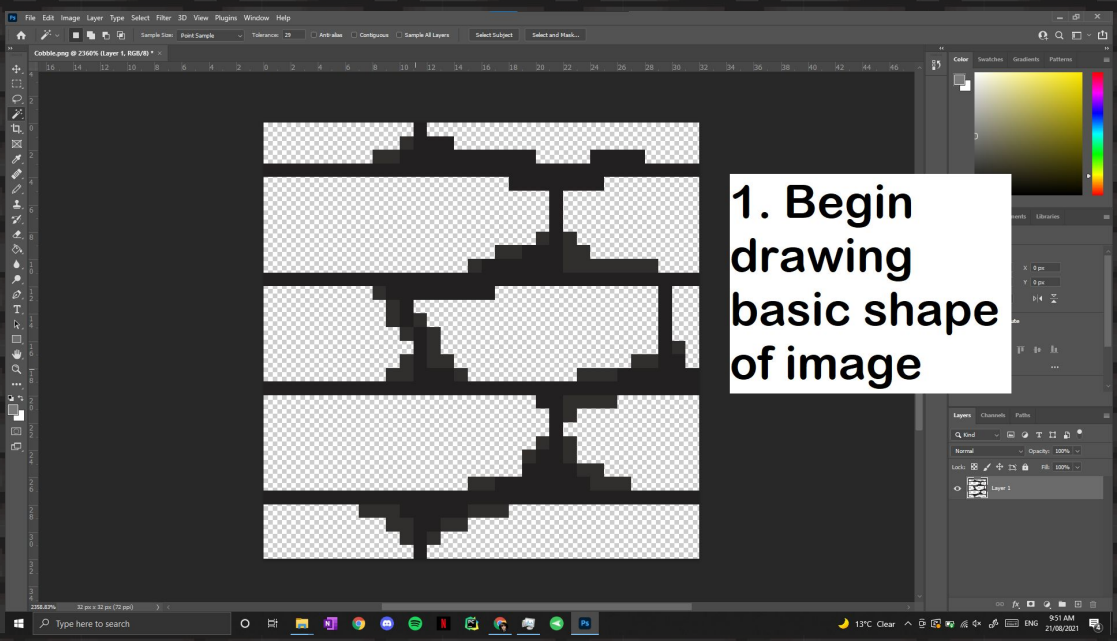
However, I decided to scrap this menu idea and start new as it simply did not fit the intended purpose of a retro video game. The design did not fit the feel of an old school arcade game, and it wasn't aesthetically pleasing.

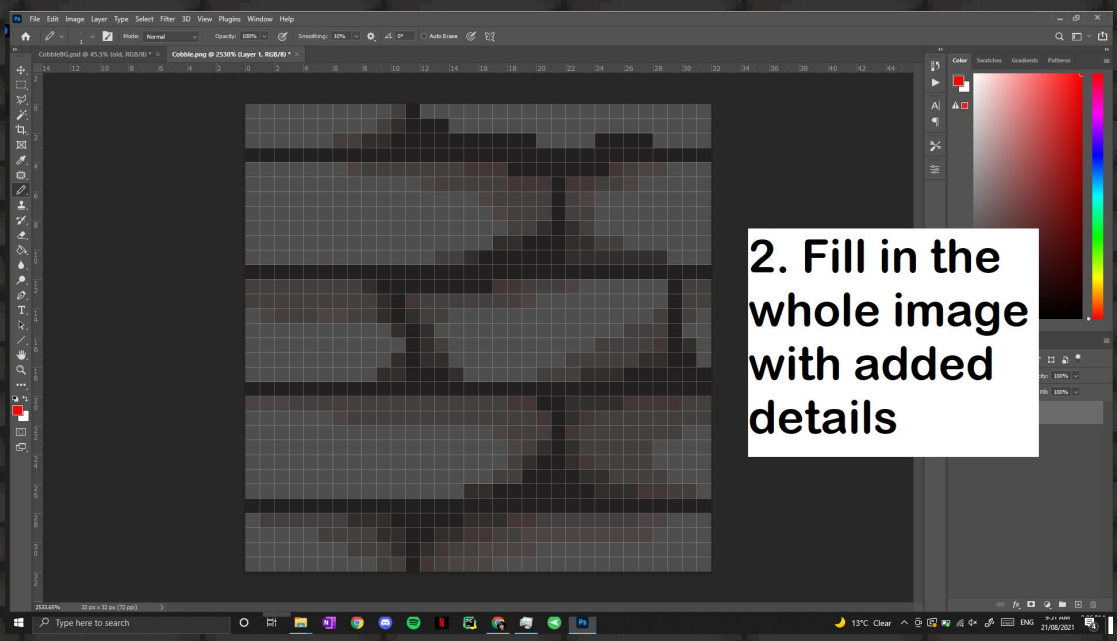
Record of Production

Instead, I decided to utilise the skill of drawing pixel art that I learned from Youtube tutorials, to draw the basic components of the menu that I would put together into a functioning landing page.

As I planned my dungeons to have a dark and grungy art style, I decided to make the main menu follow a similar style to blend in with the gameplay. To design the brick wall texture for the menu:

1. **I opened a Photoshop document and set it to the resolution of 32x32**
2. **I used the pencil tool along with a colour palette to draw the brick texture**
3. **I made refinements to the texture and added shading between the cracks**

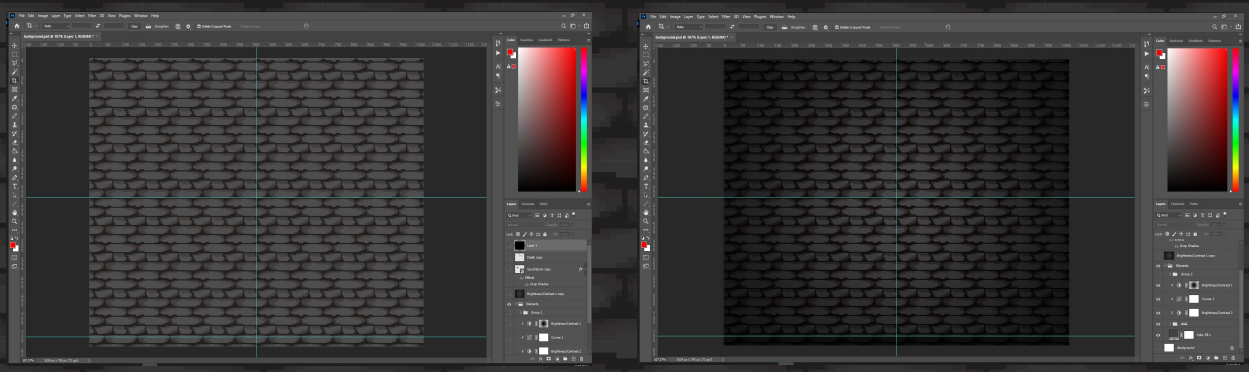




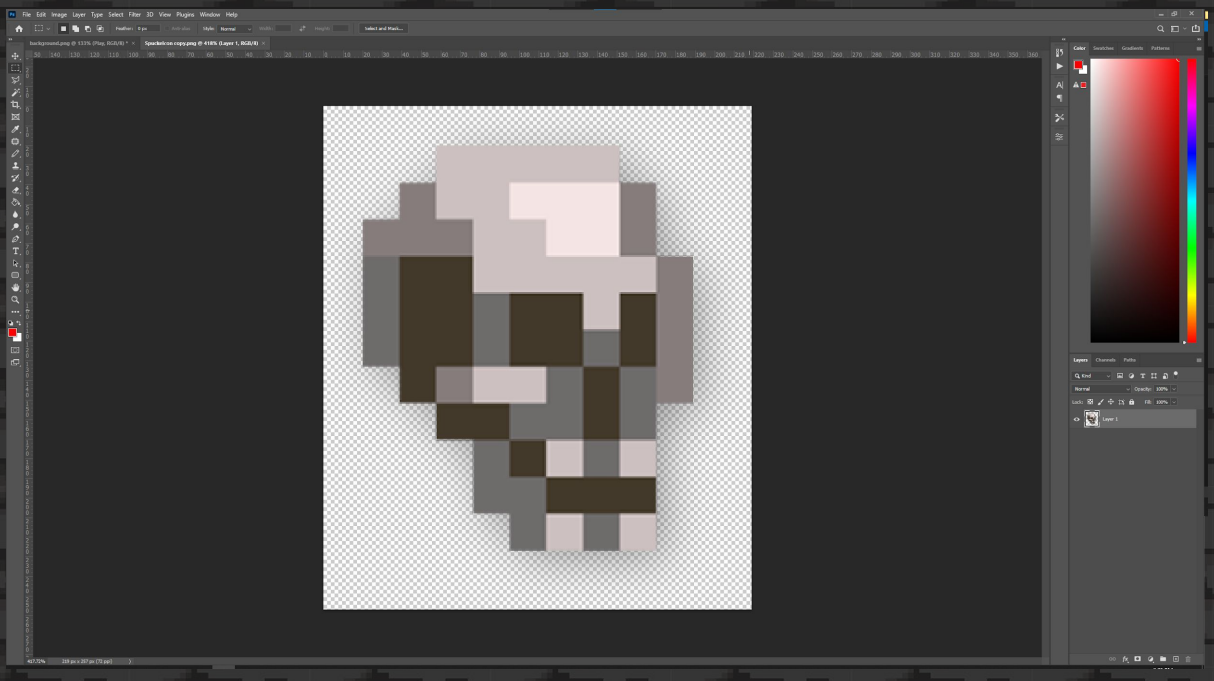
Record of Production

Next, I multiplied the brick wall texture a few dozen times and put them together to create a large wall image. The wall looked too light and low contrast, so I decided to change the brightness and contrast of the image. To do this, I:

1. **Extended the document boundaries by 10x horizontally and vertically**
2. **Copied and pasted the texture until it spanned the entire document size**
3. **Increased the brightness and contrast of the image**
4. **Added a dark but soft vignette circling the borders of the image**



I drew a simple pixel art image of a skull for the main logo sprite of my videogame. I placed this skull into the main menu and I'll also use it later on as one of the enemies in my game.



Record of Production

The menu needed a title, so I reused the medieval font from my original menu design as I quite liked the aesthetic of it. However it looked a little bland, so I experimented with special effects such as drop shadows and bevel and embossing. It turned out a lot better than I expected, looking really retro and almost 3D. The steps I took to accomplish this were, I:

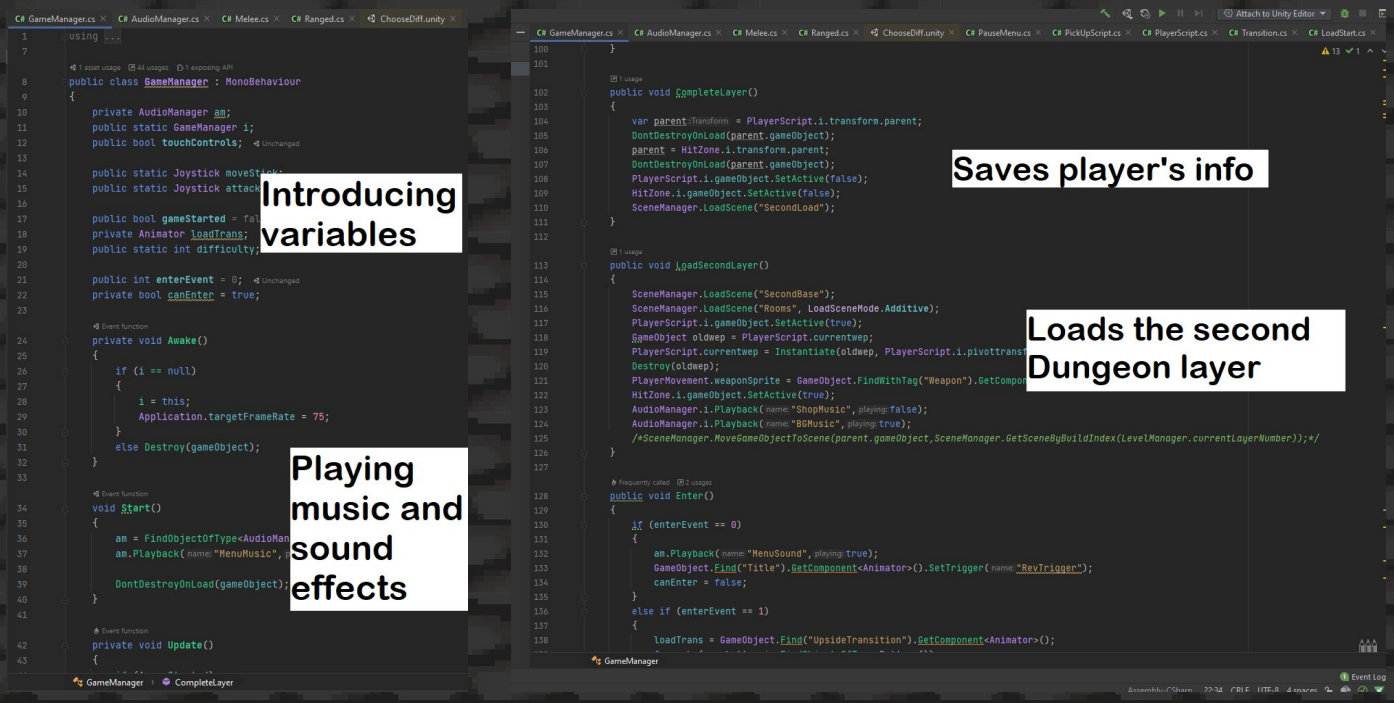
1. **Placed the pixel art image of the skull onto the canvas and added a drop shadow**
2. **Created a simple white title with the text "Dungeon of Death"**
3. **Centred and spaced these images on the canvas**
4. **Added glow, bevel & embossing and a drop shadow to the title**

At first, I struggled with drawing the pixel art of the skull and background wall as drawing doesn't come naturally to me. However, with constant practice and prototyping, I drew images that I was very pleased with and that resembled nostalgic graphics from old-school retro video games.

To actually make everything run, I had to code all the mechanics and scripts behind everything in the game. Coming off my very basic knowledge of coding in the Python language, it was a struggle to learn and create an entire full-scale game project. I followed tutorials and experimented with different coding methods and built my skills.

Record of Production

I coded a script named "GameManager" that stays running throughout the gameplay and controls everything from button presses to animations. For the Main Menu, I ran a listening function that waits until the user presses the Enter key to continue and begin the game.



The GameManager script ended up being one of the longest code files I wrote -- over 160 lines of code. It was extremely difficult to keep track of hundreds of functions running every second just to make the game playable. If I was to redo this manager script, I would make it neater and split it up into smaller files that are easier to keep track of.

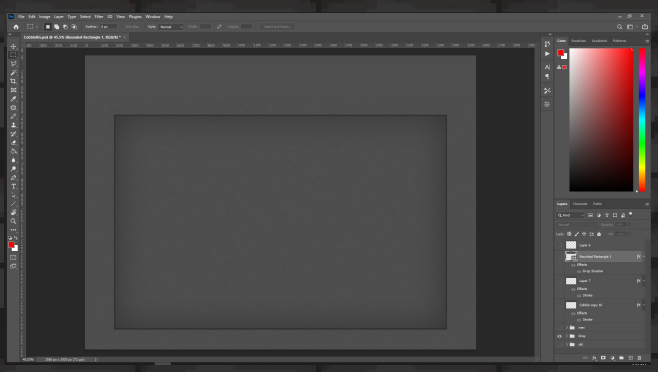
Record of Production

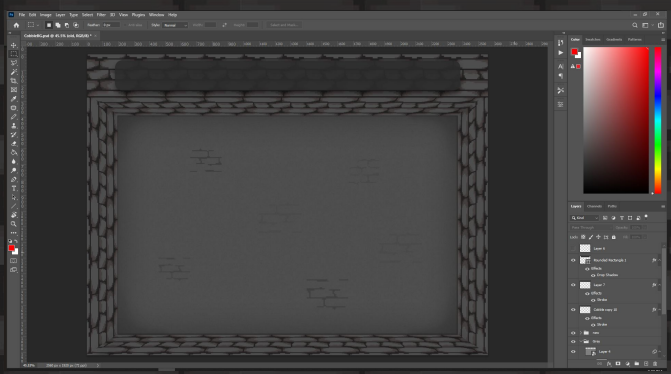
Dungeon Layers

A grungy, cracked and broken down basement is what I planned for the first layer of the dungeon. It will all be drawn in an 8-bit pixel art style matching the main menu and the aesthetic of old school retro games. To me, this style is extremely nostalgic and brings back so many memories of my childhood.

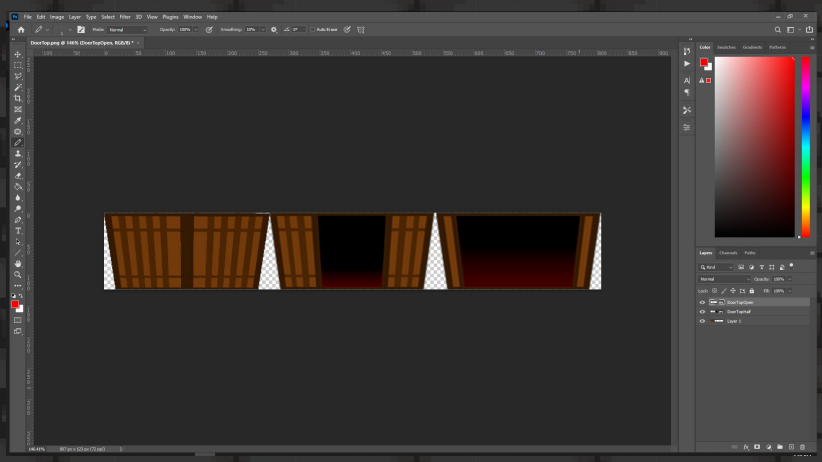
To create the first dungeon layer design, I picked a monotone colour palette consisting of greys and blacks. The look of each layer will be synonymous across all levels, but the design of obstacles and enemies will vary. To construct the base layer design, I:

1. **Formed a rectangle outlining the boundary of the playable level region**
2. **Took the previous brick texture and perspective transformed it to fit along the edges of the rectangle**
3. **Added shadows and small cracks across the level floor to add depth and atmosphere**
4. **Placed a slightly larger array of bricks above the level to house the health, bomb and coin stats**





The level required doors for the player to navigate through. They couldn't open as soon as the player walked into the level as they would be able to run straight through the enemies till the end. Therefore, I needed to make a closed door and a door-opening animation.

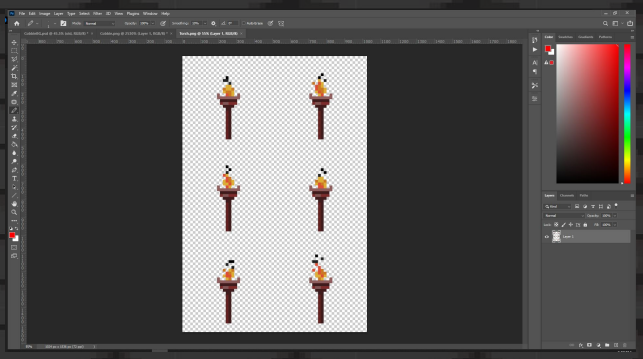


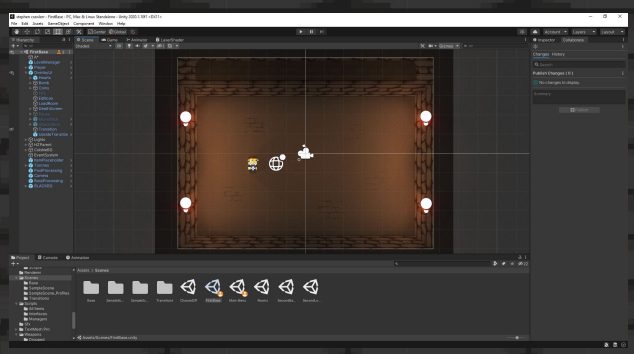
Record of Production

The level still looked quite bare and didn't meet my expectations at first. I thought I had done a great job at drawing all the aspects of the level but something was missing and I didn't quite know how to fix it.

I decided to try and implement a form of lighting into the game. The torches can be animated in Photoshop but the lighting couldn't; it's dynamic and affected by the player. To create the lighting, I:

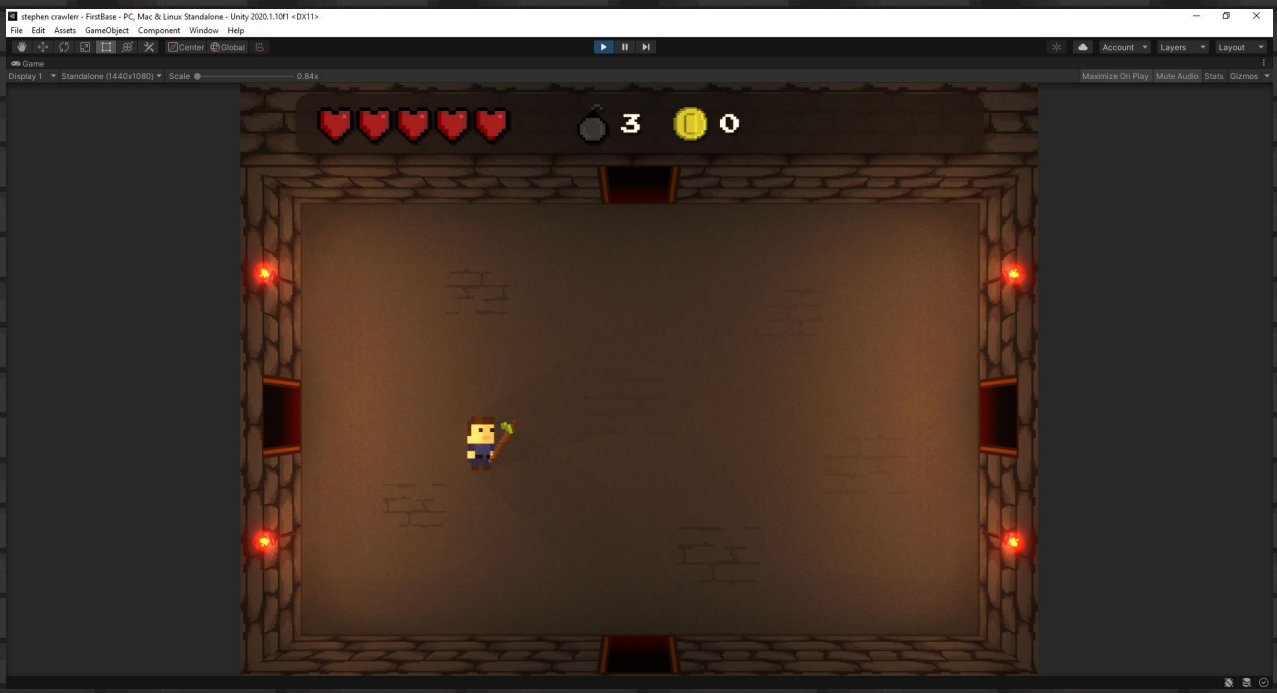
1. **Drew a torch animation sprite-sheet in Photoshop and exported it**
2. **Imported the level background into Unity**
3. **Imported the torch animation and placed torches in 4 places on the walls**
4. **Added Unity lighting effects and post processing**





The addition of lighting into the game through Unity improved the look of the game significantly. The contrast of the lights and postprocessing against the dark, gloomy background provided a very aesthetically pleasing background design. It fit perfectly between old-school retro and new modern graphics.

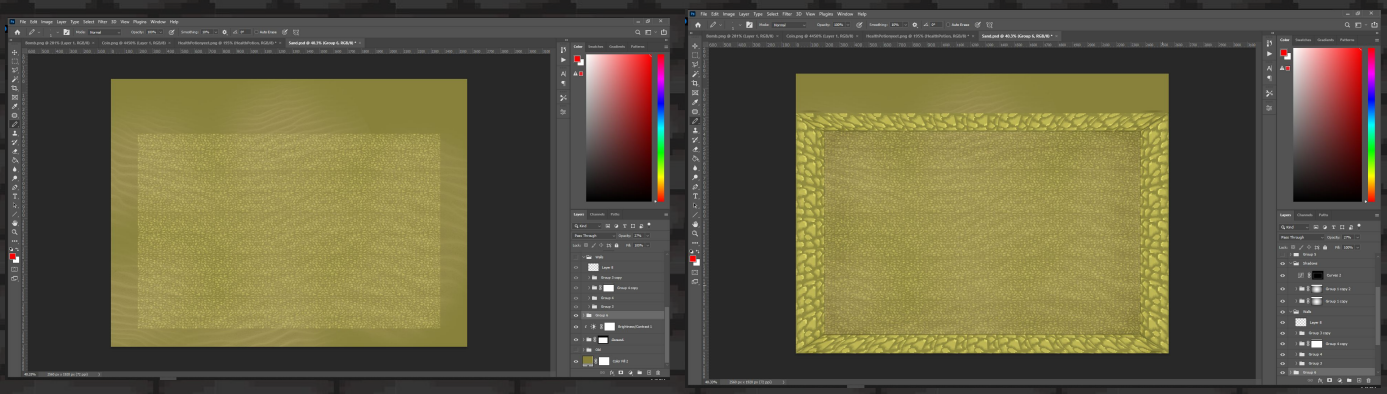
Record of Production



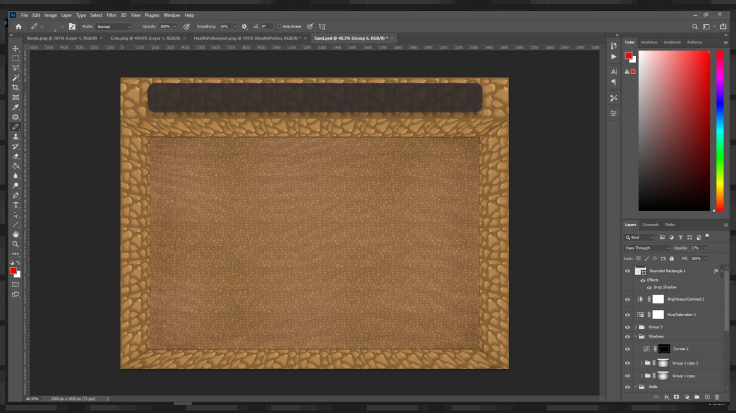
For the second layer of dungeons, the Sand layer, I wanted a similar design but a drastically different aesthetic. I wanted to go for a light, sandy, desert-like background with waves of sand running across.

I used the background of the first layer as a template. To draw and construct the sand layer, I:

1. **Drew a sandstone texture for the ground**
2. **Painted it lightly with waves resembling sand formations**
3. **Drew another sandstone texture, this time larger and resembling a wall**
4. **Added the box for game stats, torches and doors**

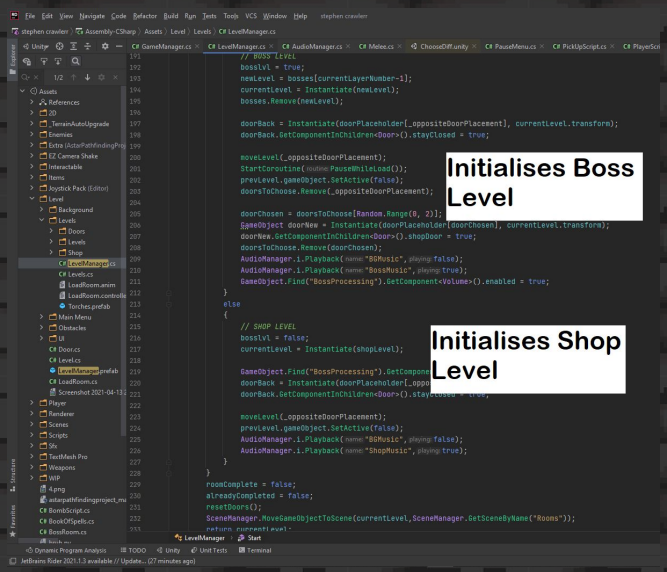


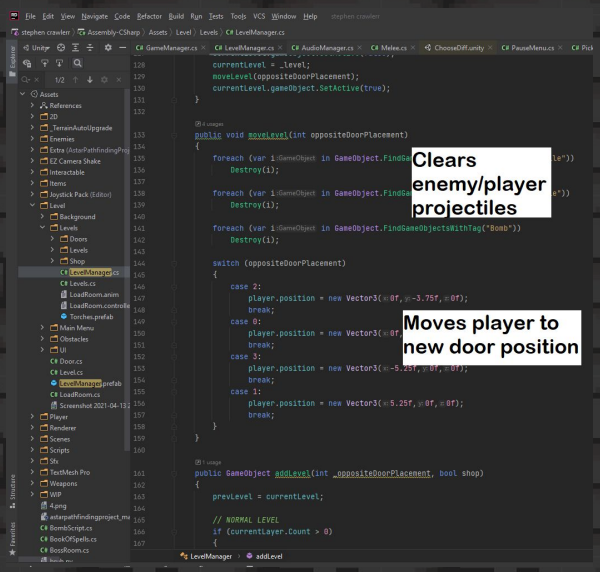
Record of Production



To make the levels respond to game events such as every enemy being defeated or the player being killed, I needed a Level Manager script to handle these events. When all enemies are dead, the doors open up, allowing the player to either discover new levels or venture through old ones.

The Level Manager required difficult coding techniques to generate new rooms while simultaneously keeping old rooms stored but disabled for later use. I utilised a C# Dictionary variable to store every room instance and link it to every door that it leads to.





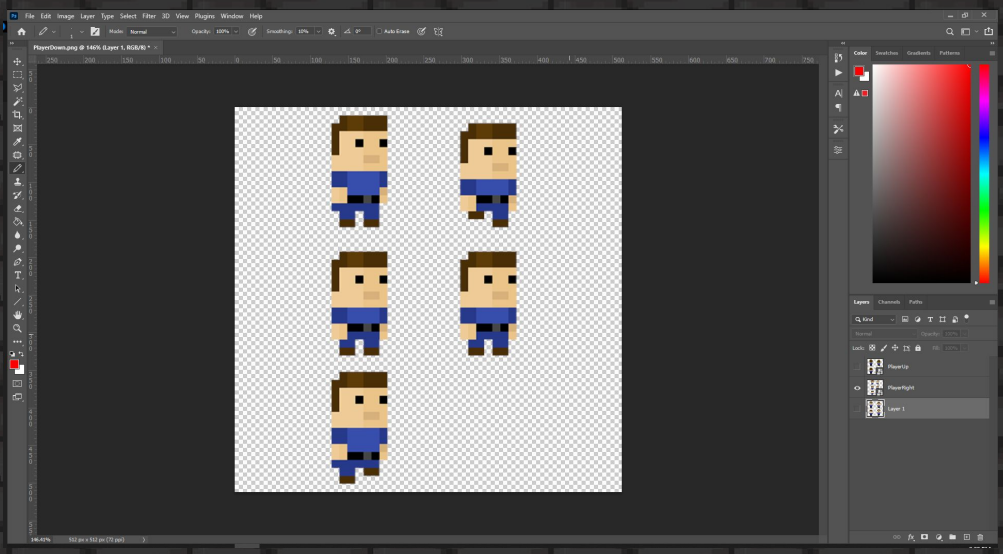
Record of Production

Player

The player character I was aiming to create was not a knight fit for the exploration of a dungeon, but a simple man from a village. He's not particularly strong and doesn't wear a suit of armour. This fits the narrative of the game; a simple man exploring a dreaded dungeon.

I took inspiration from my idea generation and drew a simple 8-bit player character design. Of course, the player couldn't just float across the screen when controlled, so I needed to draw a sprite-sheet animation for it. To draw the player animation, I:

1. **Drew the basic figure of the player**
2. **Extended the canvas and duplicated the player a few times**
3. **Adjusted each image slightly to give the impression that the player is walking**



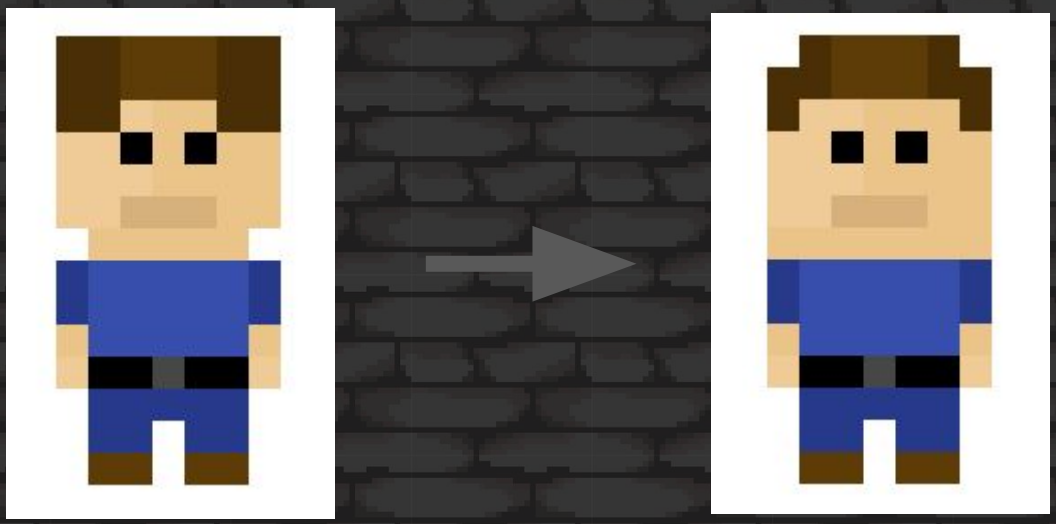
I realised that as the player walks around the level, the animation looks quite unnatural since it doesn't look up or down, only sideways. This bugged me, so I then decided to draw two new animations to correct this issue; one facing up, and one facing down.

Record of Production

To draw these new animations, I:

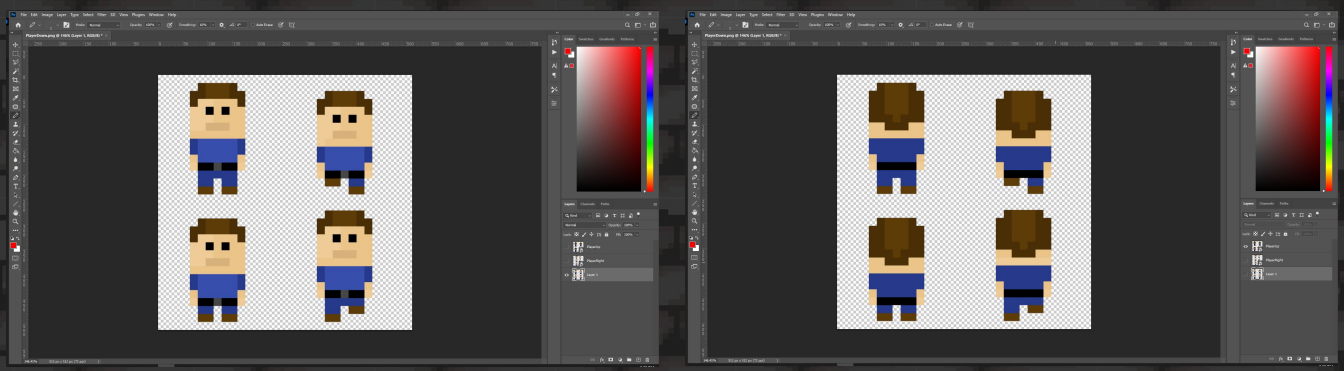
1. **Drew the same basic figure as the original walking animation of the player**
2. **Adjusted the shape of the face and body to match up/down direction**
3. **Duplicated the image and corrected legs to create walking animation**

I experimented with different prototypes of the player design, one with a small neck and a boxier hairline, but it did not suit the character and made him look rather odd.



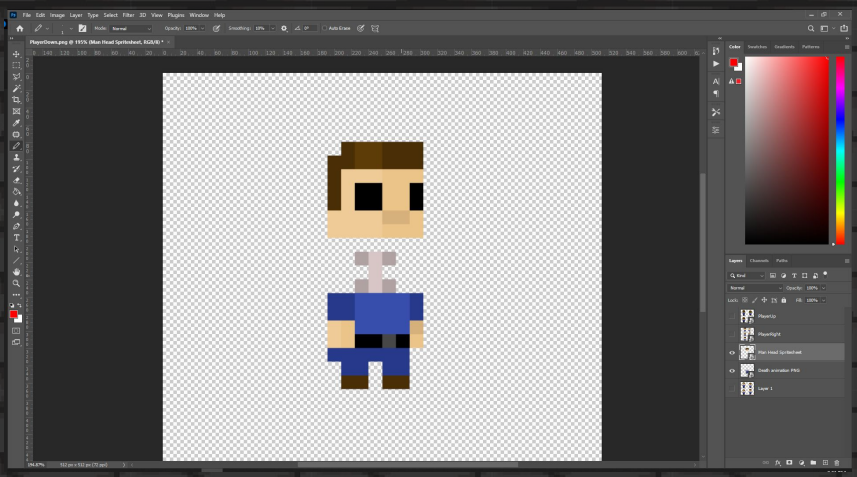
Although the neck was quite wide, I liked this version of the player model better and it made it a lot easier to draw the side and backward animations.

To animate the player facing to the left, I decided to just flip the animation across the X-Axis in Unity, instead of drawing a whole new animation for it.

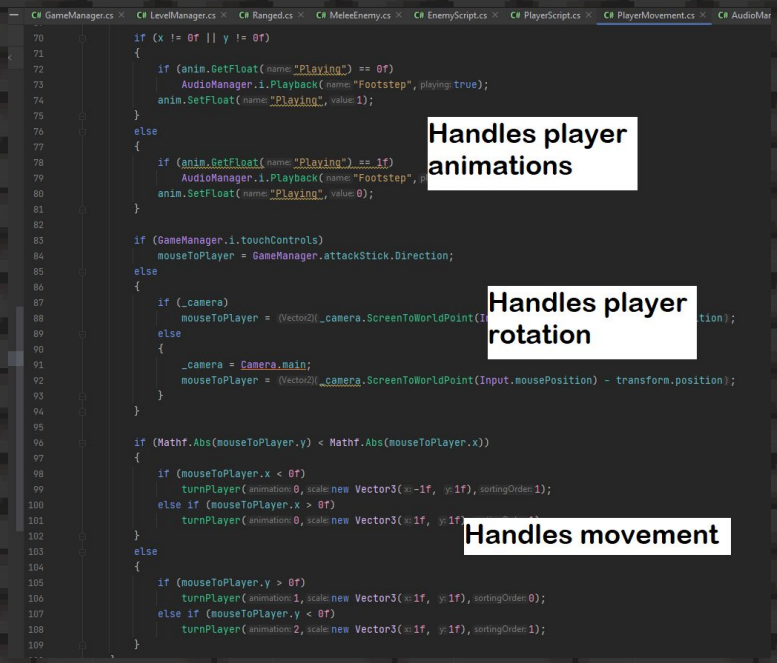


I also created a death character sprite for the player when it runs out of health. I did not need to animate this death in Photoshop as I will use Unity physics to simulate this for me. The eyes are blacked out and the player's head is detached from the body.

Record of Production



To code a functional and playable character, I needed to develop many systems including movement, animation, health and attack. I started coding the basic movement of the player and added in all the animations, along with attacks using a placeholder weapon. I also added a health counter represented by five heart icons.



The player animations followed the direction the player was moving but not where it was attacking. While just running around, this looked fine, but when initiating any attack it looked extremely unnatural. I didn't want the player to look like it was attacking backwards, so I developed a brand new system that makes the player face positions relative to that of the mouse.

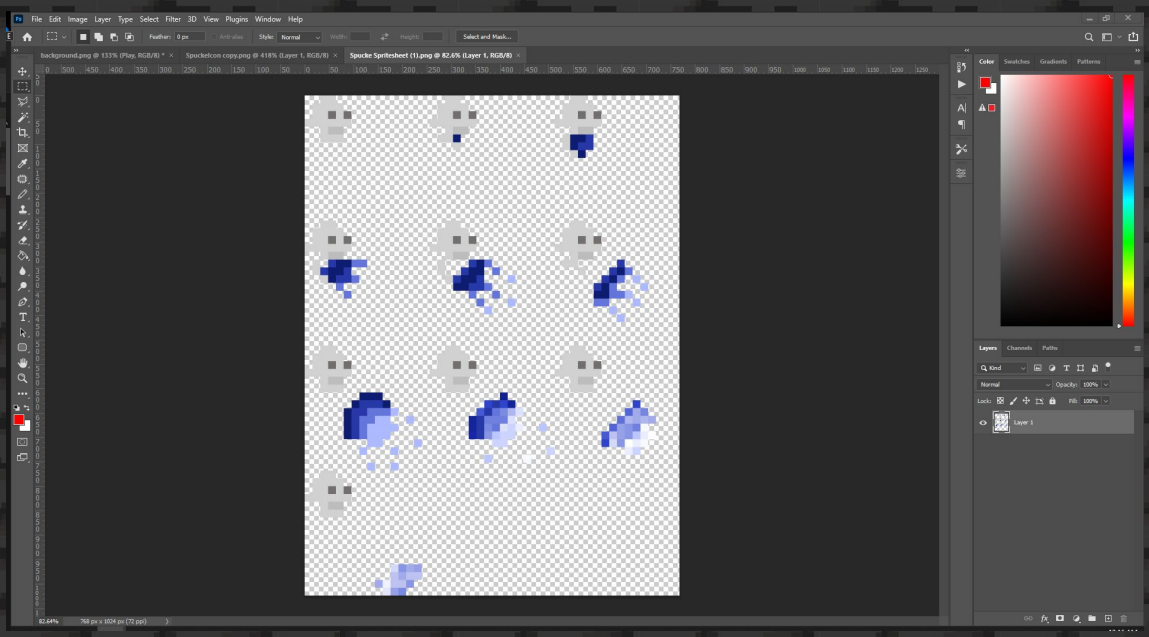
Record of Production

Enemies

Similar to the player animation designs, each enemy requires at least 3 unique sprite-sheet animations facing to the side, up and down. These enemies will be largely fantasy and scary; akin to something out of a horror movie but toned down. They will also be drawn in an 8-bit art style in accordance with the rest of the video game to resemble an old retro arcade game. This will also ensure that I meet my initial intentions for the game and also my target audience.

Spucke

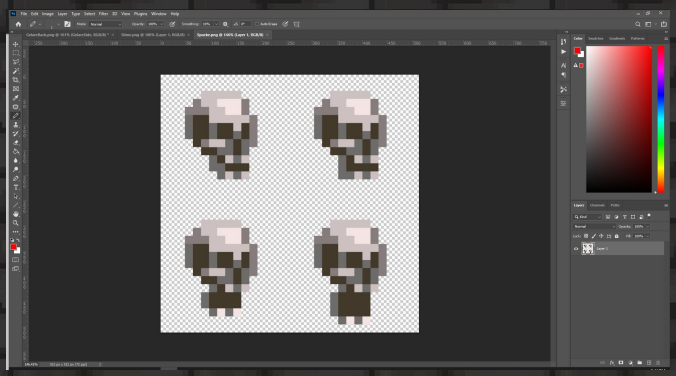
With the experience I gained from drawing the menu and player design and animations, I drew several prototypes of enemies in Photoshop.

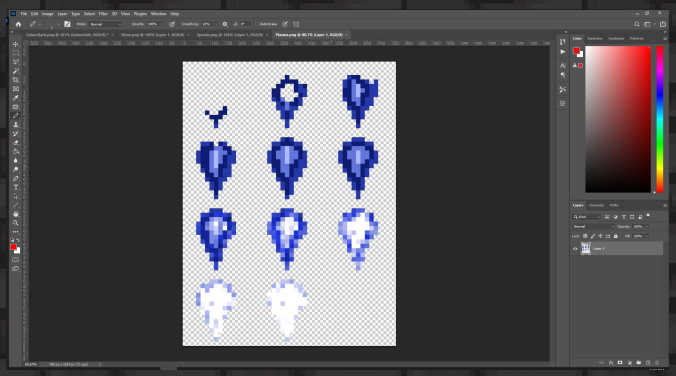


The first prototype of the skull shooting plasma balls looked terrible and was not close to the level of quality I desired. I spent a long time on this but I could not figure out how to make the animation look natural in a singular sprite sheet. However, I know that this first step is necessary even if it doesn't meet my expectations.

Record of Production

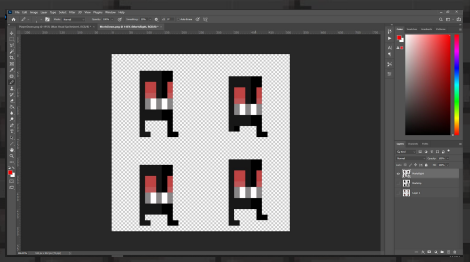
To fix this problem, I realised that I needed to split the creation of the plasma ball into 2 animations: the skull and the plasma ball. I called this skull enemy a Spucke.

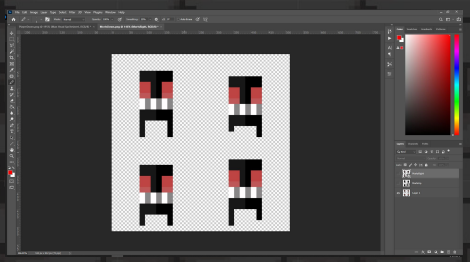


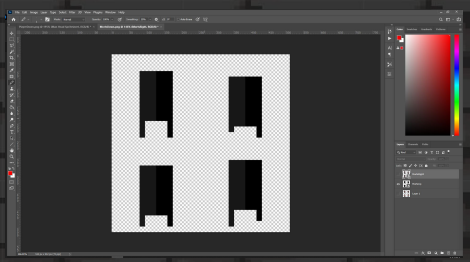


I used the same processes to create the rest of the enemies in the game. Based on my idea generation, I drew several more monsters to incorporate into my game.

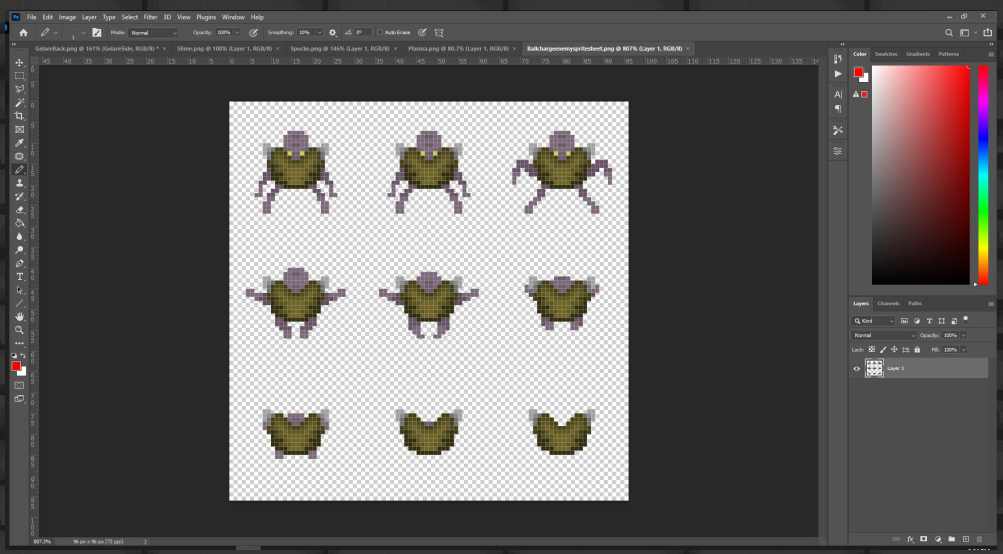
Mort





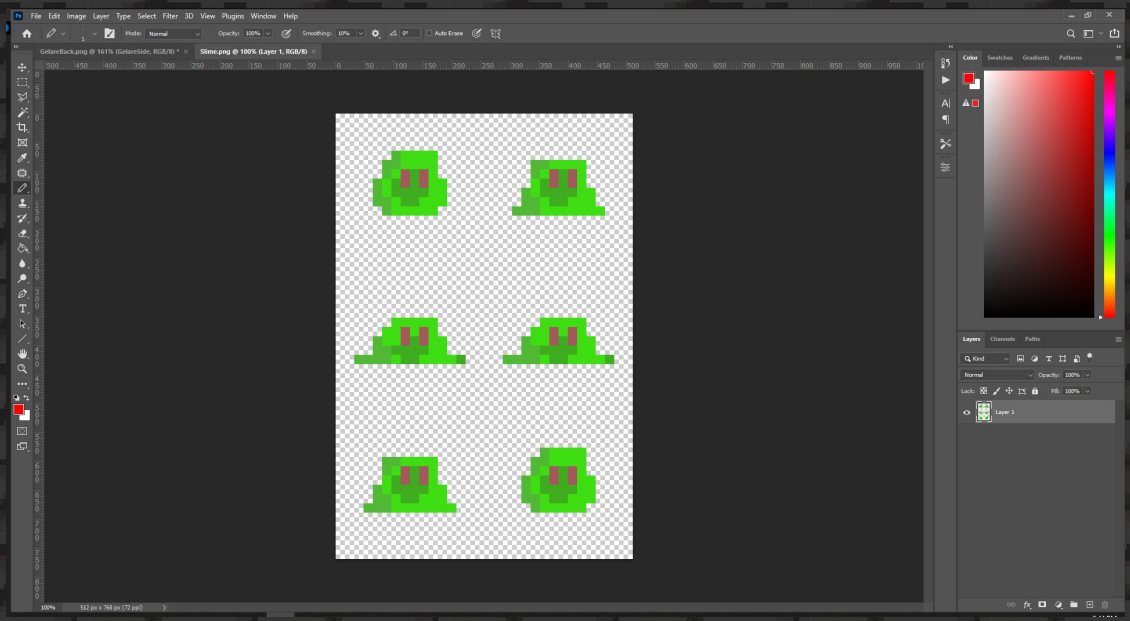


Charger

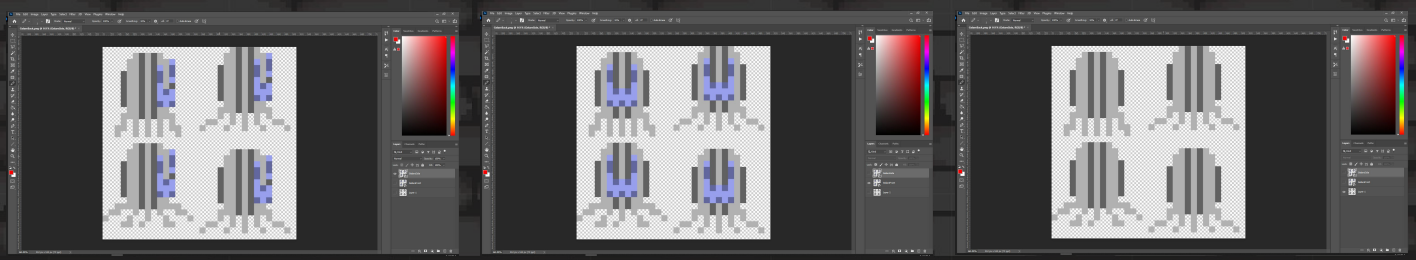


Record of Production

Slime

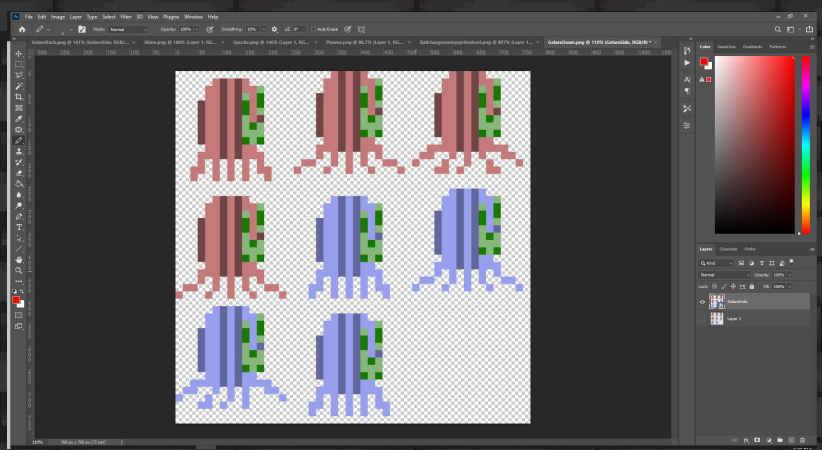


Gelare



Gelare Boss

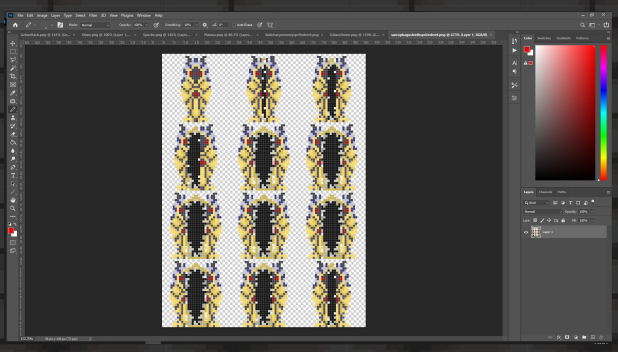
The Gelare Boss is basically an overpowered Gelare with high health and attack damage. It moves completely differently to the Gelare and is much larger. I took the design of the normal Gelare and tweaked the colours to represent when it is in the attack phase

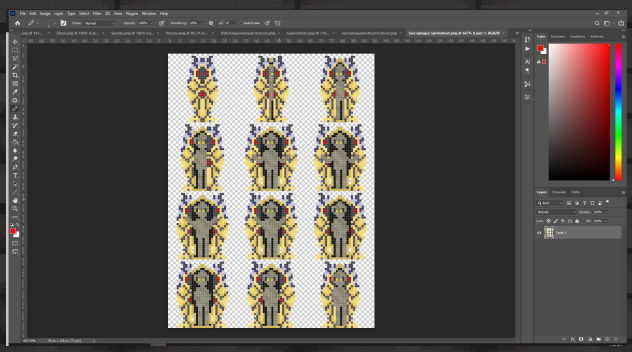


Record of Production

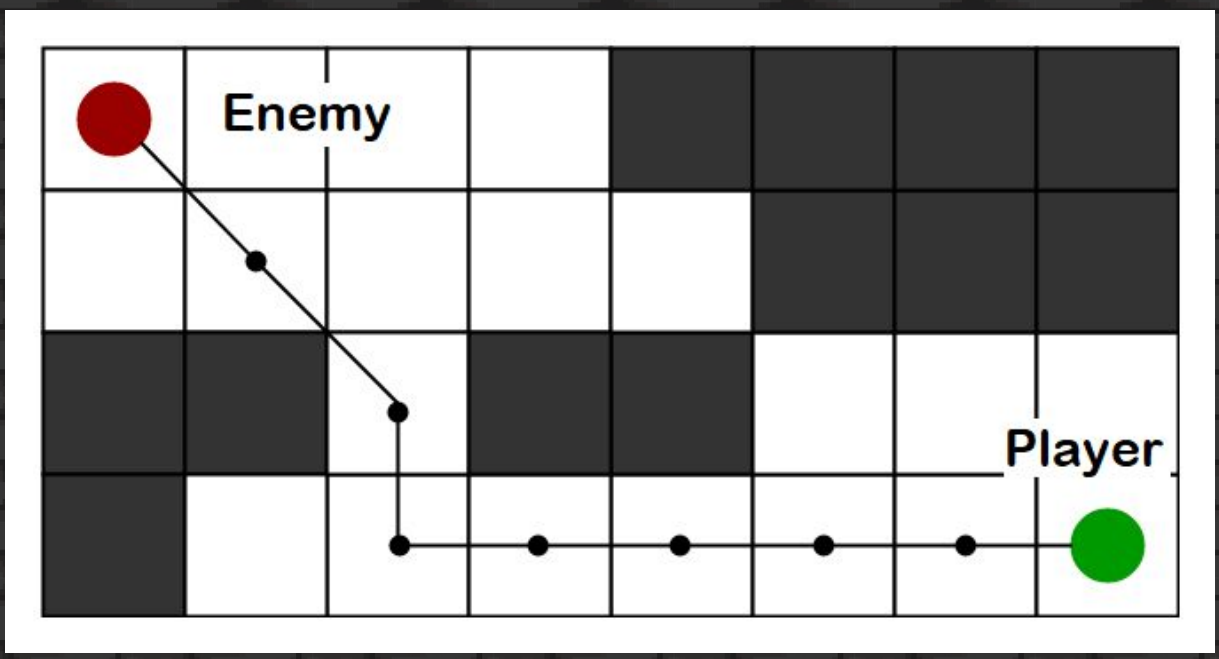
Sarco

The Sacro is the final boss the player faces. It never touches the player but instead summons other monsters to do its job. Also, it shoots lasers as it floats across the screen. The spawning animation took many frames to play smoothly.



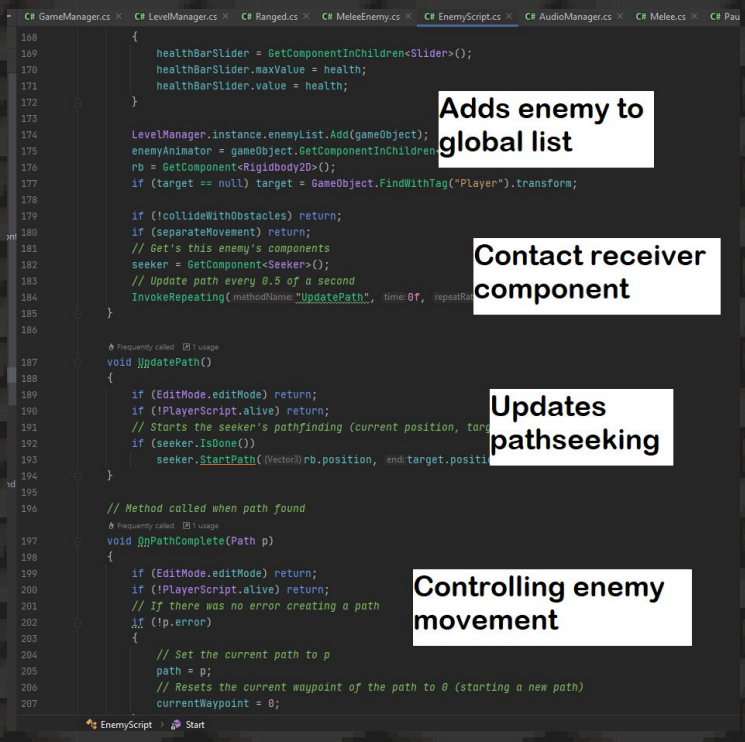


To code the enemy movement and attack script, I was able to reuse sections from the Player script to move and animate each enemy. However, I was forced to learn how to code Real-time AI Pathfinding so the enemies followed the player around the map.



This diagram serves as a visual representation of the AI Pathfinding algorithm that I incorporated into the enemies' source code. The black squares represent grid sections where an obstacle has been placed such as a rock or a chest, and the white squares represent empty space. The enemy script that I created utilises this spatial awareness, along with factors such as speed and weight, to calculate the exact path it needs to take to track the player.

Record of Production



The first prototype of the skull shooting plasma balls looked terrible and was not close to the level of quality I desired. I spent a long time on this but I could not figure out how to make the animation look natural in a singular sprite sheet. However, I know that this first step is necessary even if it doesn't meet my expectations.

Weapons & Items

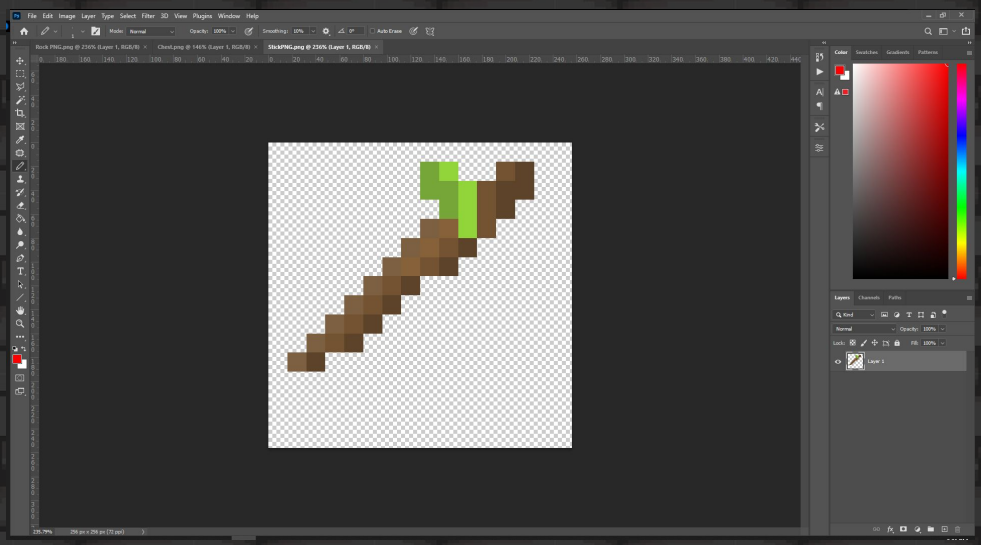
Monsters will need to be killed with the use of several weapons implemented into the game and found across levels. These weapons can be picked up and dropped, but only 1 can be carried at a time. This ensures that there is an added level of challenge for players as well.

The weapons I included in the idea generation of this game have two types: Melee and Ranged. Melee weapons include the Sword and the Stick. Ranged weapons differ to melee as they are not used directly, but instead shoot projectiles such as arrows and particles. These ranged weapons include the Bow and the Book of Spells.

Record of Production

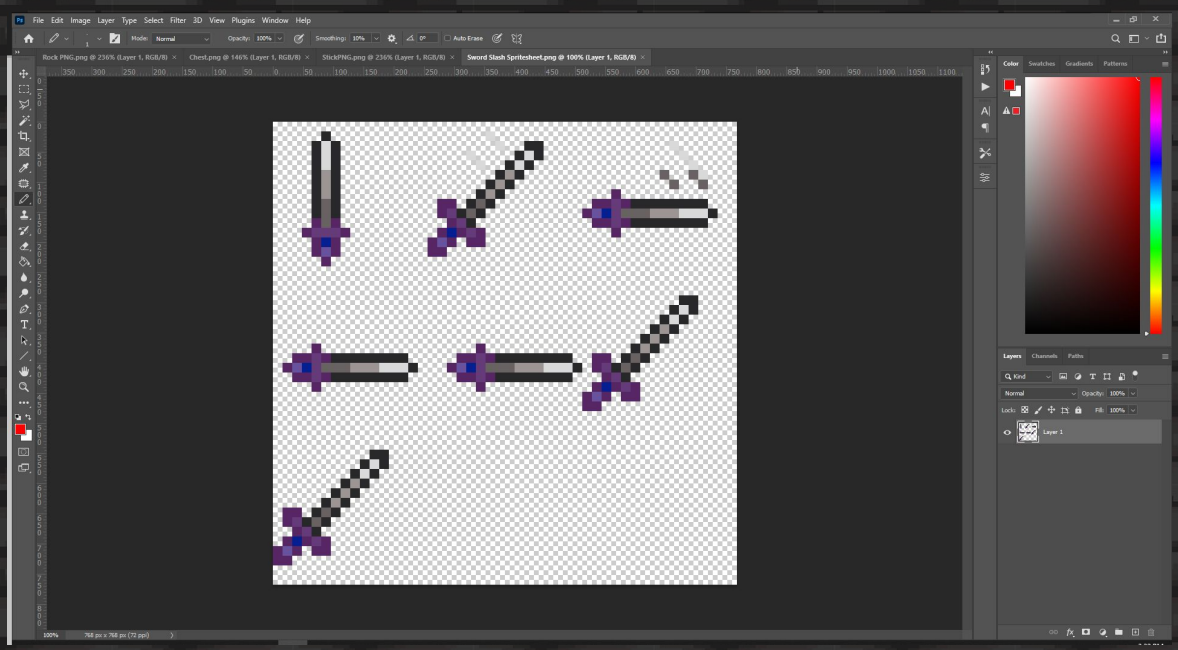
Stick

The stick is the most basic weapon in this game. It is provided to the player at the beginning level, and is therefore the weakest weapon. The swing attack is animated in-game, so I only drew the singular sprite image.



Sword

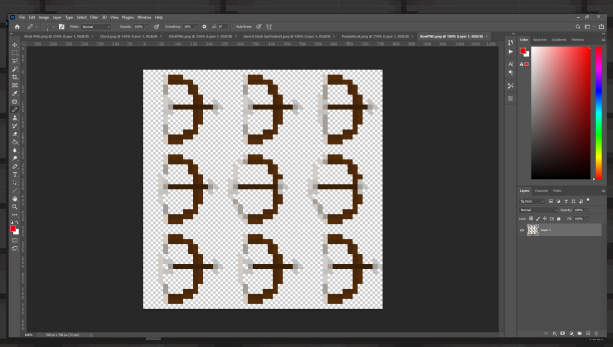
The sword is more complex than the stick as it has swinging effects that are animated on the sprite sheet.

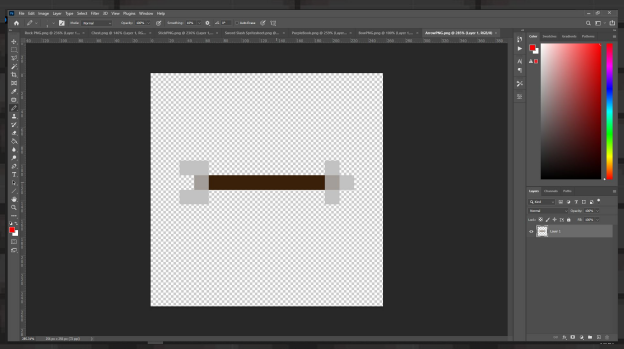


Record of Production

Bow

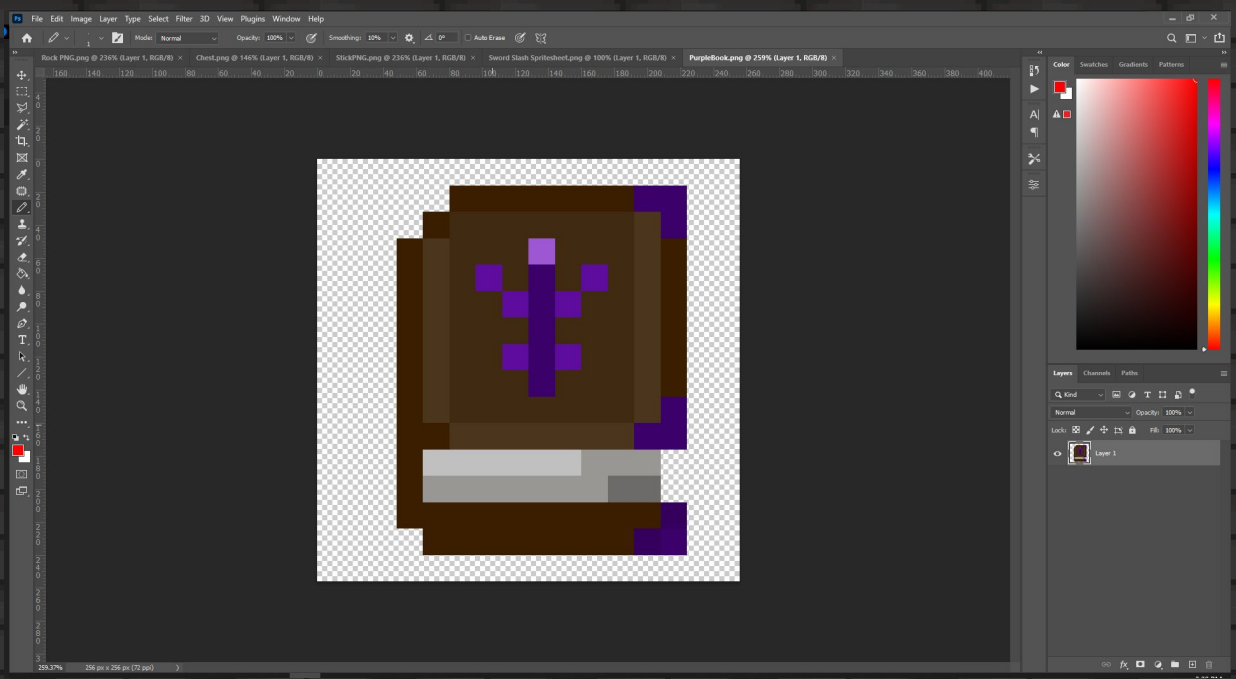
The bow is split up into 2 parts: the Bow and the Arrow. Initially, the bow and arrow were in the same animation. However, as the animation ends, the arrow is spawned in-game as its own separate object





Book of Spells

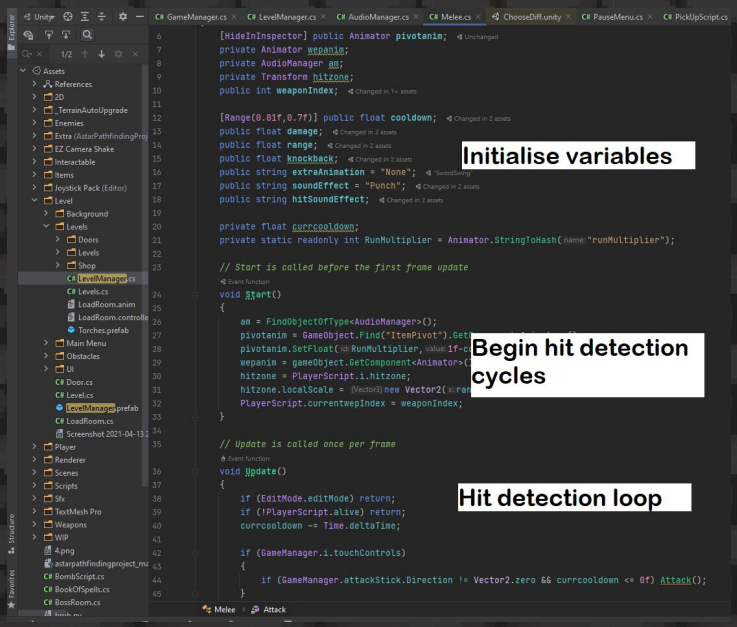
When utilised, the book of spells fires small square particles that glow repeatedly and bounce off walls before coming in contact with enemies. I drew just the book and then added in the particles inside the Unity Game Engine.



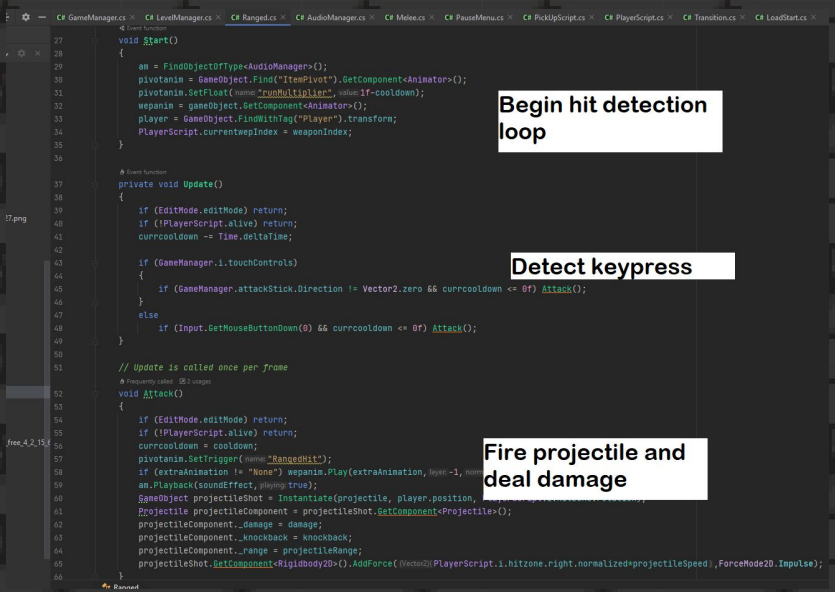
Record of Production

As the weapons come in melee and ranged, they operate quite differently from each other and require 2 unique scripts to run. The melee script for the stick and sword was relatively easy to write. To create the melee weapons, I:

1. **Created a hitbox that follows the mouse relative to the player's position**
2. **Wrote a script to detect what enemies laid inside the hitbox**
3. **Wrote a function to detect when "left click" was pressed and damaged the enemies**



The script for the ranged weapons was a bit more difficult to implement as I had to summon a new game object, be it an arrow or a particle, and add a velocity with the correct degree of rotation. Through this process, I was able to make sure that the weapon itself did not cause the damage, but the projectile that hit the enemies.



Record of Production

Level Design

To maintain a perfect balance of difficulty and also playability, levels will have to be designed in a layout where they are not impossible to beat, but also not too easy that it takes away the challenge of playing. The player keeps all their stats including health, bombs, coins and weapons from one level to the next. Factors on whether the player can defeat the entire game rely not purely on skill, but on luck as well. This adds variation to each playthrough of the game and creates a feeling of risk and uncertainty.

I began by sketching a prototype of a level, experimenting with different placements of items, weapons, rocks and monsters. I made sure not to add too many or too few entities in one level, as I wanted to avoid making it feel either too crammed together or too empty. After I settled on an aesthetically pleasing, playable and challenging design, I began creating the level inside the Unity game engine.

Record of Production

After implementing all the elements of the level inside Unity, I booted up the game and played it through with varying levels of health, different weapons and items. When my testing was complete and I found it good enough to play, I saved the level and added it to my LevelManager script. I repeated this process 12 times for every level I added into the game, ensuring the highest level of quality design in each.

A problem that I didn't realise at first was that through hundreds of runs of playtesting, I built up a lot of skill in my own game that made the levels feel easier to play. When I tested new levels that I created, their difficulty felt perfect in each playthrough. However, when I asked my friends and family to play the game for the first time, they struggled greatly in surviving even the first few levels. This idea of peer review and feedback made me realise that newer players will struggle much more than I do, which greatly helped me in balancing the difficulty of past and future level designs.

Through a lot of testing, as well as trial and error, I created several more levels adding up to 12 in total, including the boss fights.

Record of Production

Video Trailer

The video trailer for Dungeon of Death is responsible for teasing and showing off the quality of the gameplay including short cut clips of enemies, weapons and levels that the player can expect to see. Not only does it give insight into the game to the viewer, but it also provides a cinematic movie-like experience which hooks the viewer's attention and makes them eager to play the game themselves.

I took a look at the trailers of video games that inspired me to create my own. I paid attention to other 2D retro dungeon crawlers, dark and rogue-like games and how their developers created a trailer that sold millions of copies of their game and amassed enormous fanbases. They all seemed to have quick-cut edits, tense music and a cinematic vibe.

Spelunky (2012)

Binding of Isaac: Repentance (2021)

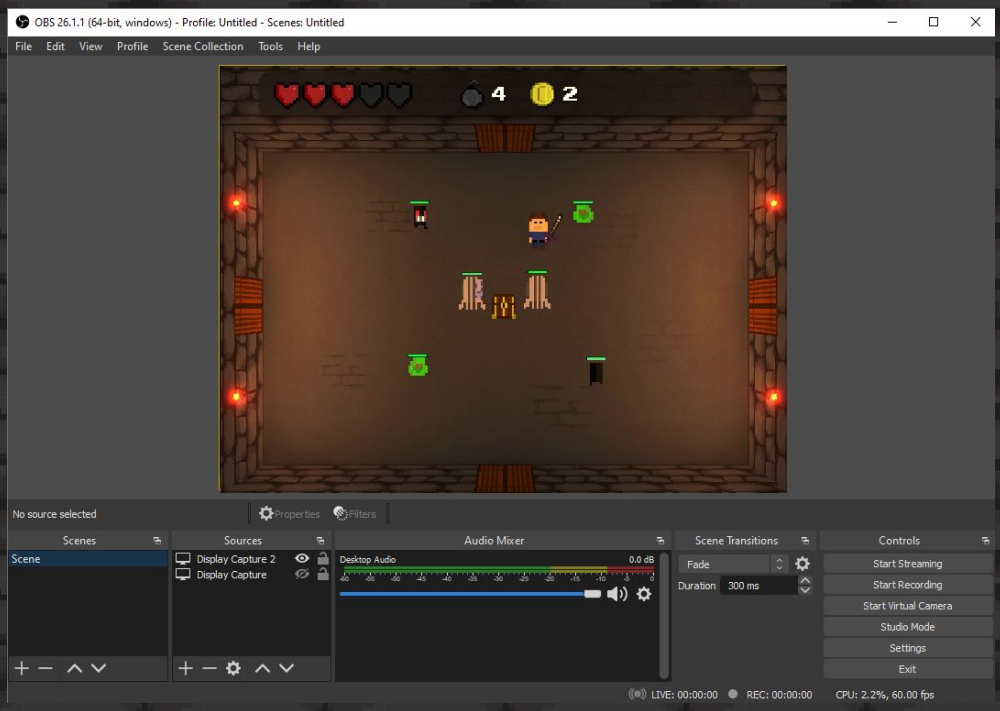
Terraria (2011)

In order to create the trailer, I had to record short clips of gameplay from my game and stitch them together in a video editing suite. To capture these video clips, I used a program called Open Broadcaster Software (OBS) which allowed me to film gameplay with high resolution and a high refresh rate with minimal loss of image quality.

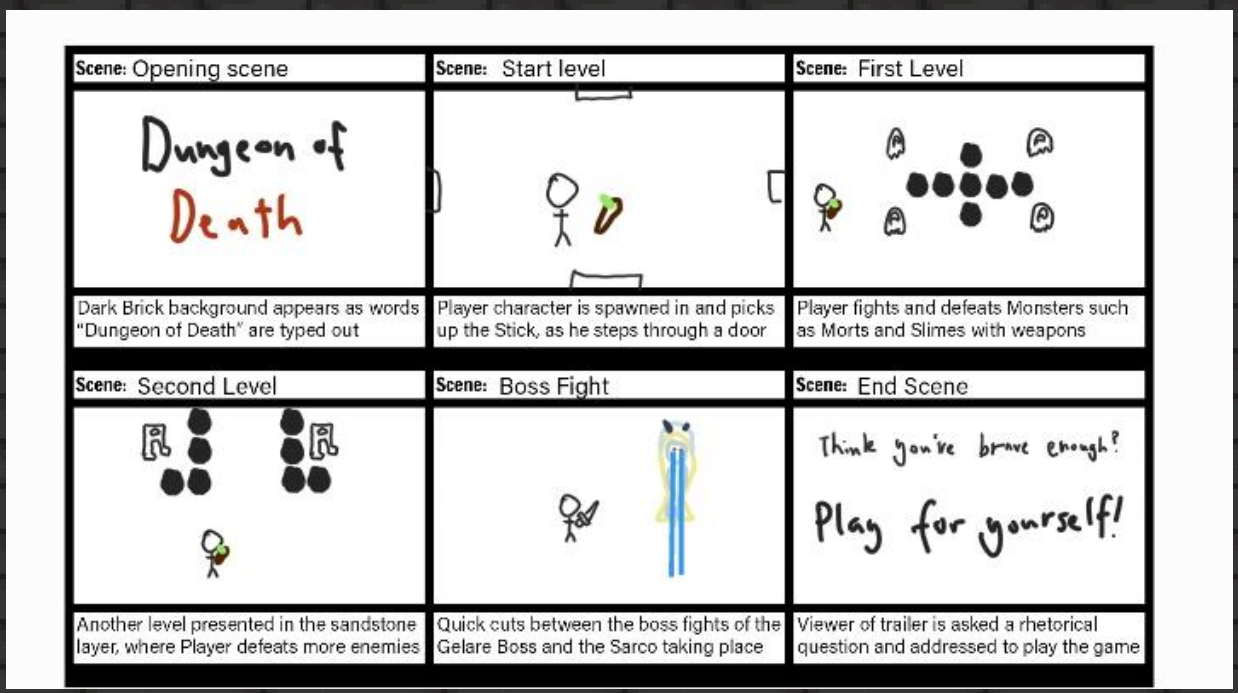
The approach I took in recording these clips was:

1. **Launch Dungeon of Death and OBS**
2. **Set up OBS to capture the game window at the correct resolution**
3. **Switch to the game application and hit the hotkey (F5) on my keyboard to begin recording**
4. **Complete a few levels of the game**
5. **Press (F5) again to end recording**

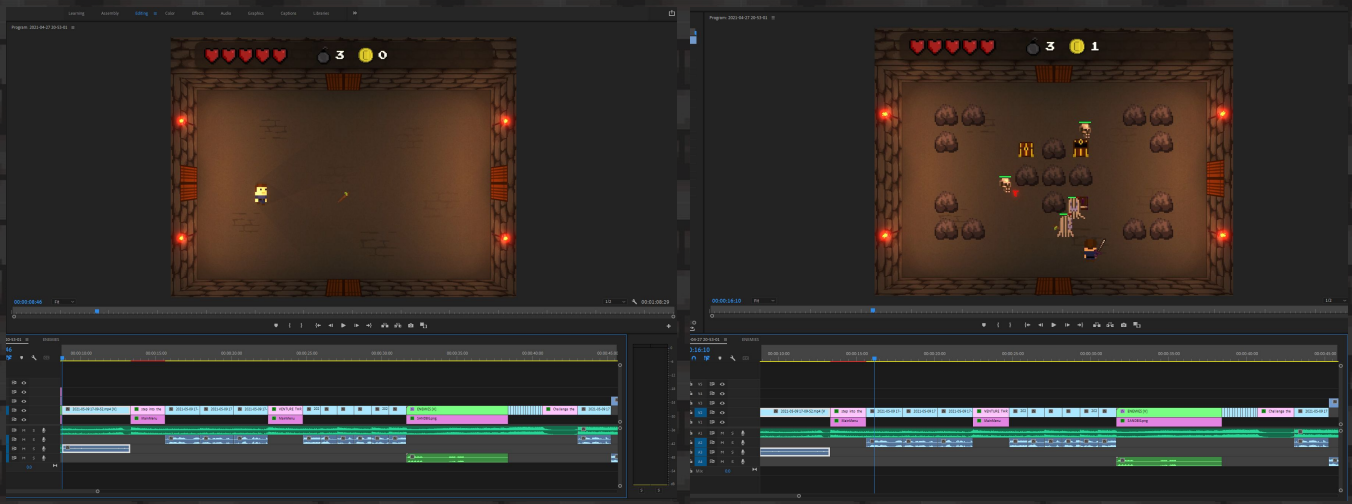
Record of Production



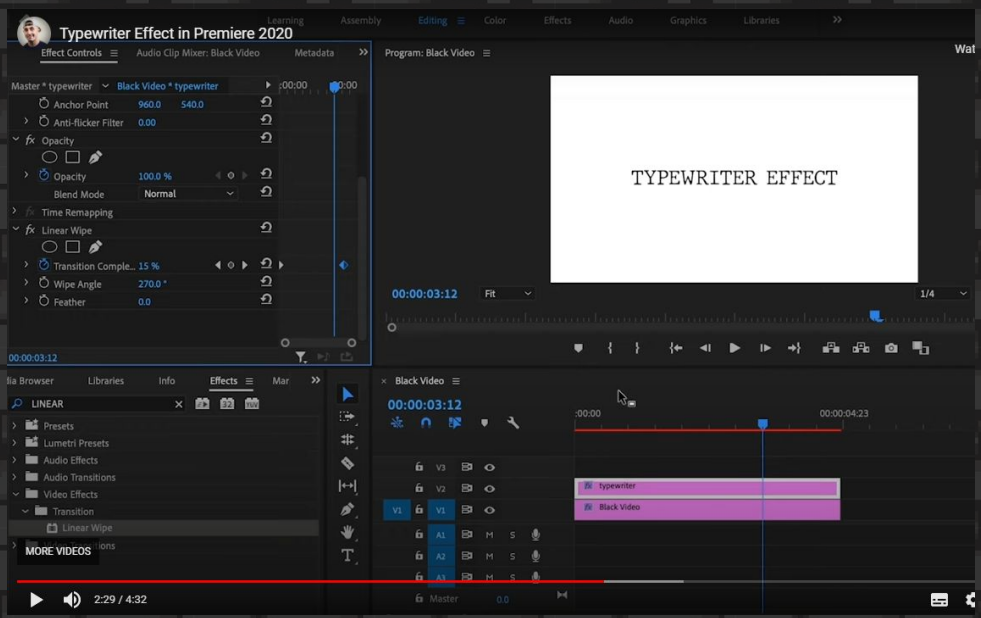
I repeated this process several times to record around ten 1-minute long gameplay clips that showed off pretty much every aspect of the game including levels, items, weapons and monsters. After I gathered all my clips, I imported them into my video editor of choice, Adobe Premiere Pro. I placed all of my clips into the timeline and cut and sorted them in the order they would play out. This order was based on the storyboard that I created during my idea generation.



Record of Production



Then, I added in a title screen at the beginning of the video displaying the name of the game “Dungeon of Death”. Originally, this title screen was static and quite boring, so I decided to follow a tutorial I found on YouTube that outlined how to create an animation that showed the text being typed out across the screen.



https://www.youtube.com/watch?v=wtNYGZXqTJo

I incorporated this effect into my own video, combined with a button click sound that I recorded from my own keyboard, to create my own Typewriter effect inside Premiere Pro.

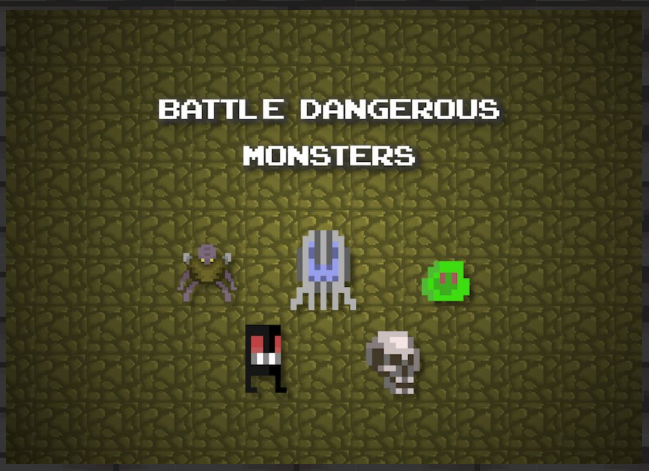
Record of Production

Playing back this simple skeleton prototype of my trailer, I realised it was too boring and needed a bit more of a storyline to hook the viewer and engage them further. My video follows the protagonist as he defeats monsters and bosses to claim the dungeon. However, a first time viewer would not realise that this is the point of the game, so I created several more title screens displaying text that helps immerse the viewer.

I followed the same few steps in creating these additional title screens and transitioning them into the rest of the clips that I recorded:

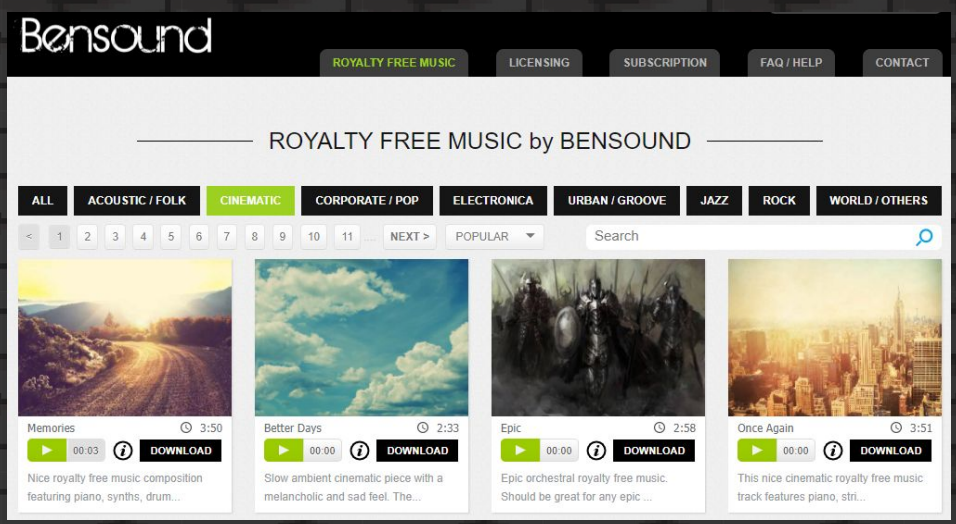
1. **Create a base template with the brick background used in the Main Menu**
2. **Insert a small text box in the center of the screen**
3. **Type the desired text into the text box**
4. **Animate the box to enlarge slowly over a few seconds, as the background image fades into black**
5. **Add transitions to the start and end of the title screen, making it play fluidly with the rest of the video**
6. **Repeat steps 2-5 for any additional title pages**

Record of Production



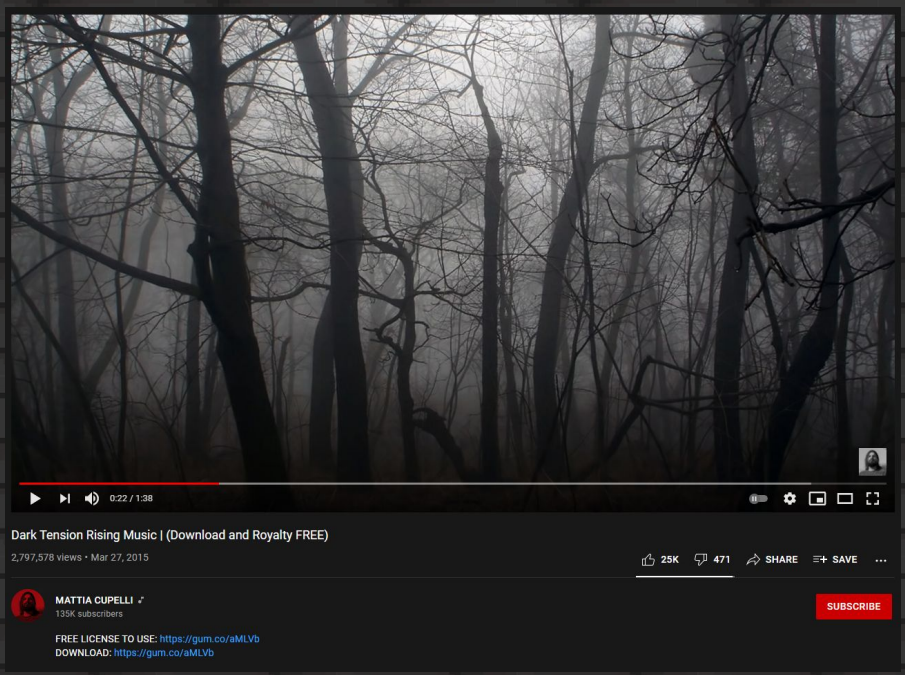
The inclusion of these title screens add more insight into the trailer while also giving the viewer a greater sense of what the video game is about and making it more enjoyable to watch.

Now that most of the visual elements of the trailer were completed, I needed to focus on the auditory elements such as music and sound effects. These factors are just as important as the visual aspect of the video and can ultimately make or break the final product. For the main background music, I knew I wanted an intense piano/orchestrated melody with a crescendo.



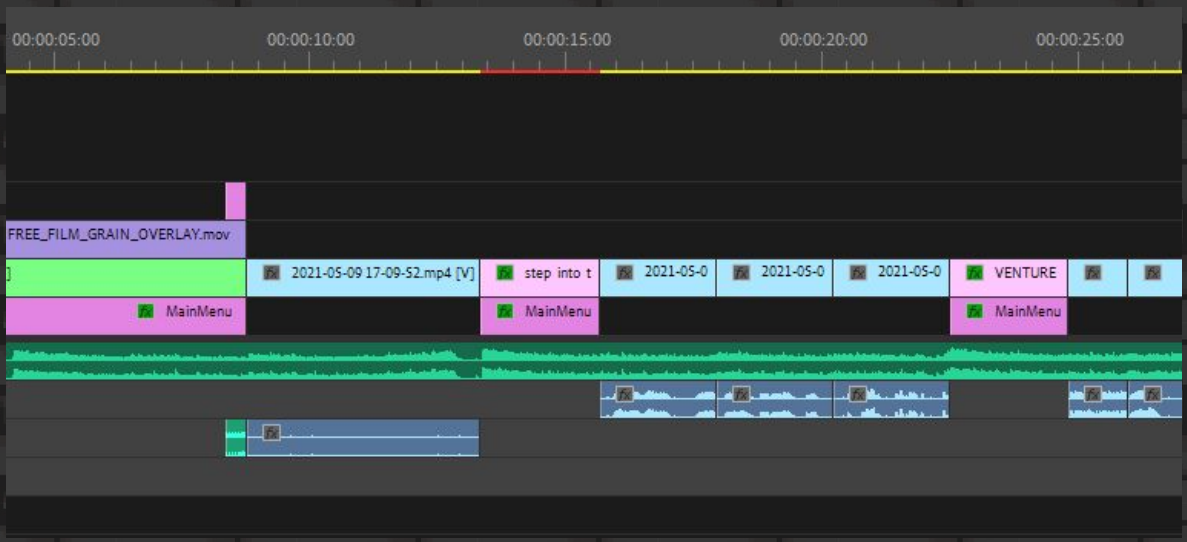
Record of Production

I explored websites such as Bensound and Freesound.org that offer countless royalty free music tracks and sound effects, none of which had the right sound that I was going for. Finally, I browsed through Youtube for dramatic royalty free music and found the perfect track that I was looking for. Included in the description was a download link and proof that the track was indeed Royalty Free.



https://www.youtube.com/watch?v=-zvQoPyY2XE

I imported this music track into Premiere Pro and listened to all of it, to see if it matched the gameplay. Unsurprisingly, the melody was perfect and only required minimal cuts to make it suitable for my trailer. I cut down the length of the track and placed it at the start of my video.

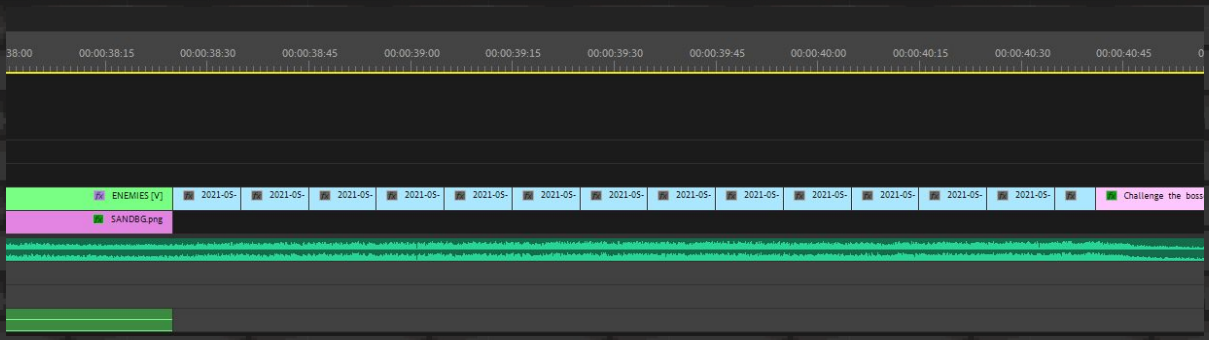


Record of Production

Next, I took every clip in the timeline and arranged the transitions to match the drum pattern of the music track. Not only are the clips visually aesthetic now, but they also match the beat perfectly and are cut exactly where they are supposed to be.

I noticed a section towards the end of the music track where the drum pattern speeds up significantly during the crescendo, right before the beat drops impactfully. I decided that the increase of speed was perfect to create an extremely fast quick-cut segment of video clips that aligned with each bass drum hit. The job of cutting these clips to the perfect length and timing was an extremely precise process, and had to be done frame by frame. I did this by:

1. **Count exactly how many bass drum kicks are present in the sequence**
2. **Play the sequence back slowly and add markers to where every drum hits**
3. **Import the video clips into the sequence and fit them to the markers**
4. **Play the sequence back and make small adjustments**



I absolutely loved this small but important addition into my trailer as it helped so much in varying the speed and tone of the video, instead of it remaining plain and predictable. It allowed the video to start off slow and mysterious, slowly rise with the crescendo of the backing track and then taper off towards the end.

After this sequence and the peak height of the music track, I added a final title screen introducing the two bosses of my game, the Gelare Boss and the Sarco. Then I imported more clips of these boss fights and repeatedly switched back and forth between the clips for added tension.

WHS - Workplace Health & Safety

Work Health & Safety (WHS) involves the **assessment and mitigation of risks that may impact the health, safety or welfare of those in a workplace.** To reduce potential hazards and risks, I made sure to enforce safe working practices throughout the production of my Major Project.

HOME WORKSPACE

My home workspace poses less risk in comparison to the workspace at school, since it is much smaller and therefore easier to manage.

In my home setup, I have tucked my cables behind my desk and utilised a bluetooth speaker and wireless internet adapter. I have also kept my table empty especially from food and drinks, to minimise as much potential hazards as possible. I kept my blinds open and lights turned on to make sure my room is well lit. The door and windows also remained open while I worked to ensure good ventilation.

SCHOOL WORKSPACE

The school workspace, as a classroom, poses several health and safety risks that must be assessed in order to perform productively and efficiently during the creation of my Major Project.

At my school workspace, I ensured that I placed my bag at the front of the classroom every lesson. My teacher made sure that the windows were open and the temperature was controlled through an air conditioner. I kept all hazardous material such as food and drink away from my desk and made sure to tuck all cables away behind my monitor. The classroom also has a fire extinguisher near the front door at all times in case any fire breaks out.

WHS - Workplace Health & Safety

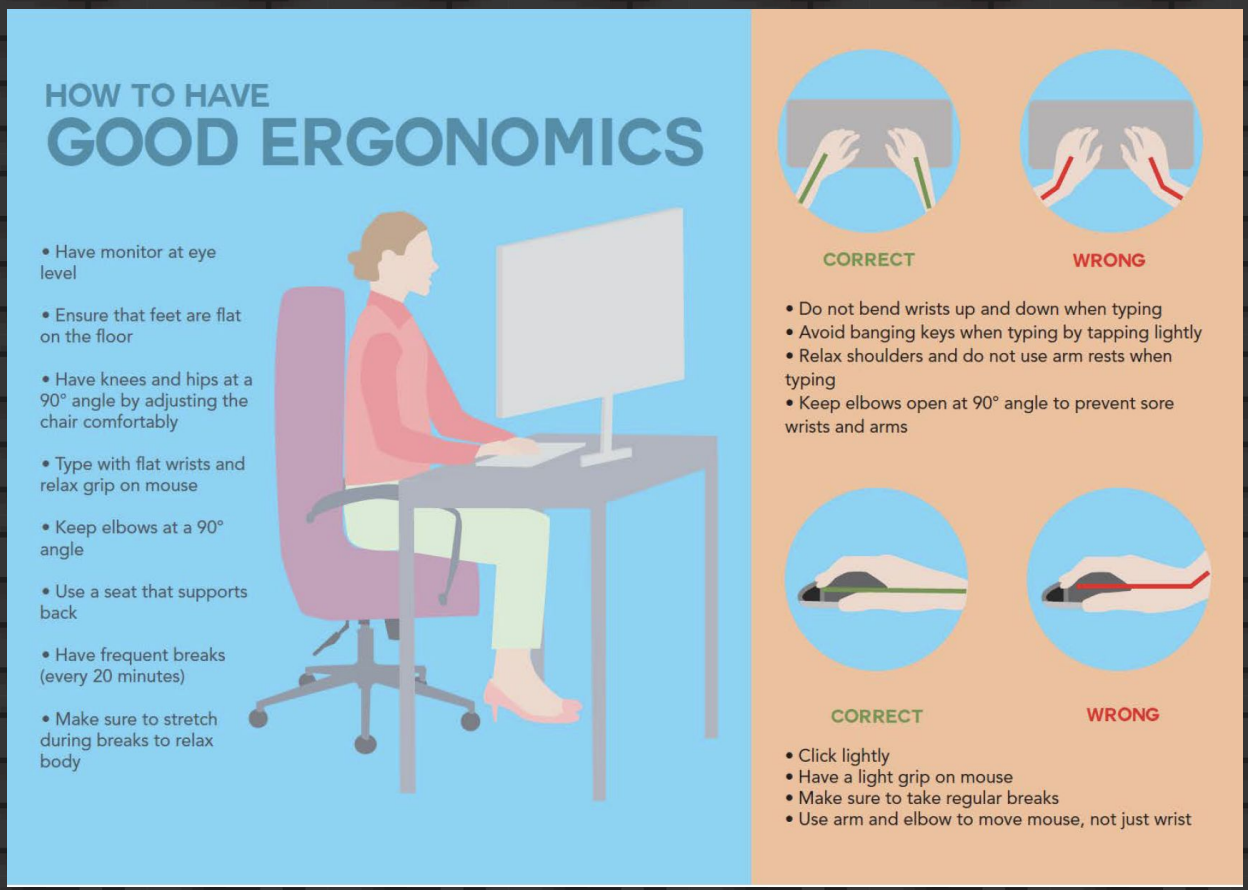
As my project consisted of mainly computer based components, most of my time was spent at a desk sitting in front of my laptop or a computer. I have outlined the potential hazards and accompanying solutions which I undertook below:

POTENTIAL HAZARD: REPETITIVE STRAIN INJURY (RSI), BACK & NECK PAIN, EYE STRAIN & HEADACHES

Both workspaces could result in back and neck pain due to low chairs, badly placed equipment and bad posture. Extended sessions in front of a computer could also lead to eye strain and headaches. As well as this, given the nature of my project, I often make small repetitive movements and this could cause RSI.

SOLUTION:

The key to reducing wrist, finger, neck and back pains is ergonomics. Ergonomics focuses on creating comfortable environments for people. I followed the below diagram, ensuring that I positioned my workspaces appropriately and took regular breaks to protect my health and safety.



WHS - Workplace Health & Safety

POTENTIAL HAZARD: UNORGANISED WIRES & OVERLOADED POWERBOARDS

Unorganised or exposed wires and having too many devices plugged into a single outlet poses many risks. Wires are a tripping hazard and they, as well as an overloaded powerboard, could cause fires and electrocution.

SOLUTION:

I ensured that any cables used were not creating a tripping hazard by positioning them in the corner of my workspaces, away from any walking paths. I also organised and managed wires by tucking them behind tables and monitors. Where possible, I tried to avoid using any cables (e.g. by using a bluetooth mouse). I also ensured that I did not eat or drink around the computer incase I spilled something. A fire extinguisher is also available in case of emergency in the classroom together with evacuation guidelines.

POTENTIAL HAZARD: POOR LIGHTING & VENTILATION

Poor lighting can result in eye strain when reading monitors and/or documents. Long term effects can result if this risk is not monitored. Poor ventilation and lack of temperature control can also create uncomfortable work conditions which then affect productivity.

SOLUTION:

While working on my project, I made sure that blinds were open and that overhead lights were switched on if needed. I also made sure that windows and doors were open in most circumstances to ensure adequate air flow and ventilation. As the windows at school do not open, the air conditioner was used to control temperature and provide a more comfortable environment to work in.

Final Evaluation

I believe that the video game “Dungeon of Death” and the trailer that I created for my Major Project was a huge success and effectively fulfilled the purpose it was intended for. It encapsulates my love and passion for retro arcade gaming immensely, bringing back many nostalgic memories during the planning and production process. Not only would it bring back similar great feelings for others that have played my game, but it does so in an entertaining and immersive way. Throughout the ups and downs of the year, I persevered through my work and put as much skill into my work as I could, although there are more things that I could have added, if I had some more time.

The production of my Major Project went extremely well throughout the year, as I planned all elements of my project accordingly and managed to stick pretty closely to my timeline. My project did successfully match my intentions, being a fully playable and entertaining video game with all the elements of a small game title. Although I doubted my artistic abilities at the beginning, through extensive sketching and prototyping, I created an art style and look that I am very happy with. I think the 8-bit retro graphics served a huge role in making the player feel as though they are back in the 90’s era of gaming. I made sure to research as much as I could on Unity and coding through sites such as YouTube and official documentation, so as to not jump straight into programs I knew nothing about. Having almost **3900 lines of code and 446 files altogether**, I am very happy that I organised my project neatly into dozens of marked folders and code classes, making navigation and production highly efficient.

I ran into a few problems during the development of my Major Project, mainly due to the time constraints I was given. I planned to add a lot more elements into my video game, especially the opening animation, but I couldn’t do this because I was unable to learn Adobe Animate in the time I had. Although I feel like the video trailer was enough, I couldn’t express the back-story of my video game the way I wanted, and largely had to scrap the storyline altogether. The COVID-19 pandemic, although not affecting my project directly, did take a large mental toll on my motivation and work rate. Being at home all day took away a lot of my friend support and prevented a lot of teacher and peer feedback. I feel like I could have worked a lot harder during this time and utilised the extra free time I had to my advantage. However, I stayed optimistic, overcoming these problems by making sure I set a daily work schedule, talking to friends online and taking enough breaks throughout the day. I did have to scrap features from my game such as the opening animation and customisable armour/accessories, but I made sure the rest of my project was as perfect as it could be.

In conclusion, I am satisfied with the quality and outcome of my Major Project. I learnt countless new coding, design and time management skills that will be greatly beneficial for my future. By constantly pushing myself, I created a project that I was passionate in and that achieved the objective of my Statement of Intent. I hope I can bring this passion to other people, especially the younger generations, and spark their love for gaming and game development in the same way that my game inspirations did for me.