The Lost Ark

Liam Rickman | 1902527

Contents

This portfolio shows my design and development process I took whilst completing my DES205 final project.

The contents are as follows:

Brief & Concept

Chosen Brief

I chose to follow the adaptation brief for my project as it allowed me to base the project around a particular story and theme whilst having lots of reference imagery and information to support it.

Concept

I wanted to create a range of traps and obstacles to include in my level that the player would have to traverse. This would allow me to showcase and learn a variety of Unreal Engine features while creating the traps.

After deciding I wanted to create various traps, I started looking at films and TV series that would fit this theme. One series I looked into was Indiana Jones, notorious for temples and booby traps Dr Jones had to avoid. I researching each film in the series, taking notes on the trap types and themes to try narrow down my choice to one particular film. Eventually I decided on Raiders of the Lost Ark, and it's opening scene where Indiana Jones is investigating a Peruvian temple, searching for a legendary golden idol. This section of the film features a wide range of traps in the temple that could be adapted into an interesting level.

The traps will use triggers, scripting and animation, there will be collectable objects such as the golden idol and user interface elements will display health and other necessary information.

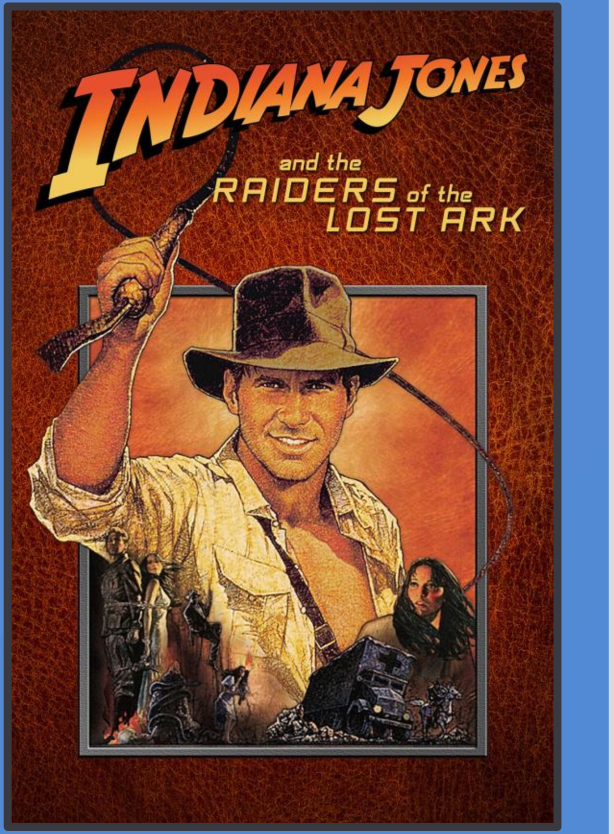


Image Credits: https://filmconcertslive.com/movies/raiders-of-the-lost-ark/

Research

Indiana Jones Series

The first research I did was looking at the Indiana Jones series. I began watching through the films, looking for sections that included traps and interesting locations that could be adapted into game mechanics and a level.

- Raiders of the Lost Ark: Boulder chase, wall spikes, floor puzzles, rope swing, light traps
- Temple of Doom: Bridge crossing, minecart chase, conveyer belts
- The Last Crusade: Fake floor across chasm, crouch to avoid blades, choosing the correct grail

After this research I decided to go further and adapt Raiders of the Lost Ark. Compared to the other two films, it had a wider range of trap options that I could use in the level, as well as an obvious goal for the player in the form of a golden idol hidden in the temple.

Raiders of the Lost Ark

Now that the exact film was chosen, I started researching further into the types of traps and sections I wanted to take further in my level. Some notable areas were:

- Tunnel 1: Cobwebs, no light except from the entrance, cobwebs halfway through, large open area before end.
- Light trap room: Small room with gaps in roof where light passes through onto the floor.
- Rope Swing: large pit with a rope swing to swing over the gap.
- Floor Puzzle: Wide room with pattern on the floor, stairway to idol room at end.
- Idol Room: Large circular room, floor puzzle continues with the idol sitting on a platform.
- Closing Door: Door that slowly closes trapping the player behind it.

I now had a list of traps that I could take further and design to use in my level.

Research

4 Step Level Design (Game Makers Toolkit – Super Mario 3D World's 4 Step Level Design)

In an episode of Game Makers Toolkit, the concept of four step level design was discussed. The game talked about was Super Mario 3D world where the four step design is used heavily to teach the player about the game mechanics.

The four steps are: Introduce, Develop, Twist and Conclusion.

Firstly each trap will be introduced to the player with minimal risk to player health or lives. This allows the player to learn the mechanics of the trap in a relatively safe environment. Then they can continue to develop their skills by traversing the traps, increasing the difficulty as they progress through the level. Next a twist will be introduced to change up how the trap works, this could be something simple such as changing speeds of moving objects or something more complex like making a trap move instead of being stationary. Finally the mechanic will be concluded where the player can show off all that they have learned about the particular mechanic.

This design process should work extremely well for the traps I plan to create as it will teach the player the mechanics and allow them to get used to them, developing their skills along the way without ever getting boring or repetitive. This style of design will heavily influence how my level is developed due to my focus on the different traps I plan to create.

Player Engagement (Game Makers Toolkit – How to Keep Players Engaged (Without Being Evil))

Keeping the player engaged will be key in creating an interesting game for the player to experience. Varying gameplay types is essential to keep the player engaged but not overwhelmed. For my level I can vary the style and amount of traps to alternate between the slower paced puzzle style traps and faster paced reaction based traps. This should help keep player engagement high without making the game too intense and exhausting for the player. Uncharted does this by mixing in exploration sections in between intense firefights and action scenes.

Research

Health Systems (Game Makers Toolkit - How Games Do Health)

I needed an appropriate health system for my level as due to the trap nature of the level, there would inevitably some accidents where the player will be hit by a trap and need to take some form of damage. The Game Makers Toolkit video discusses some of the common practices that video game companies use for health systems in games.

Many modern first person shooter games use a regenerating health system where the player recovers health by staying out of action for a few seconds. This is a simple but very unrealistic mechanic that game designers seem to default for these types of games. However I do believe it works well for the fast paced shooters where you the player is intended to be having action moments very often. If the player had to stop and heal using a bandage or med kit, this would put them out the action temporarily and leave them vulnerable to an attack by another player or NPC.

Adventure games use a different mechanic by using health kits to heal the player. Dark Souls for example requires you to manually heal which may affect your later choices as you will have less healing items for future fights. The way the game lets the player decide when and how to heal will likely influence their playstyle until they have healed up again. A game such as Doom (2016) actually encourages the player to get aggressive when low on health by dropping health kits after finishing an enemy with a melee attack. This is the opposite of a standard reaction to having low health where you back off and play more defensive.

My traps are designed to kill the player instantly, however I do not want the player to be reset back to the beginning of the level every time they die as this my discourage players from continuing the level if they get stuck at a particular section and have to keep repeating the entire level to reach it again. To combat this, a three point health system would work well with the player losing a health point when they are killed by a trap. Combined with checkpoints, when the player loses health they will be teleported back to the last checkpoint they passed. If they run out of health, this is when the level will reset as they have failed. Health packs will be scattered around the level and can give back lost health points, up to a maximum of three.

Research

Flusher – Use of Colors in Game Design and Level Design

Appropriate use of colour in games is very important to help new players distinguish between places of interest and areas of avoid. This video by Flusher shows how to use colour in game design and level design to accomplish this.

In general, red and orange tones re bad and should be avoided as these usually convey enemies and other damaging areas.

Green colours are commonly associated with healing items, however sometimes reds are used instead for healing potions etc.

Collectable items such as coins or stars often use a yellow colour. For example in the Super Mario Games, yellow is used for the coins and show that they are items of interest and should be collected.

In my level, I plan to use green to showcase health packs and golden yellow for the idol and gold coins. If any future traps I develop feature AI characters or obvious damaging areas, I will try to incorporate red tones to help portray damage to the player.

Game Developers Conference – Level Design Workshop: Blockmesh and Lighting Tips

Appropriate lighting is also extremely useful when combined with colour to help portray areas of interest. A talk from the Game Developers Conference discussed good lighting practices to help lead the player and guide them through the level.

Areas with bright vibrant lighting are likely to catch the players area and draw them in. With this you can tier different areas of interest by changing how bright the lighting is in each area. The brighter the lighting the more important it is. Darker areas also create hidden areas to create mystery and highlight more dangerous areas. Any areas there are encounters with enemies or traps should be lit enough so that the player can see enough.

Moodboards

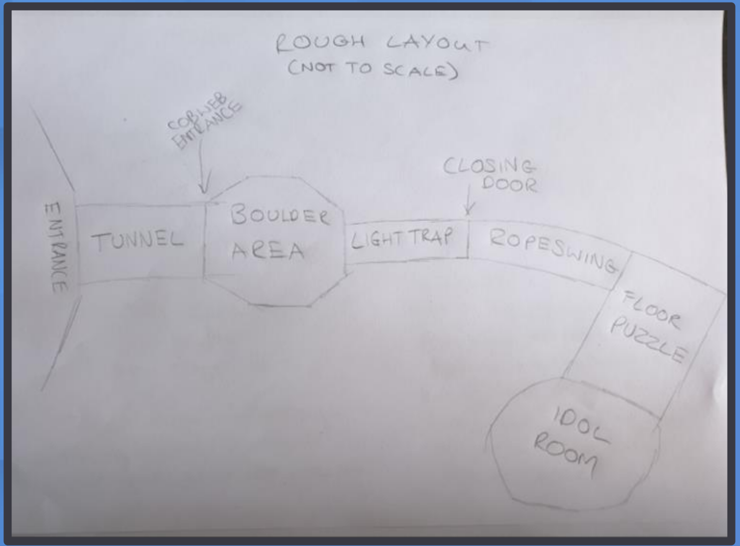
Raiders of the Lost Ark

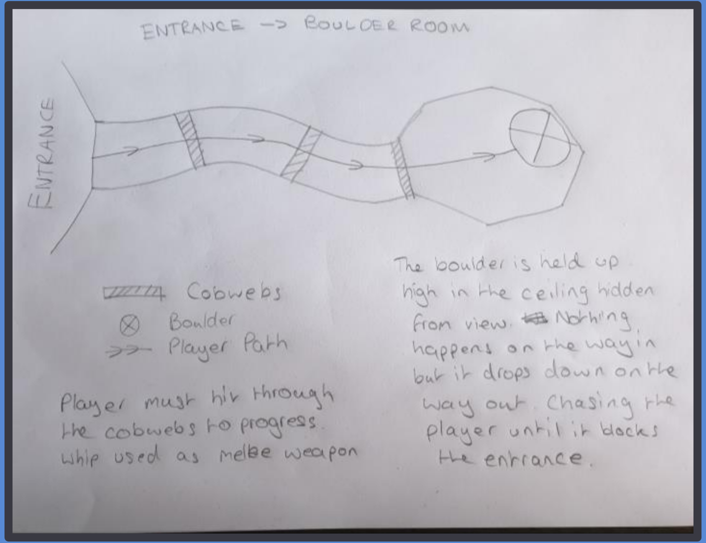
Whilst watching the film, I took various screenshots of some key scenes and sections of the temple that would help me later on. This moodboard will help me decide the style of the level as well as how the traps will look and work.

Image Credits: *Raiders of the Lost Ark*. 1981.

Pre-Production | Layout v1.0

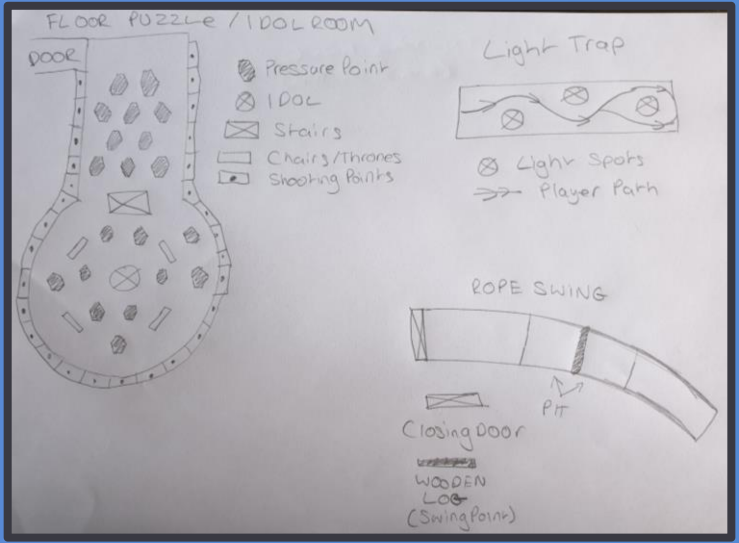
Initial layout





Player must hit through the cobwebs to progress. whip used as melee weapon

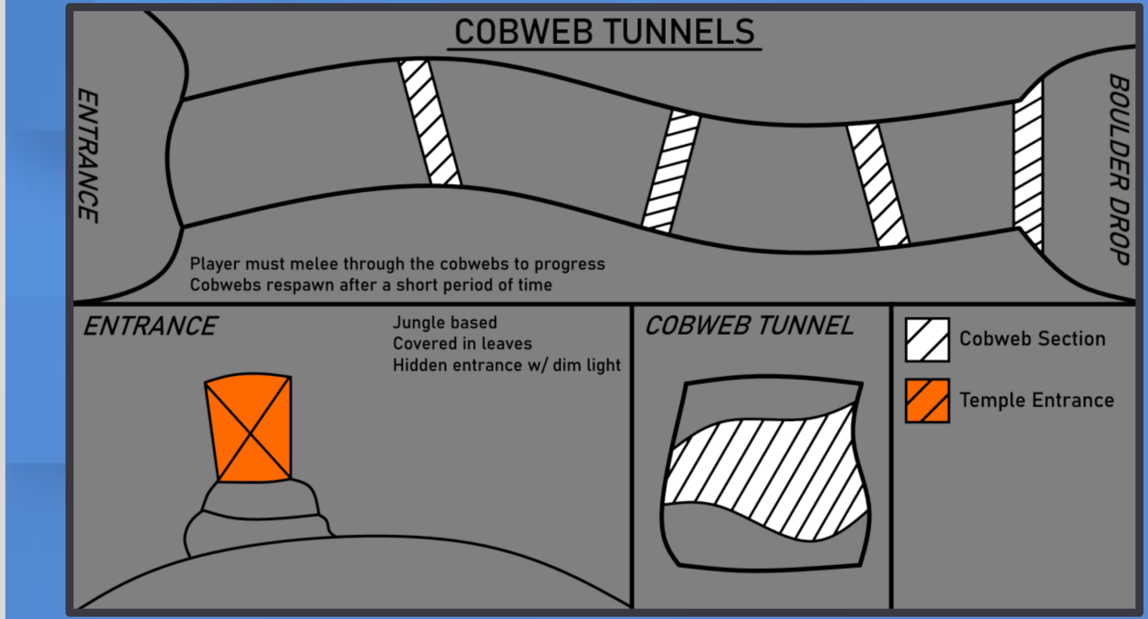
The boulder is held up high in the ceiling hidden from view. ~~[illegible]~~ Nothing happens on the way in but it drops down on the way out. Chasing the player until it blocks the entrance.



In my initial sketches, I took inspiration from the film itself and created a film accurate layout. I knew this would not make the greatest game level but wanted to gather the types of rooms and traps that the film featured. This would allow me to take it further and create my own layout that works in a game environment.

Pre-Production | Layout v1.0

Cobweb Tunnels

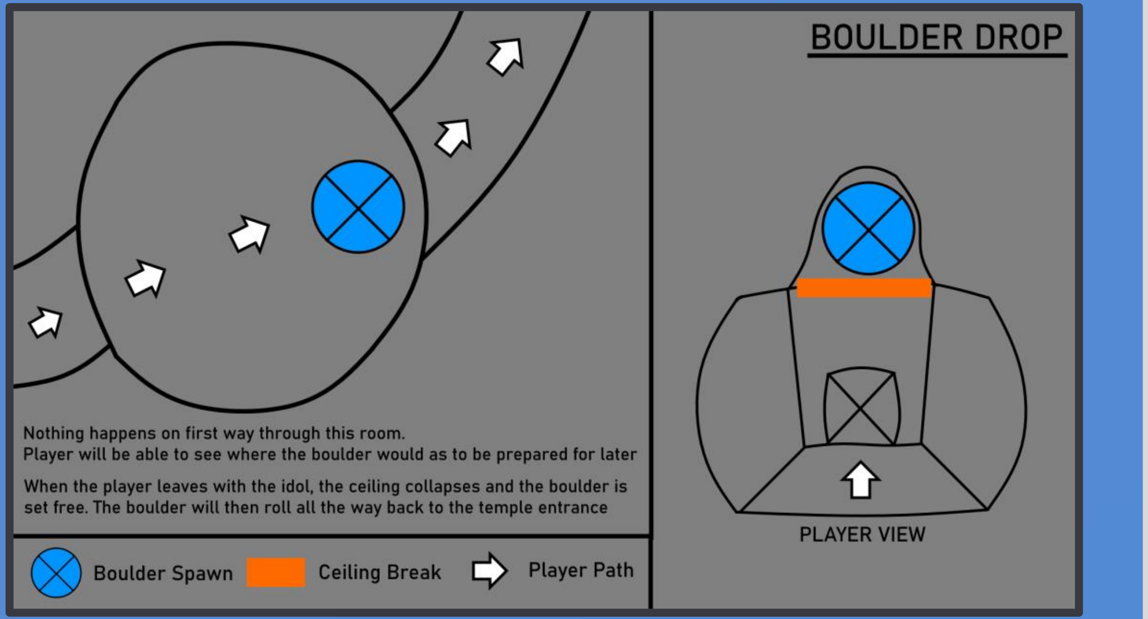


Player must melee through the cobwebs to progress
Cobwebs respawn after a short period of time

Jungle based
Covered in leaves
Hidden entrance w/ dim light

The player will be introduced to the cobwebs early on and will not be difficult to pass through, only requiring a button press to destroy. The cobwebs will mostly hide secret areas or block the player from progressing quickly.

Boulder Drop



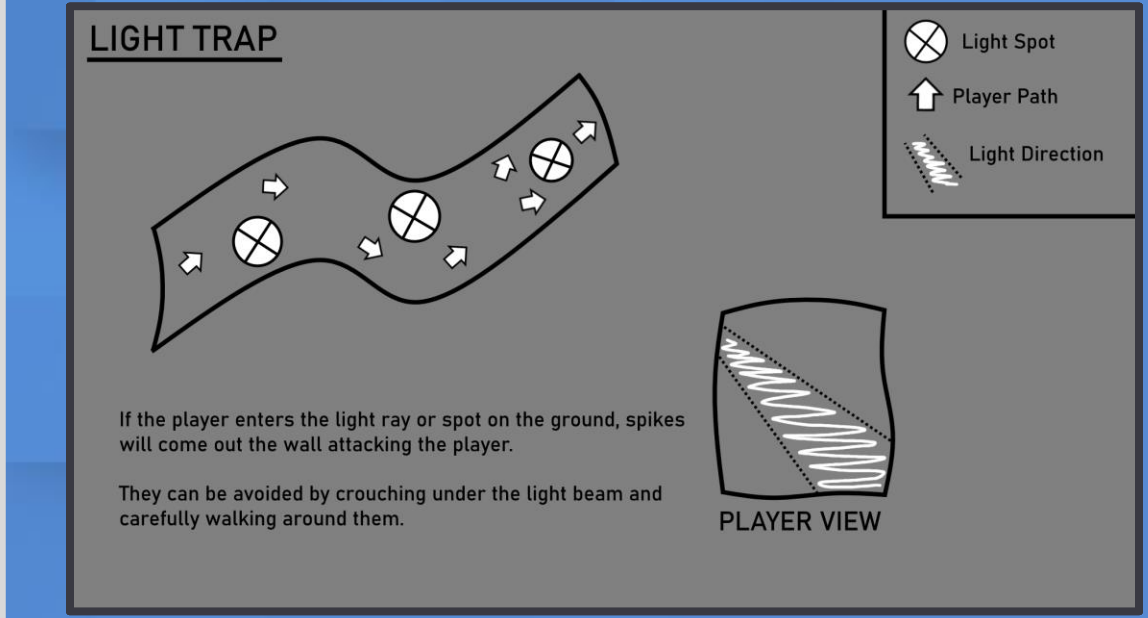
Nothing happens on first way through this room.
Player will be able to see where the boulder would as to be prepared for later

When the player leaves with the idol, the ceiling collapses and the boulder is set free. The boulder will then roll all the way back to the temple entrance

The boulder drop room will be next in the temple and nothing happens on the way in. Players who explore the room will be able to see the boulder hiding in the ceiling. When the player leaves the ceiling collapses and the boulder will roll back to the temple entrance.

Pre-Production | Layout v1.0

Light Trap

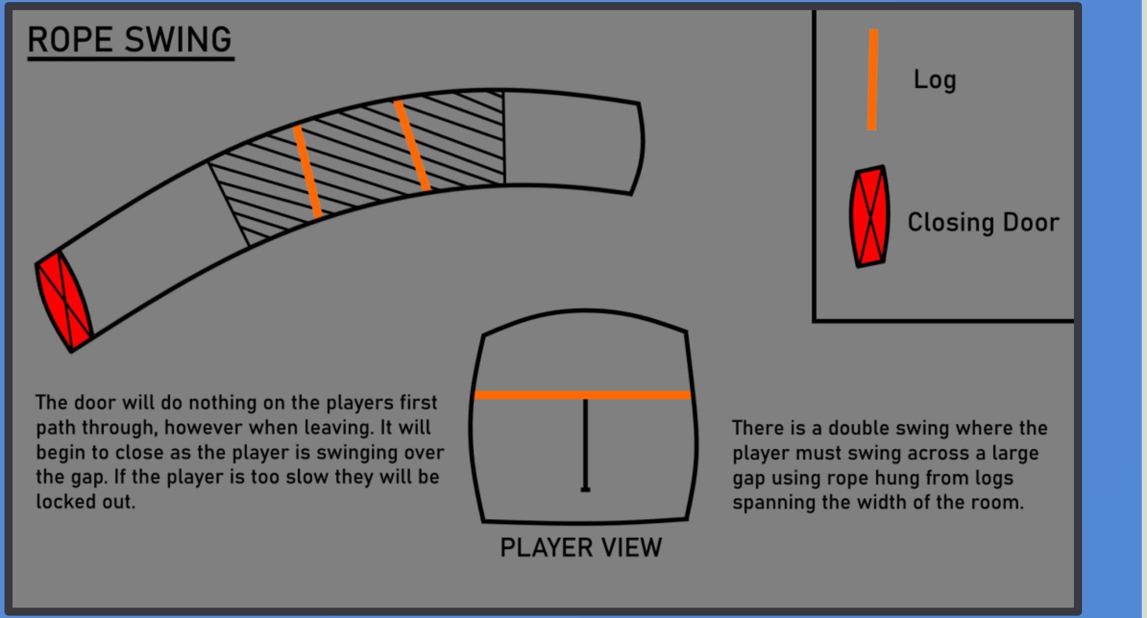


If the player enters the light ray or spot on the ground, spikes will come out the wall attacking the player.

They can be avoided by crouching under the light beam and carefully walking around them.

When in the light trap room, if the player enters the light, spikes will be fired at the player. The light can be avoided by crouching and carefully walking around them.

Rope Swing



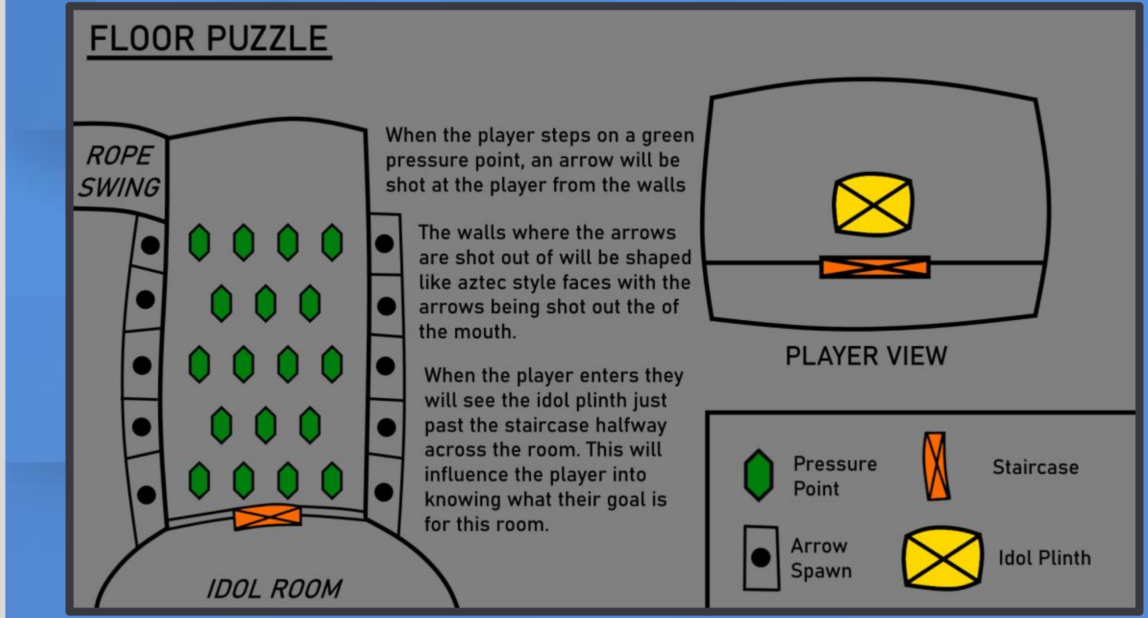
The door will do nothing on the players first path through, however when leaving. It will begin to close as the player is swinging over the gap. If the player is too slow they will be locked out.

There is a double swing where the player must swing across a large gap using rope hung from logs spanning the width of the room.

The rope swing will have a log in between the walls with a rope attached. The player must jump and grab the rope and swing across. The door will close on the players way out to ensure the player goes quickly across the gap.

Pre-Production | Layout v1.0

Floor Puzzle



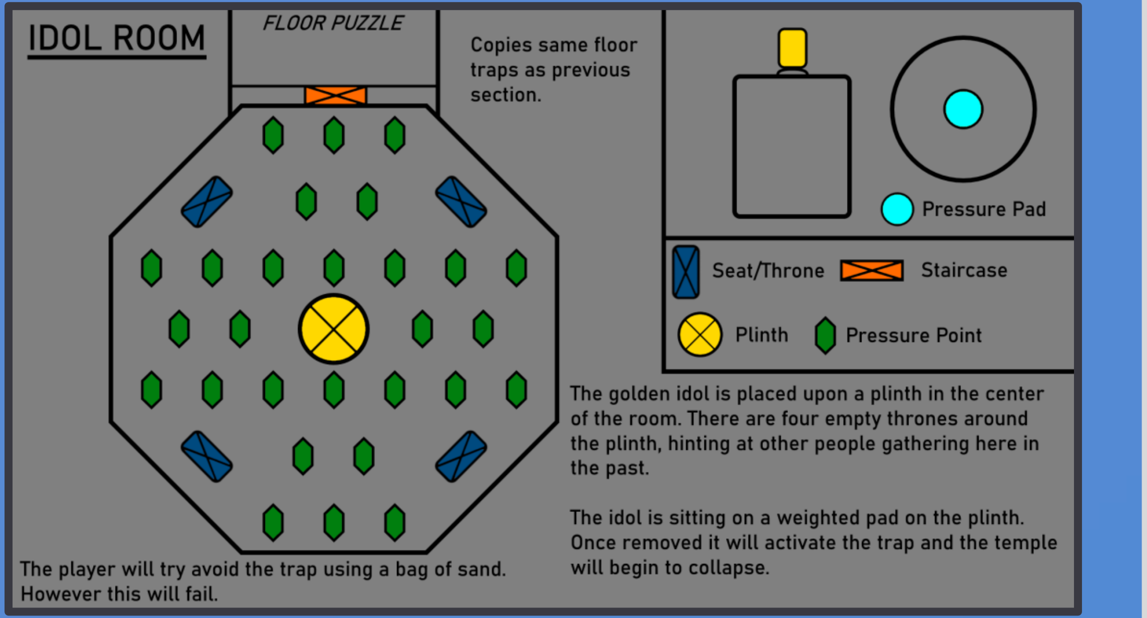
When the player steps on a green pressure point, an arrow will be shot at the player from the walls

The walls where the arrows are shot out of will be shaped like aztec style faces with the arrows being shot out the of the mouth.

When the player enters they will see the idol plinth just past the staircase halfway across the room. This will influence the player into knowing what their goal is for this room.

The floor puzzle room will have plates on the ground that if stood on, arrows will be fired at the player from the walls. The player can see the golden idol from this room and be tempted to get their quickly.

Idol Room



Copies same floor traps as previous section.

The golden idol is placed upon a plinth in the center of the room. There are four empty thrones around the plinth, hinting at other people gathering here in the past.

The idol is sitting on a weighted pad on the plinth. Once removed it will activate the trap and the temple will begin to collapse.

The player will try avoid the trap using a bag of sand. However this will fail.

In the idol room, the floor puzzle continues. In the middle of the room there is a platform with the idol on it. Once collected the temple will start to collapse and the player must hurry out.

Pre-Production | Traps

After some research I decided to rework some of my traps to fit the 4 step level design structure.

Cobweb Tunnels

The player will be introduced to the cobwebs early on and will not be difficult to pass through, only requiring a button press to destroy. The cobwebs will mostly hide secret areas or block the player from progressing quickly.

Floor Puzzle

The floor puzzle will be introduced in a small room with a 3x3 grid of pressure points that will activate the traps. It will be obvious that these should be avoided. The trap is developed by making the placement of the traps more scattered and random. The twist will be including these in the idol room making it harder to reach the idol.

Swinging Logs

The logs will swing in from one side when the player gets close. They are developed by having the logs go in multiple directions and the twist will be increasing the speed the logs swing at.

Light Trap

The light trap will be introduced using a beam of light shown on the floor of the temple. There will be a skeleton in the light as a hint that it should be avoided. If the player enters the light, spikes will shoot out of the wall, killing the player. These are developed by making the light move around the room. The twist will be using a mix of both types together in one room.

Rope Swing

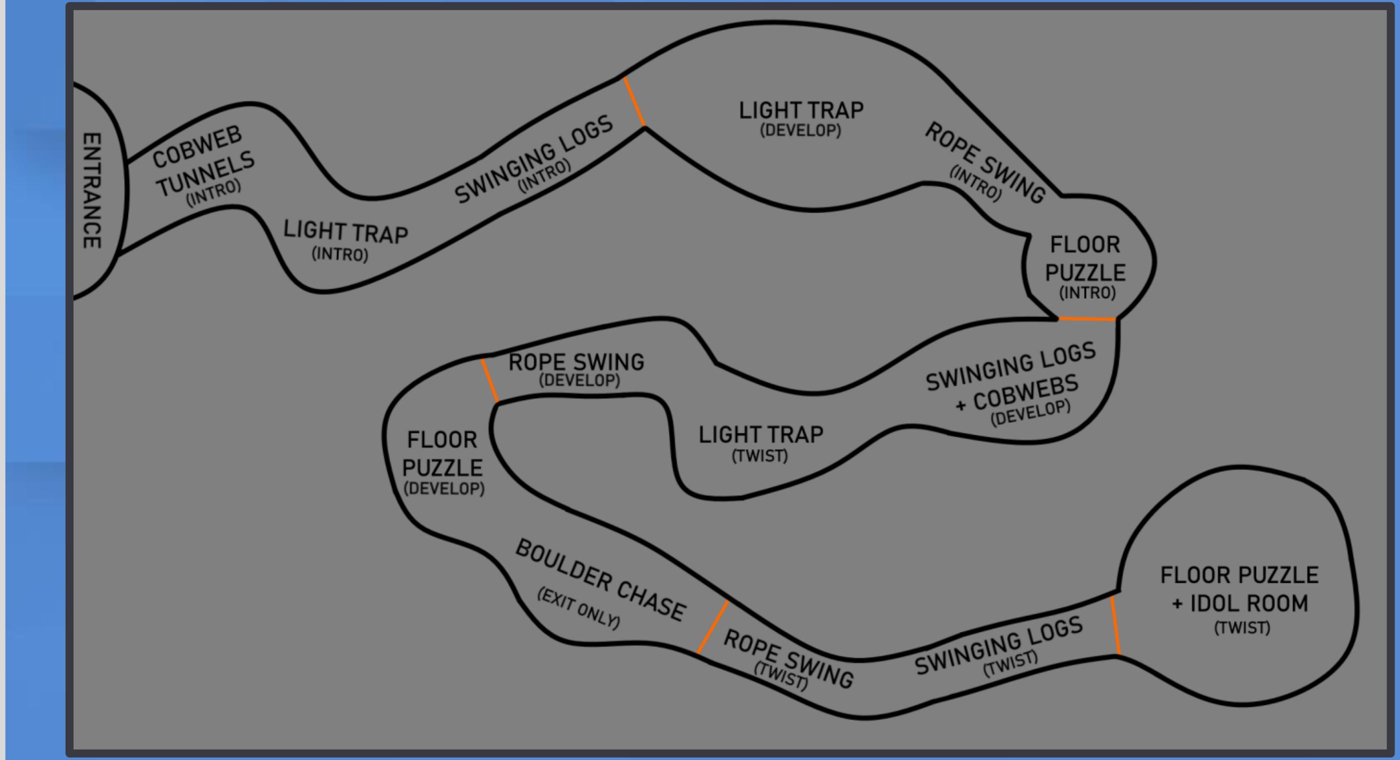
The player will be introduced to the rope swing by having to swing over a small gap. The trap will be developed with larger gaps to swing over. The twist is adding a gap where you have to use multiple ropes to swing over.

Boulder Chase

There is only one boulder in the scene. The boulder will only activate on the way out, but if the player pays attention they will notice the boulder on the way in. This will act as the conclusion for all the traps as it will put the player under pressure to escape letting them show what they have learned, combining their skills under a time limit to escape before being squashed.

Pre-Production | Layout v2.0

Layout v2.0 (Sketch)



As I have redesigned how the traps work, with them appearing in multiple sections, I needed to redesign the level to accommodate the increase in traps.

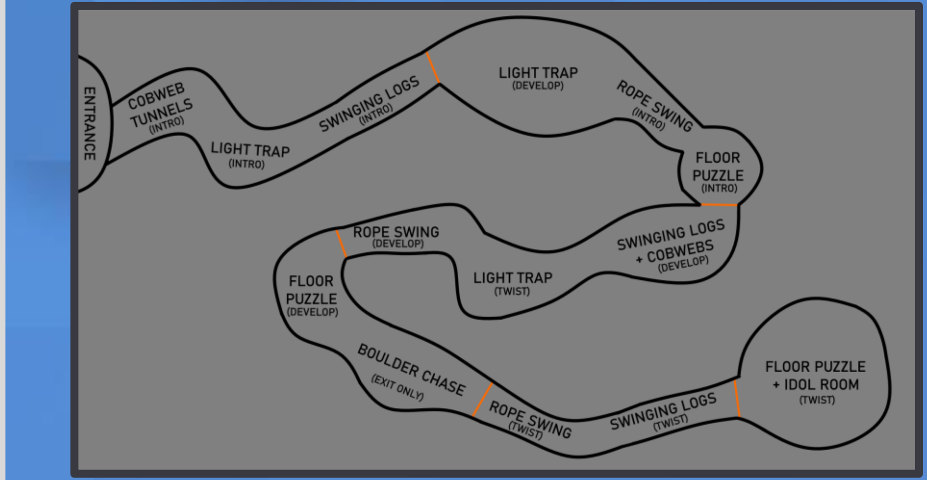
I decided to scrap the closing door trap as it felt out of place and overall felt a bit awkward for a game mechanic as if the player fails to get through the door in time.

This in turn has made the level longer and the changing traps will make the level more interesting for the player.

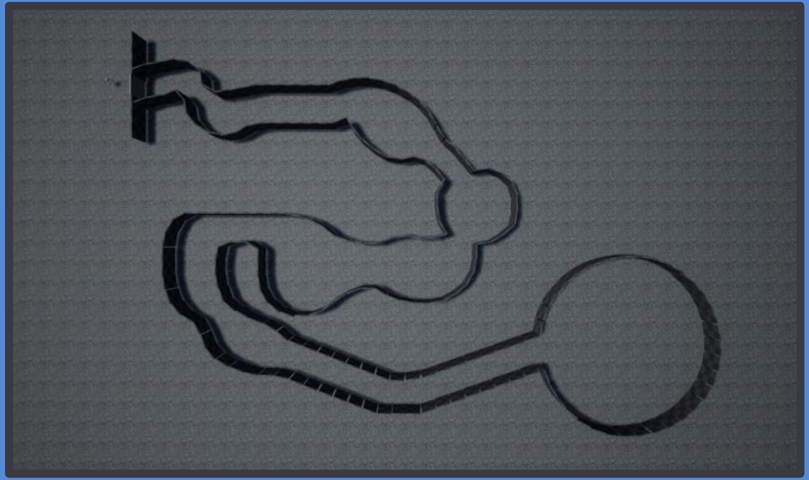
The traps now have multiple sections fitting the four step level design structure with Introduction, Develop, Twist and conclusion sections for each trap.

Comparisons | Level Layout

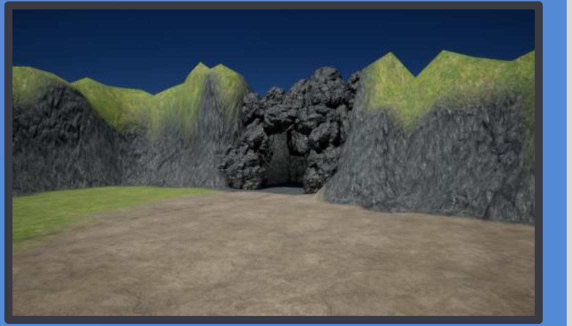
Layout v2.0 (Sketch)



Layout v2.0 (In-Engine)



Entrance



I blocked out the level in Unreal Engine using the geometry and landscape features available. While creating this layout the temple felt too small for a temple. I did increase the side of the walls, however this did not improve too much.

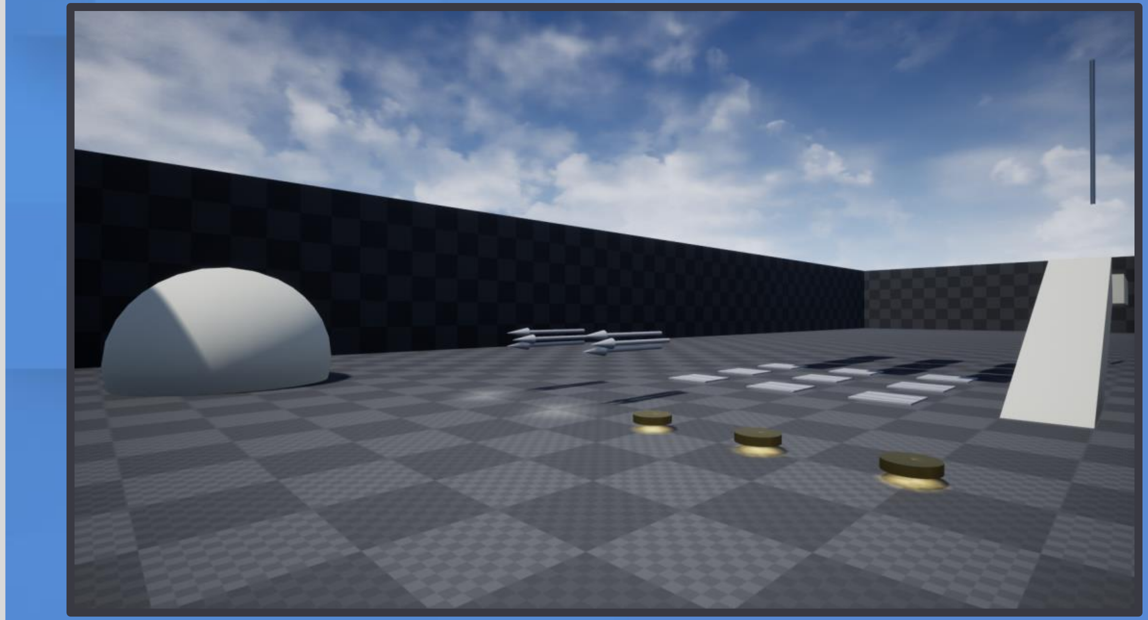
There was also no height difference throughout the level which felt unnatural, especially for a cave temple. I decided I needed to rework the layout to better fit the theme when recreated in Unreal Engine. However, before doing this I decided that I wanted to start creating the traps themselves to see how they would function in-game.

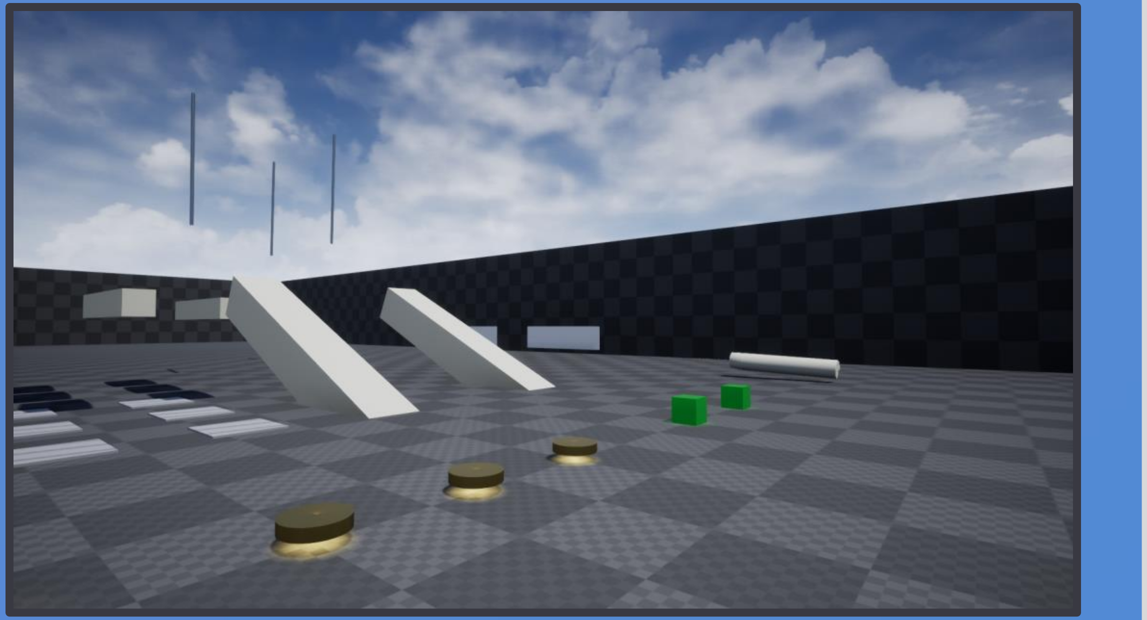
I created the outside of the temple using the landscape tools and some UE4 starter assets.

This came out a bit awkward and would need improving later.

Development | Mechanics Level

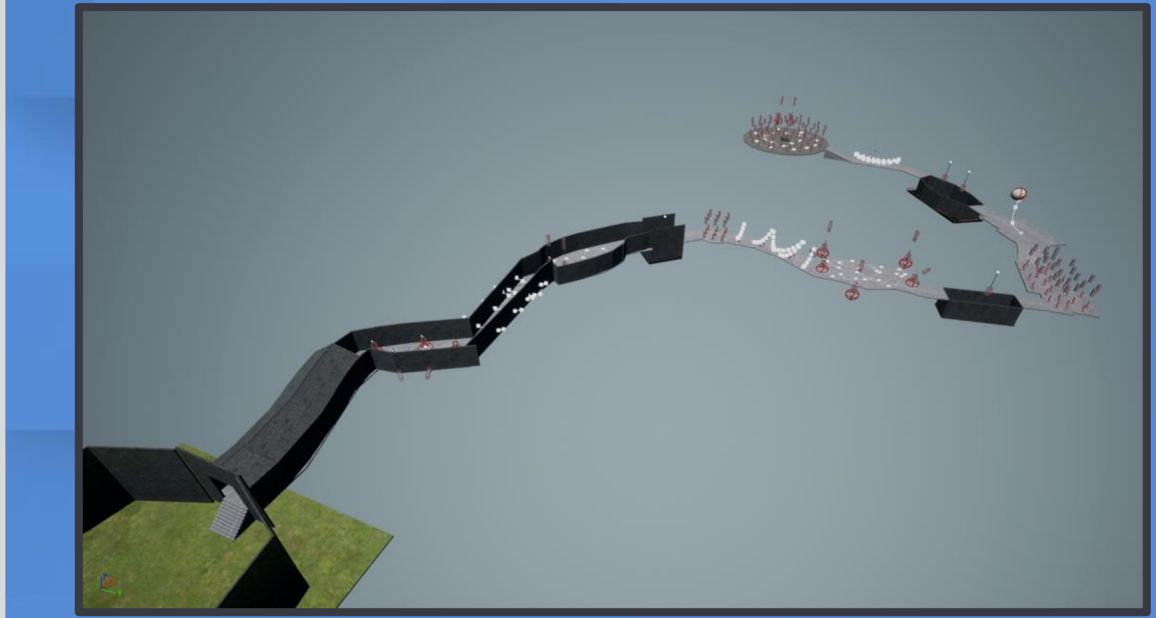
To create my traps, I started developing them in a separate level within Unreal Engine. This allowed me to test the traps in a clean environment, not worrying about how the level effects them. Here I created all the planned traps and mechanics I wanted to use in my level before going on further to design a layout that included them effectively.

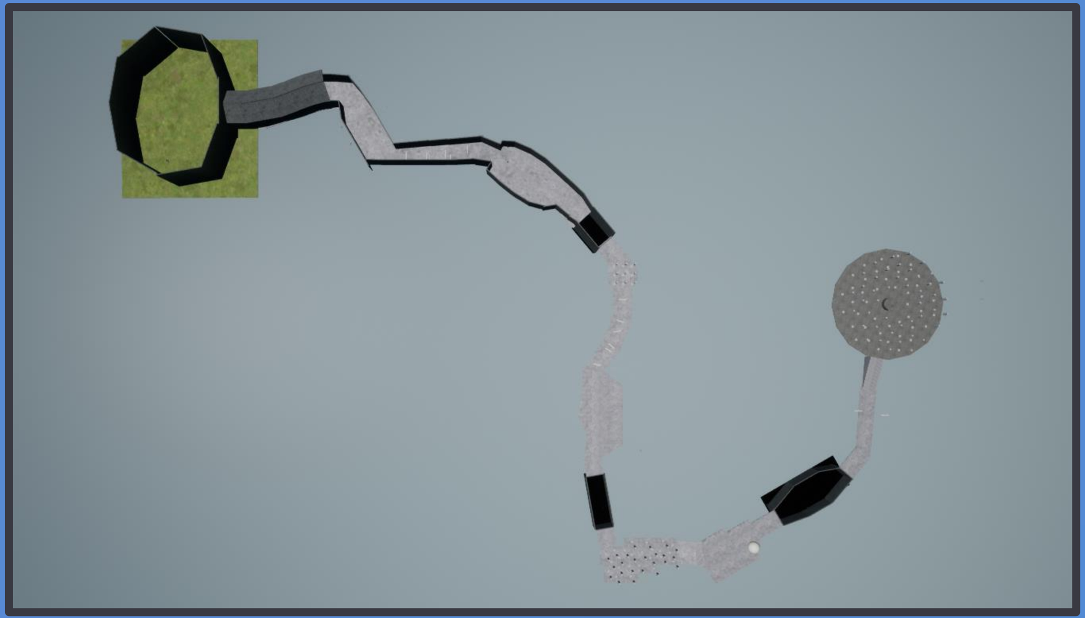




Development | Layout v2.1

After I finished creating the different traps I wanted to use in my level I began reworking the layout of the level in Unreal Engine to better fit the traps I created.



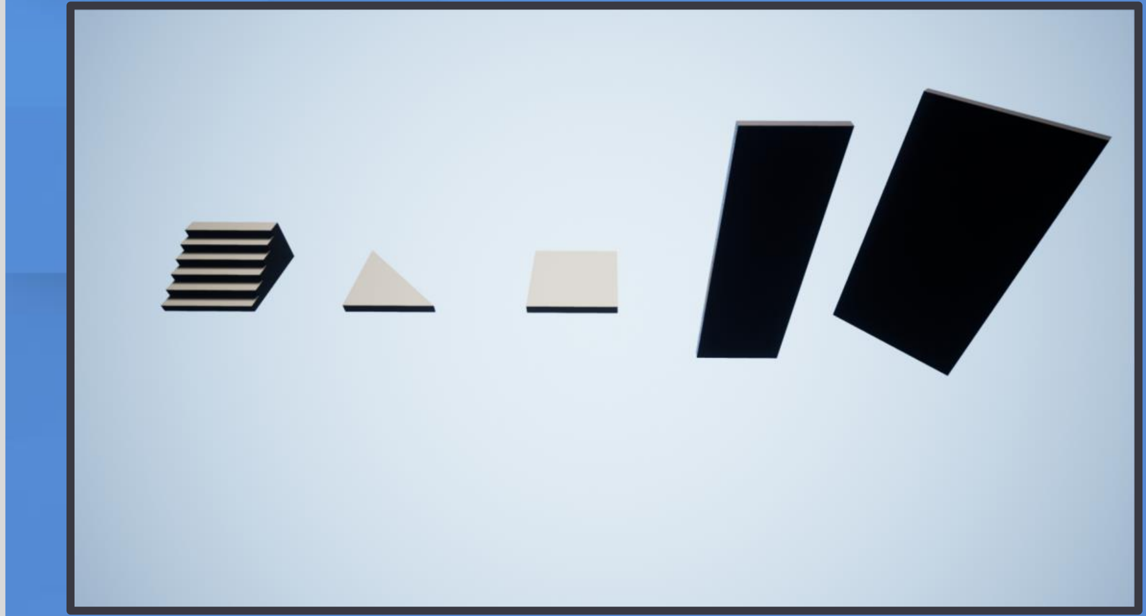


I still felt that there were issues with this layout and I didn't feel that I was blocking out the level as best I could, so I plan to rework the layout using a set of block out meshes.

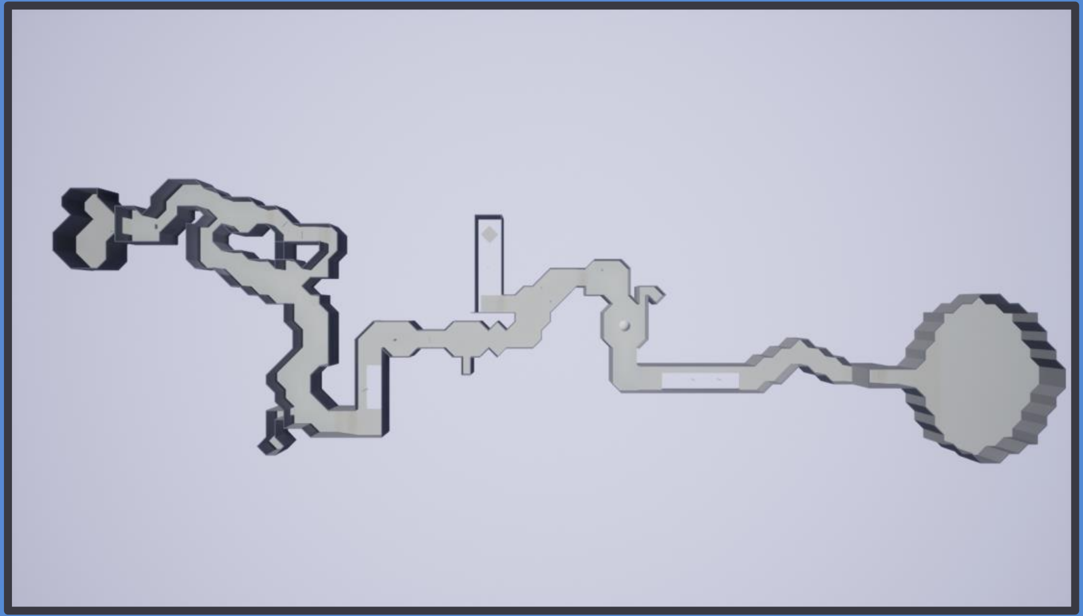
Development | Layout v3.0

For v3.0 of the level layout I decided to use mesh block outs using the geometry tools in Unreal Engine. I created variants of floor and walls as well as stairs to add height and then began blocking out the level. I created multiple routes that the player could at certain sections and hidden areas off to the side of the main path that could be used to hide gold and health packs.

Layout v3.0 (In-Engine: Block out Meshes)



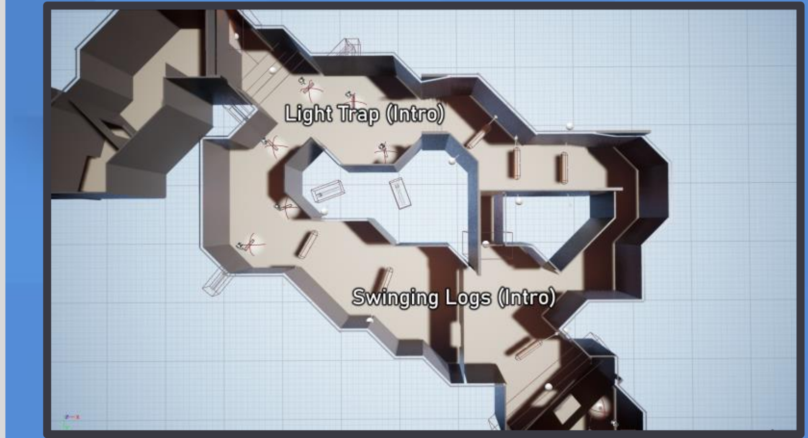
Layout v3.0 (In-Engine: Top Down)



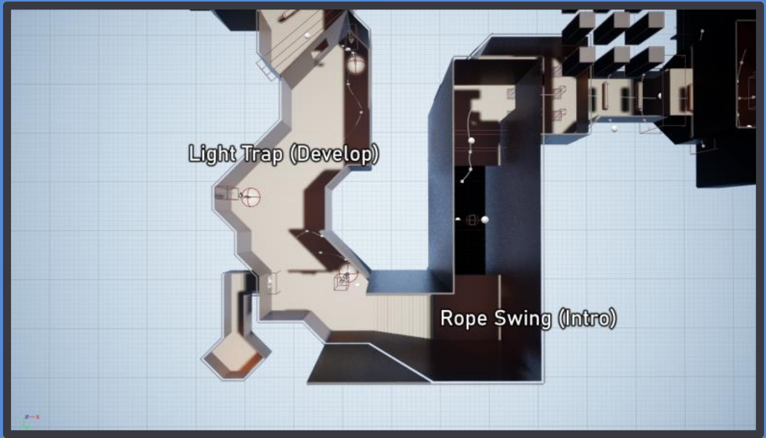
Development | Layout v3.0

Now I had finished a layout for the level, I began adding traps to each section, following the four step level design structure from my research. Annotated screenshots are shown below of where these traps fit in to the layout.

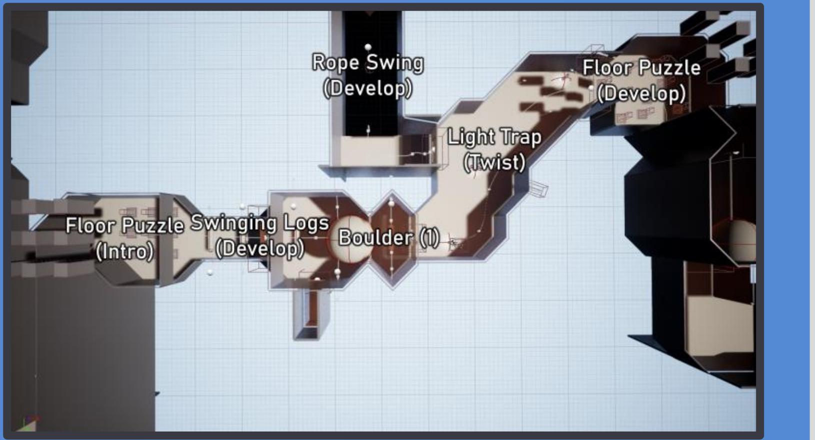
(1)



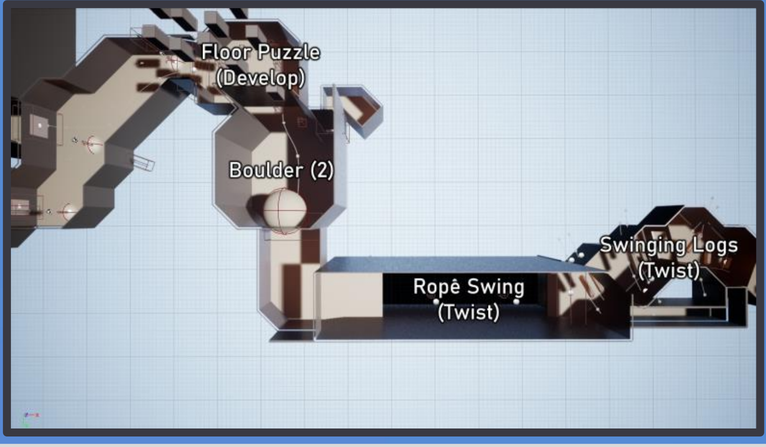
(2)



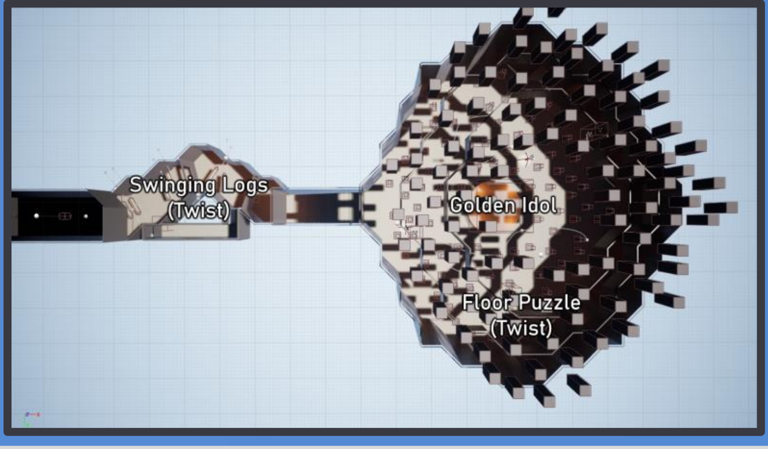
(3)



(4)



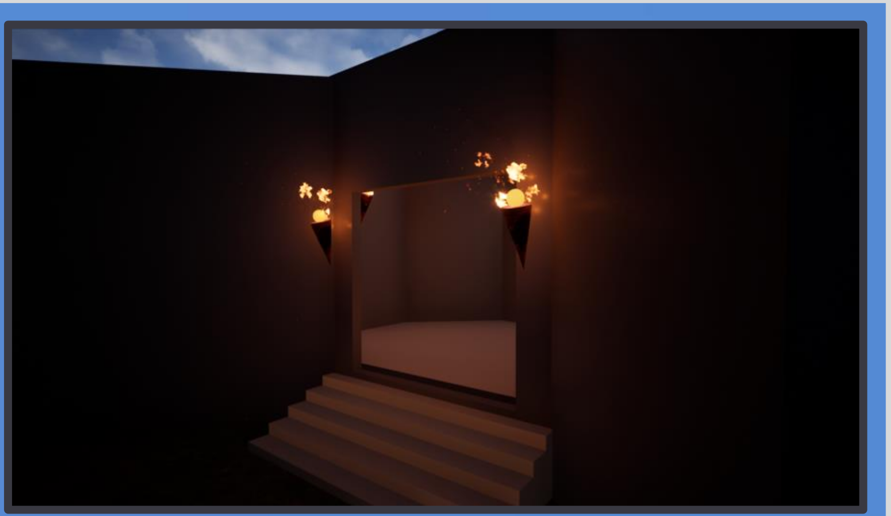
(5)

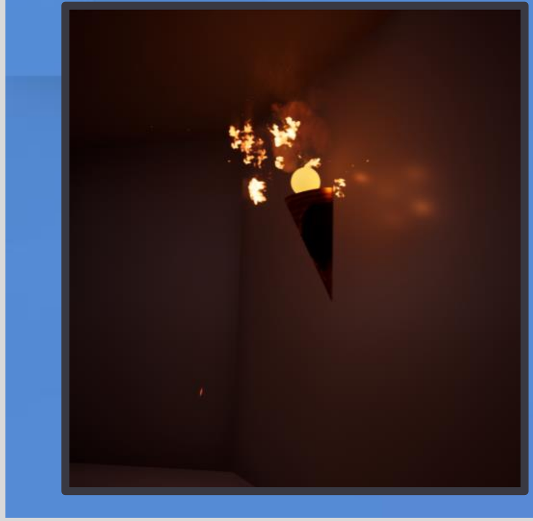


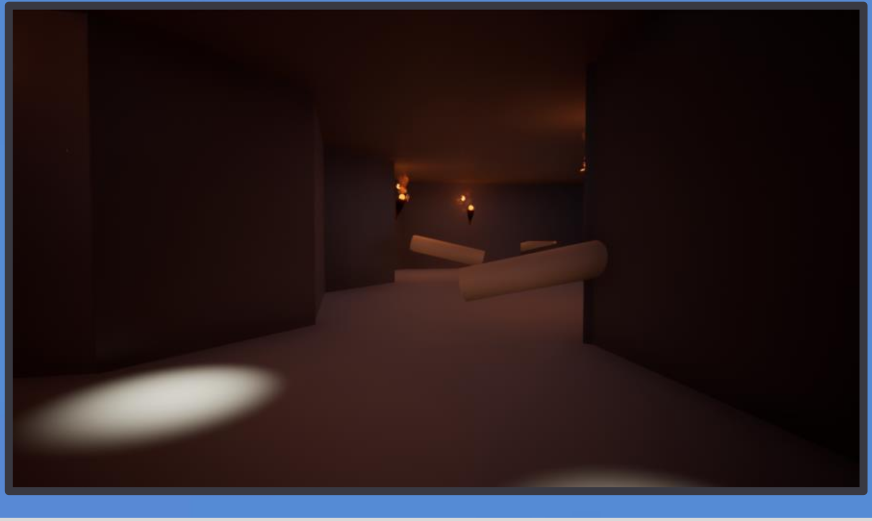
Development | Layout v3.0

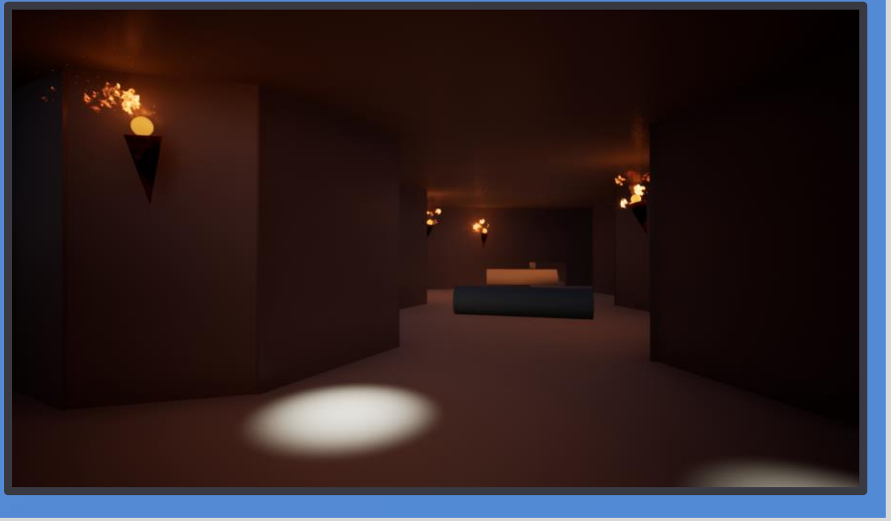
The level was currently quite dark, so I began adding lighting. After creating my torch blueprint, I started scattering these throughout the temple to light up the way for the player.

I wanted to keep the lighting quite moody as the torches were older and fire based so likely wouldn't light up the temple perfectly.



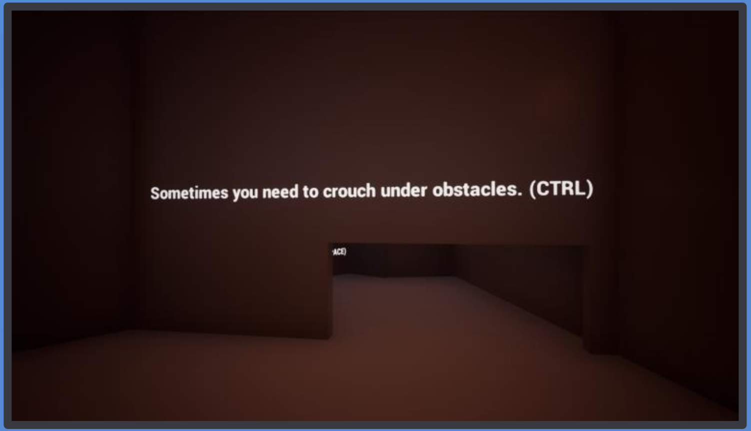




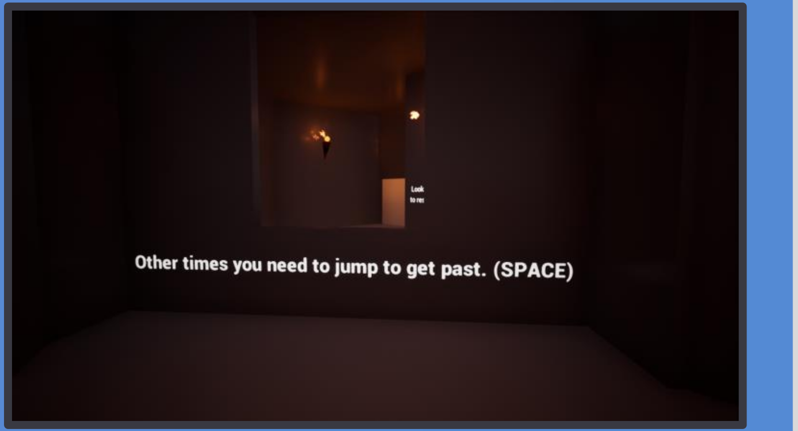


Development | Layout v3.0

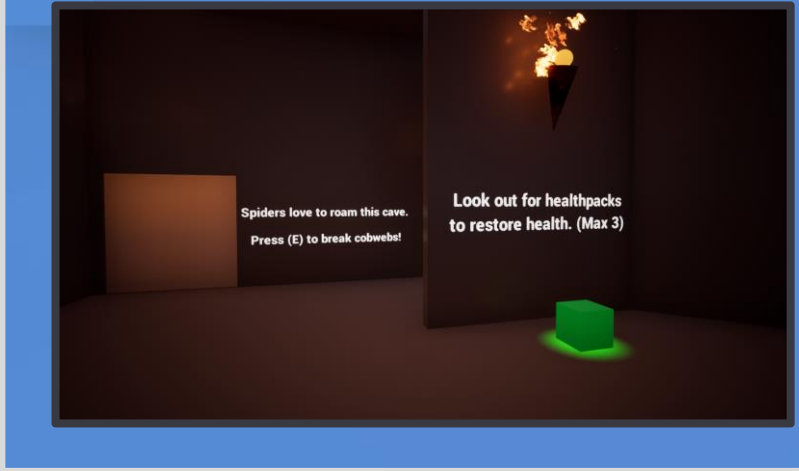
I began working on a tutorial section at the entrance and beginning of the temple. This introduced the player to the key controls needed to play as well as some tips on some game mechanics.



Sometimes you need to crouch under obstacles. (CTRL)



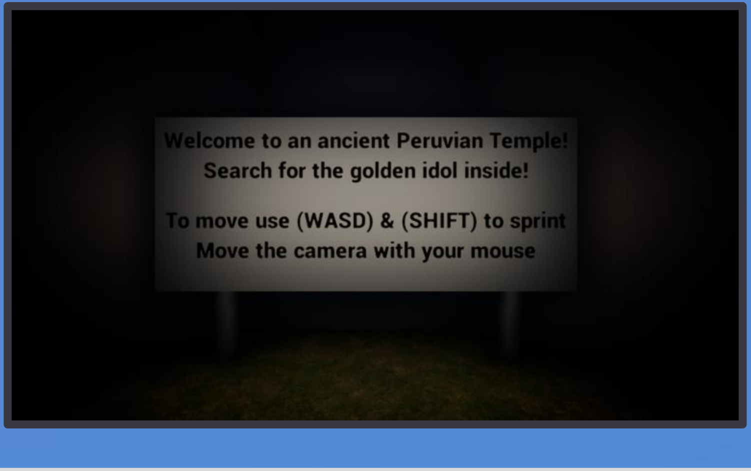
Other times you need to jump to get past. (SPACE)



Spiders love to roam this cave.

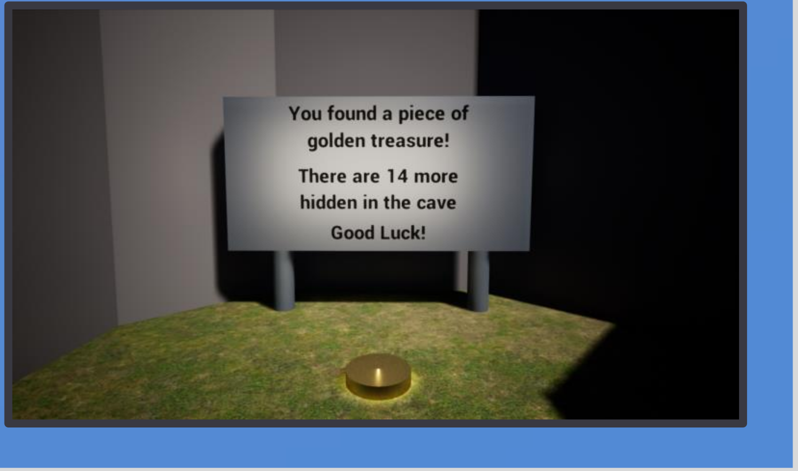
Press (E) to break cobwebs!

Look out for healthpacks to restore health. (Max 3)



Welcome to an ancient Peruvian Temple!
Search for the golden idol inside!

To move use (WASD) & (SHIFT) to sprint
Move the camera with your mouse



You found a piece of golden treasure!

There are 14 more hidden in the cave

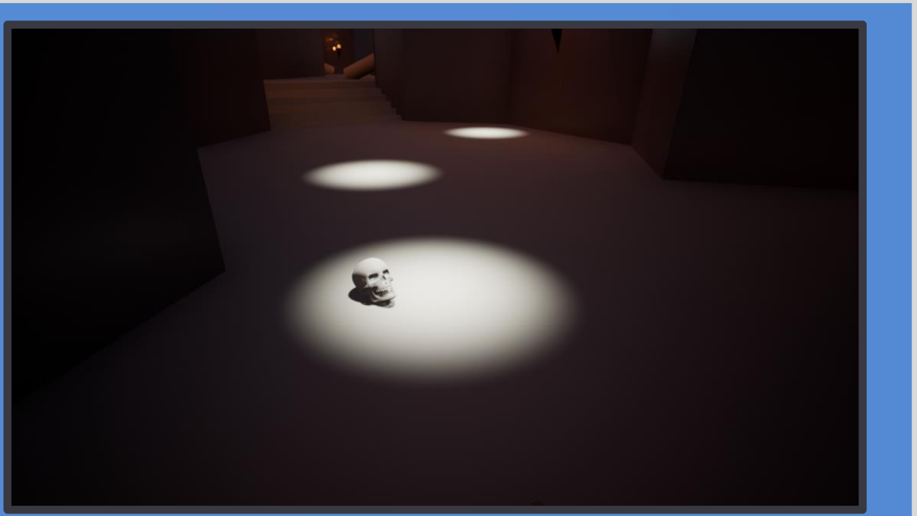
Good Luck!

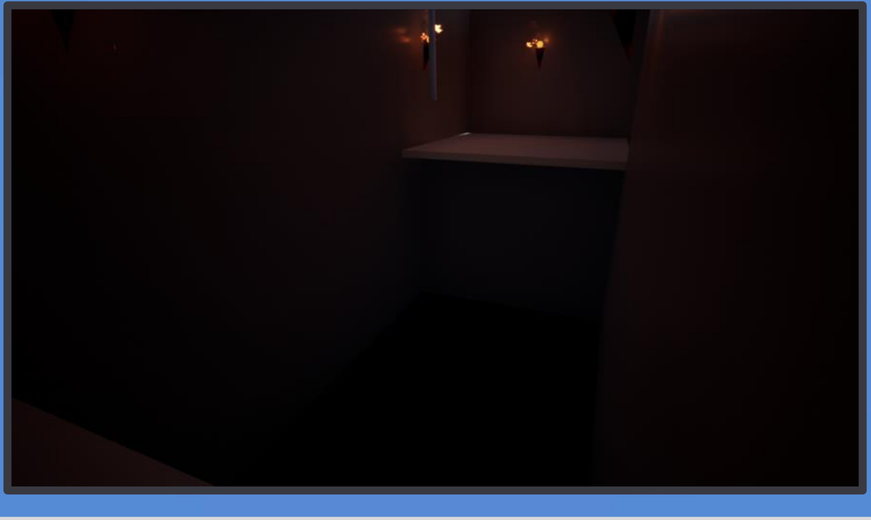
Development | Layout v3.0

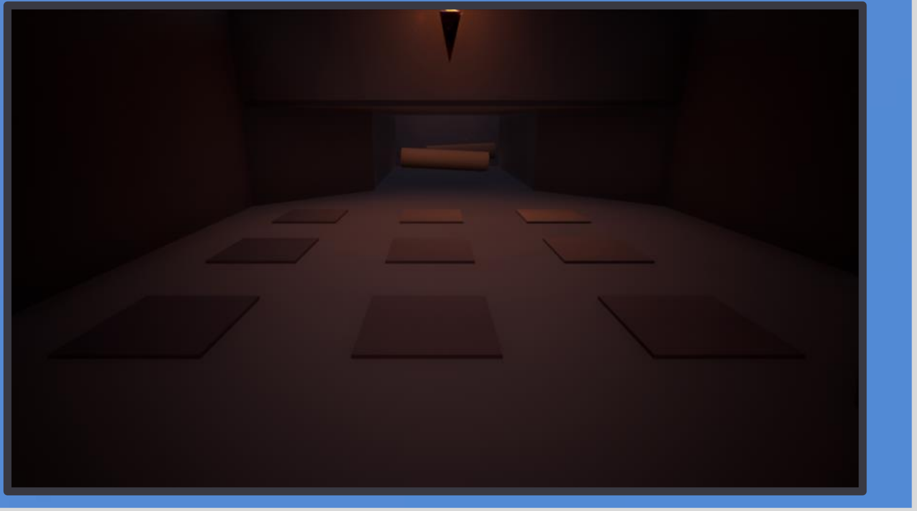
I wanted to keep the traps used fairly simple to understand and that they should be avoided without directly telling the player to avoid them.

For the light trap I did this by placing a skull in the first light as if a previous explorer had died trying to pass through it. The floor puzzle is arranged in a 3x3 grid the first time the player sees it, hinting that it should be avoided.

The rope swing has a large pit that you cannot see the bottom of it, indicating that if you fall you will die.

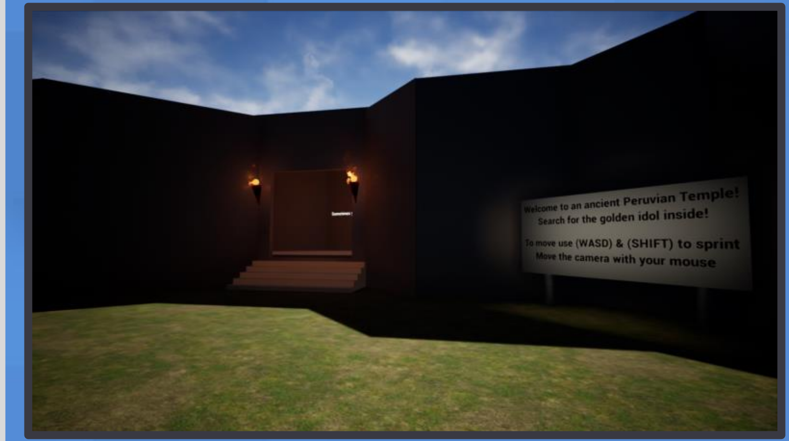

Light Trap (Introduce)


Rope Swing (Introduce)

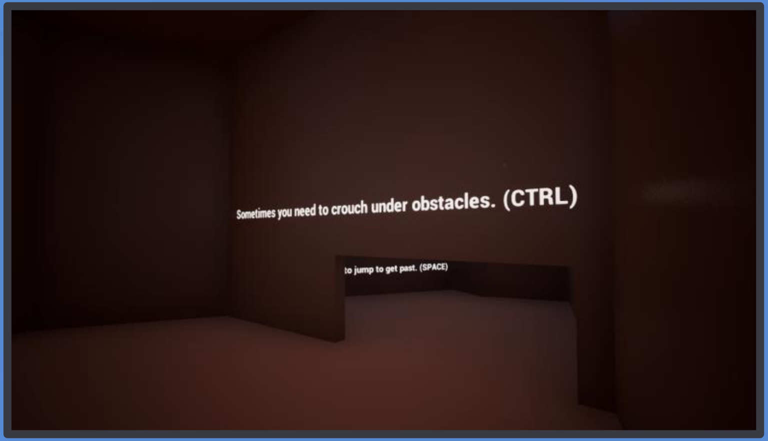

Floor Puzzle (Introduce)

Development Renders (v3.0)

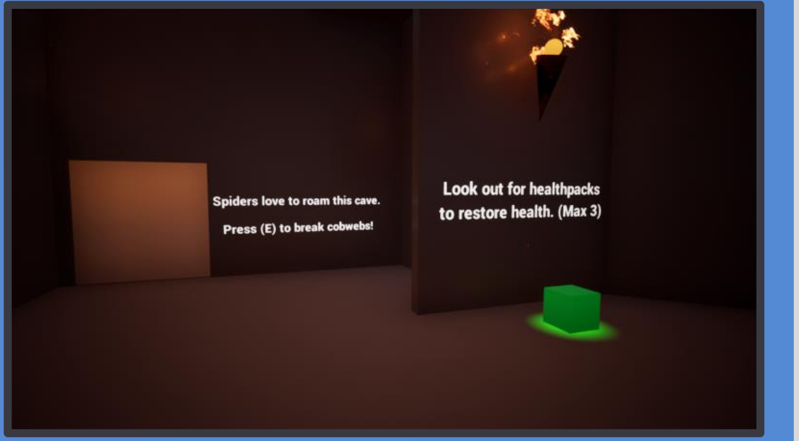
Temple Entrance


Welcome to an ancient Peruvian Temple!
Search for the golden idol inside!
To move use (WASD) & (SHIFT) to sprint
Move the camera with your mouse

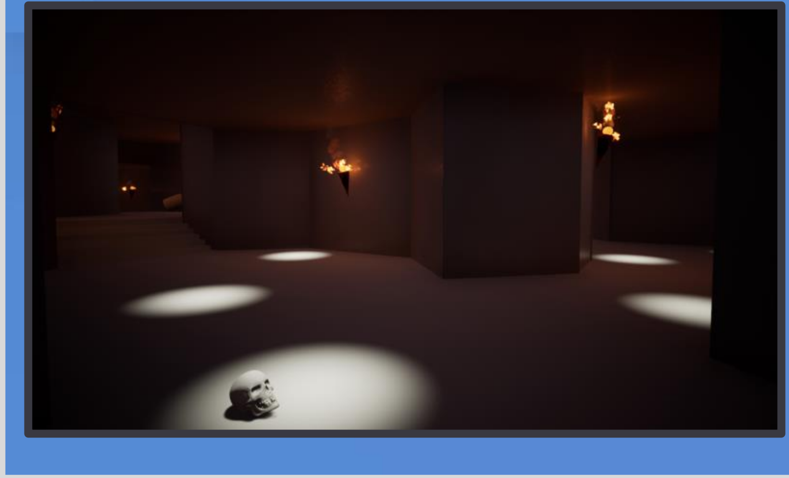
Tutorial Section (Part 1)


Sometimes you need to crouch under obstacles. (CTRL)
to jump to get past. (SPACE)

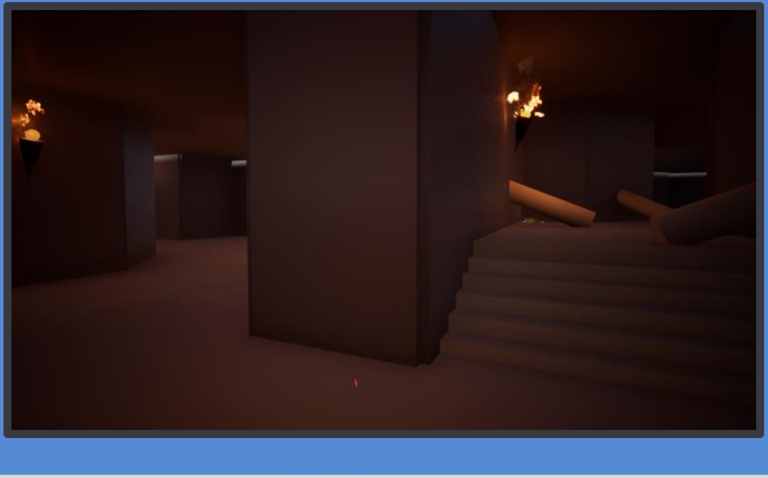
Tutorial Section (Part 2)


Spiders love to roam this cave.
Press (E) to break cobwebs!
Look out for healthpacks
to restore health. (Max 3)

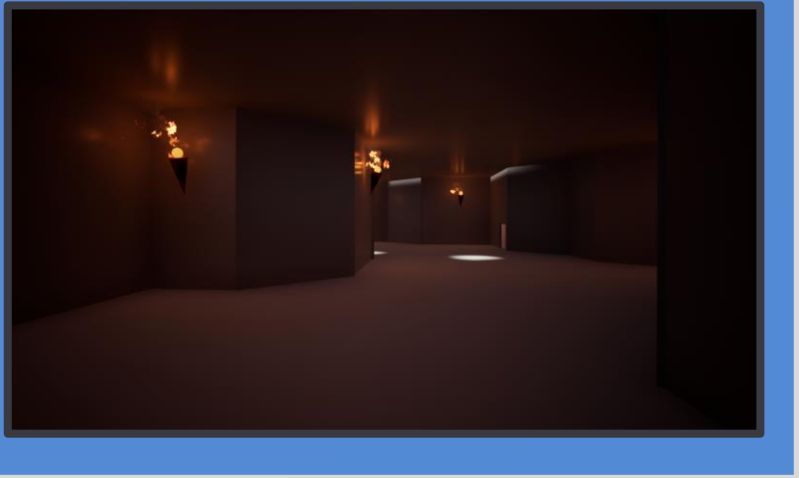
Light Trap (Introduction)



Swinging Logs (Introduction)



Light Trap (Develop)

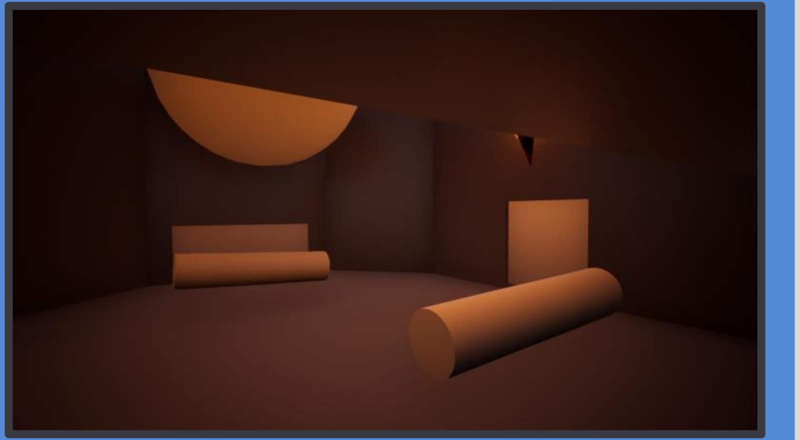


Development Renders (v3.0)

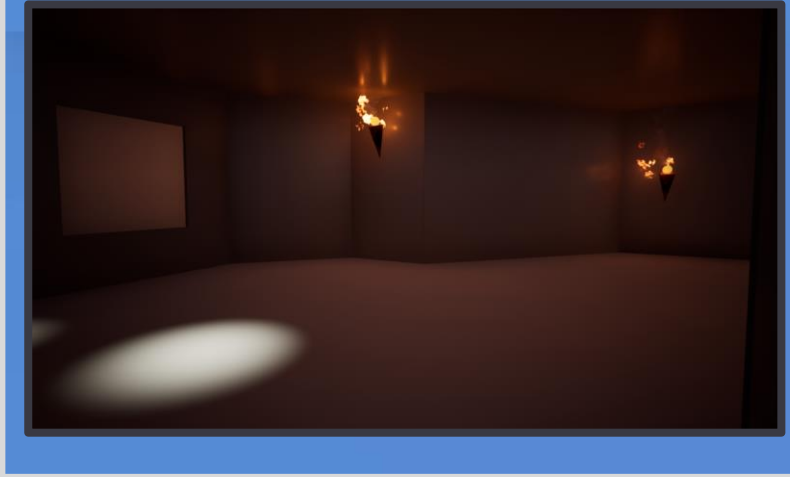
Rope Swing (Introduction)

Floor Puzzle (Introduction)

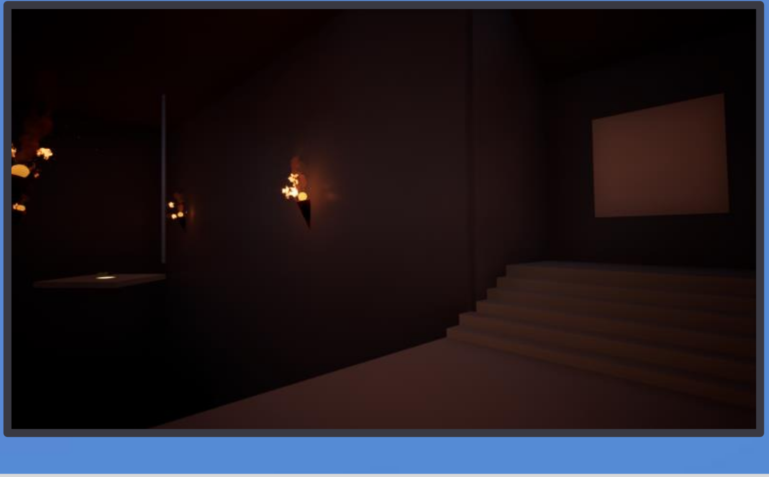
Swinging Logs (Develop)



Light Trap (Twist)



Rope Swing (Develop)



Floor Puzzle (Develop)

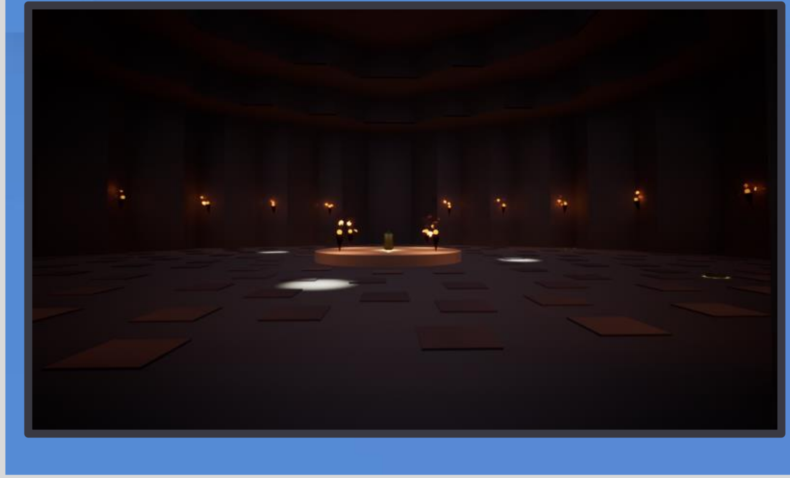
Development Renders (v3.0)

Rope Swing (Twist)

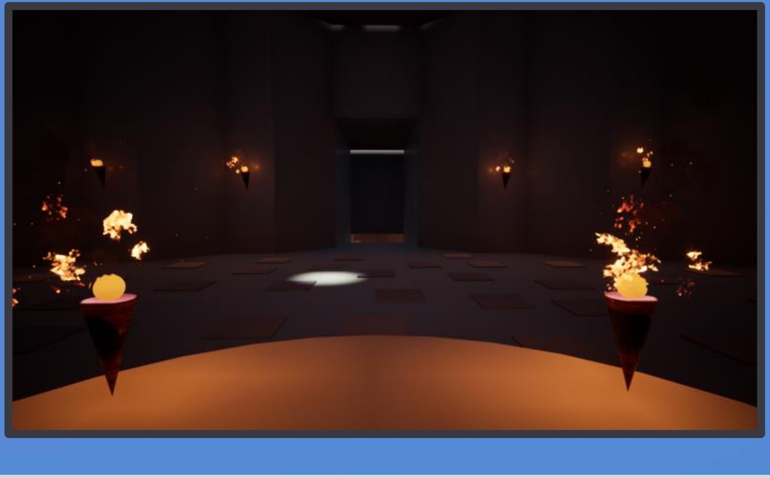
Swinging Logs (Twist)

Entrance to Idol Room

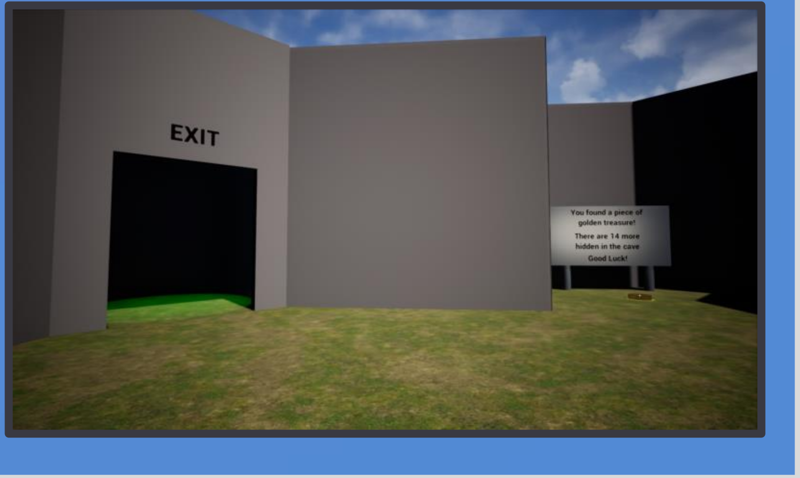
Idol Room / Floor Puzzle (Twist)



Idol Room



Temple Exit


EXIT
You found a piece of golden treasure!
There are 14 more hidden in the cave
Good Luck!

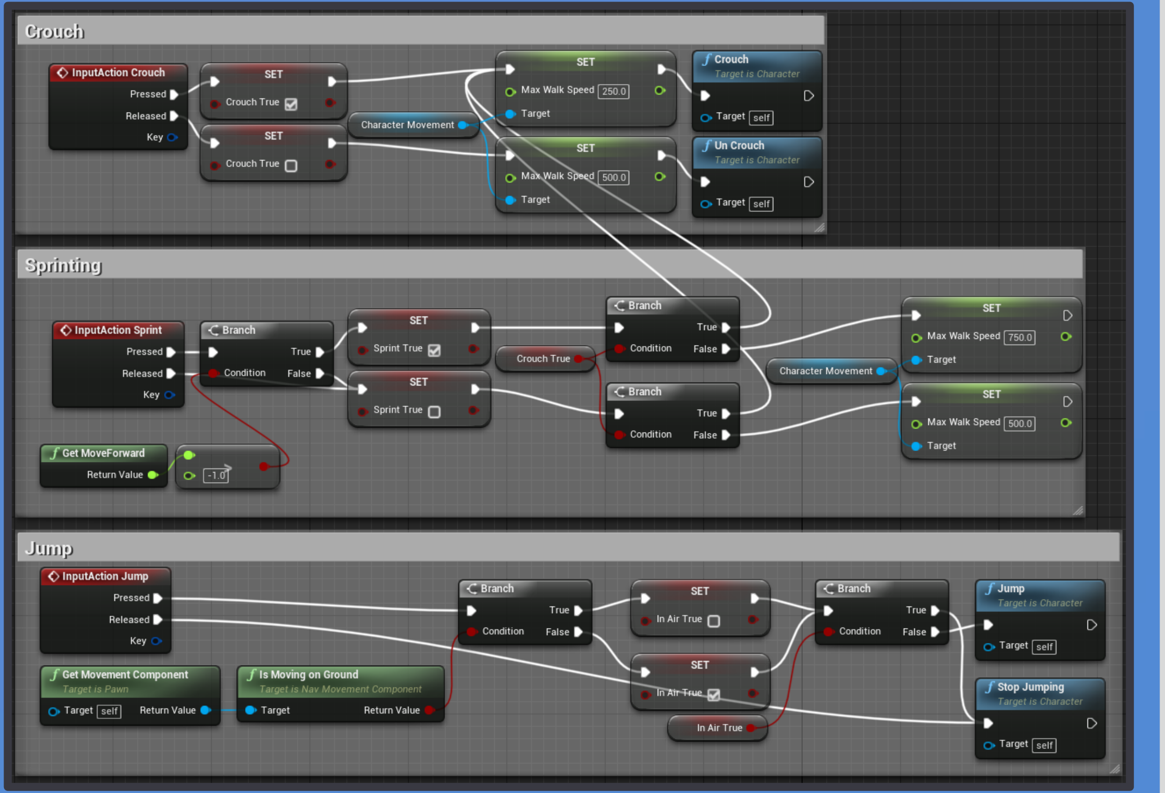
Blueprints | Player Movement

My player character started out using Unreal Engines third person character template. I converted this to first person so I could still have a player body by moving the camera and changing some camera control settings.

First Person Character

I began by creating some basic additional movement for the player. I added crouch which lowered the player speed, camera and hitbox. Sprint allowed the player to move faster than walking and finally jumping to let the player jump over obstacles. Jumping had some limiters to ensure the player could not jump whilst already in the air.

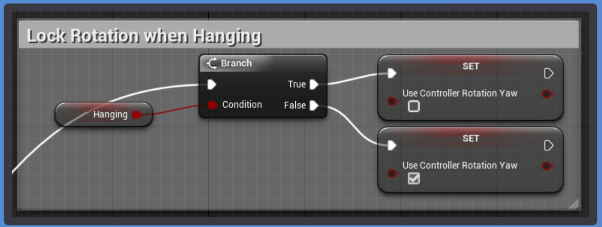
Basic movement



Blueprints | Hanging Movement

When the player jumps on the rope swing, certain movement needed to change and also be stopped from affecting the player as this was causing a range of issues when hanging off the rope.

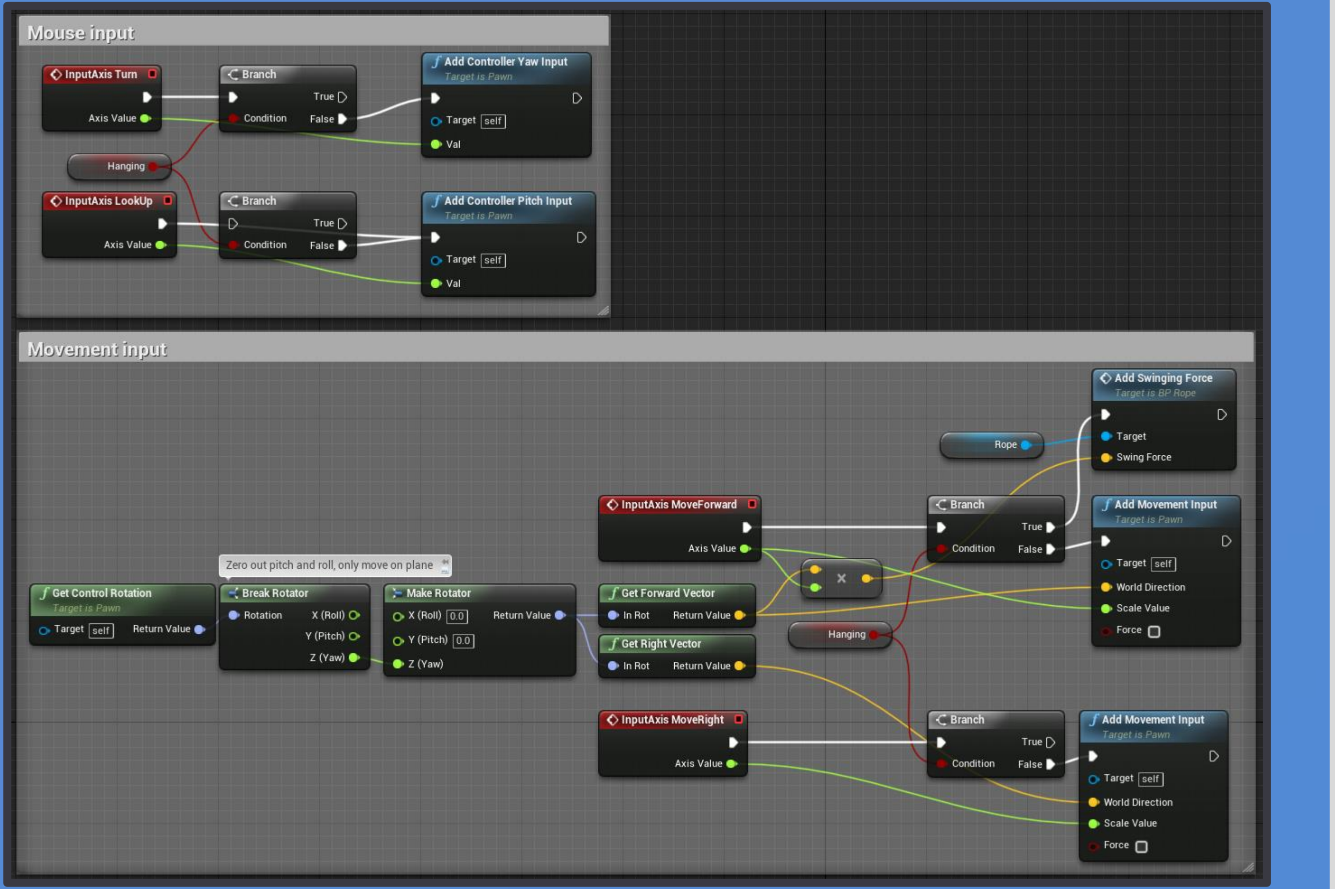
I altered the mouse and movement input to work with the rope swing. To do this I edited the default movement blueprints adding toggles for when hanging on the rope and when on ground. When hanging the players left and right camera movements are blocked to stop any odd rotation. The left and right character movements are stopped too. The front and back movement will swap between adding swinging force to the rope and standard movement. I also added a rotation lock when the player is hanging to help stop rotation issues.



Lock Rotation when Hanging
Branch
Hanging
Condition
True
False
SET
Use Controller Rotation Yaw
SET
Use Controller Rotation Yaw

Rotation lock is checked every tick using the event tick node.

Hanging Movement

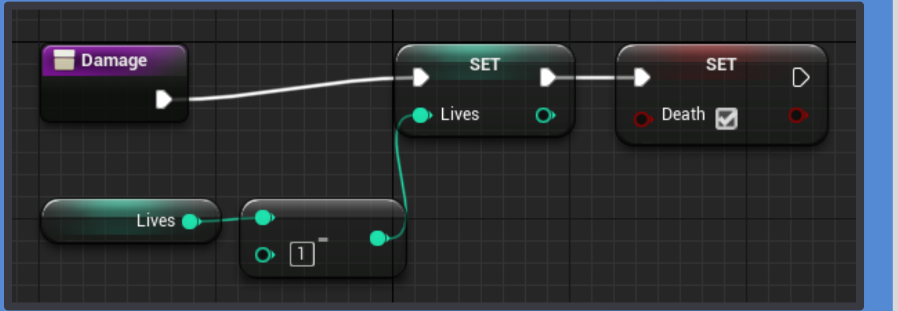


Blueprints | Damage & Respawn

When the player needs to take damage the damage script is run which takes 1 life away from the player. This can be accessed by any trap when needed.

To check if the player has taken damage a boolean is used to indicate that the player has died. A check is run every tick and restarts the game or respawns the character depending on how many lives are left.

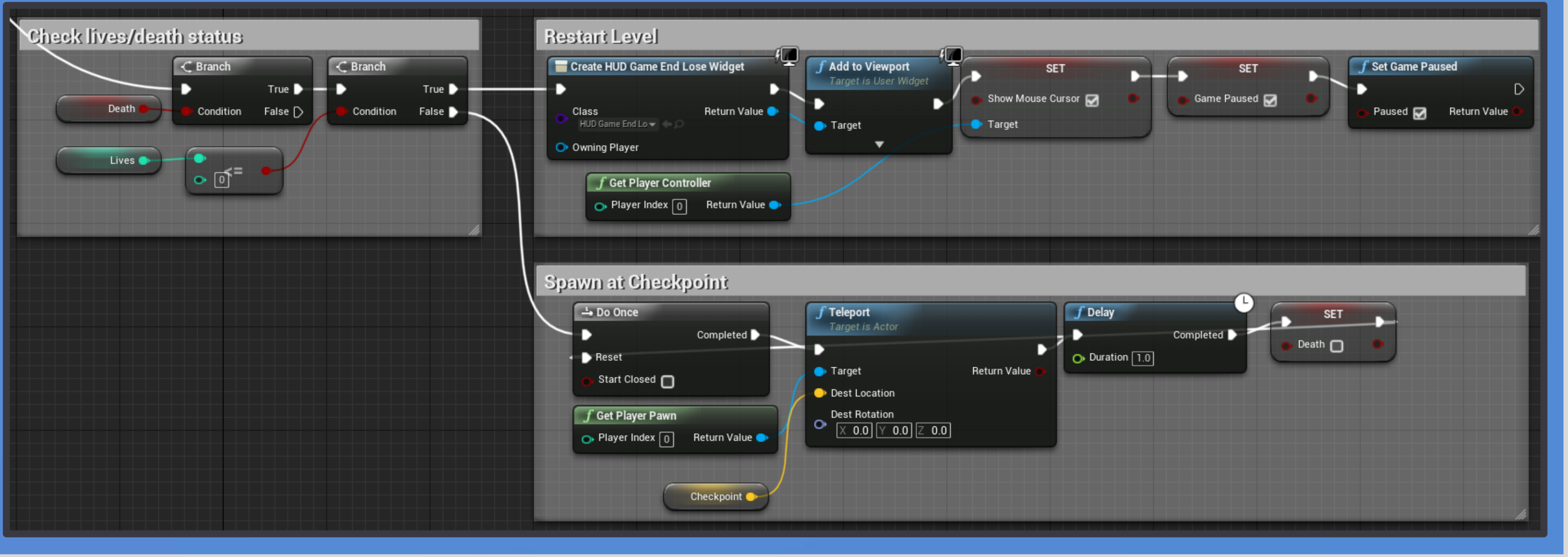
Damage Script



If the player has run out of lives the game restarts by creating the end lose HUD and pausing the game and character.

If the player still has lives remaining they are instead teleported to their latest checkpoint to try again.

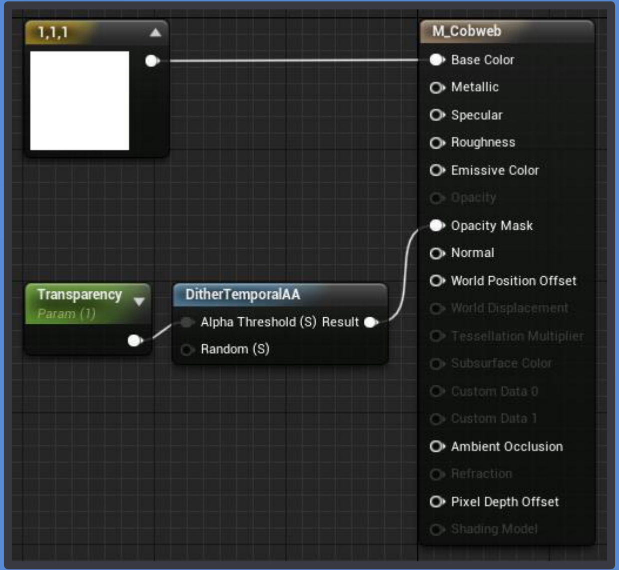
Check lives and respawn character



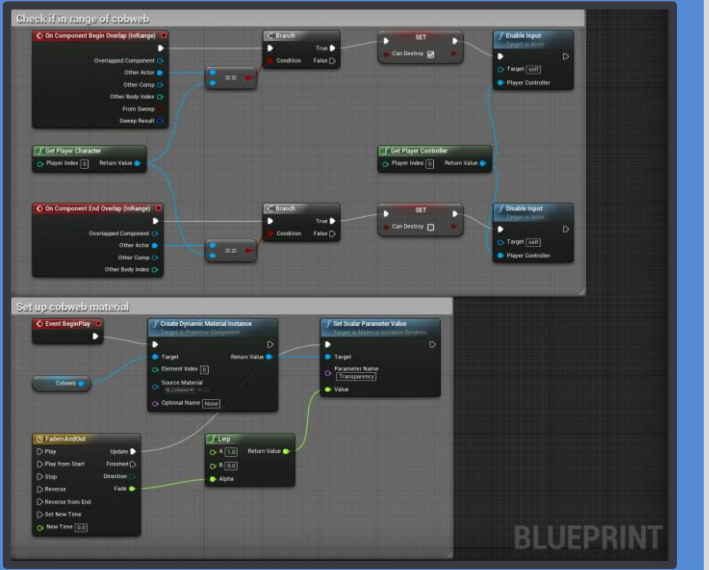
Blueprints | Cobwebs

The cobweb uses a cube and a box collision. The box collision is used to check if the player is in range to destroy it. The cube acts as the cobweb and has a dynamic material instance on it that fades in and out when the player interacts with the cobweb. A timeline was used to control the speed the cobweb fades in and out at.

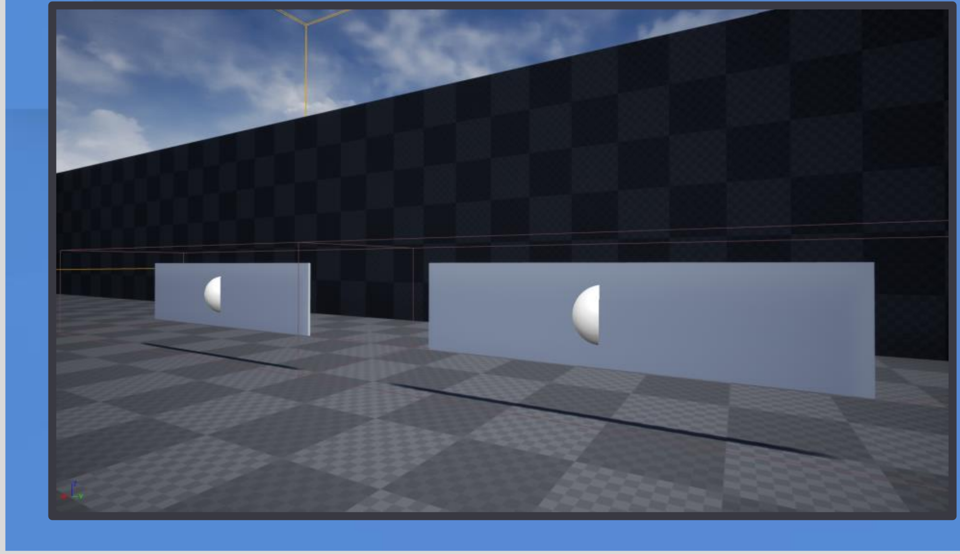
Cobweb Material



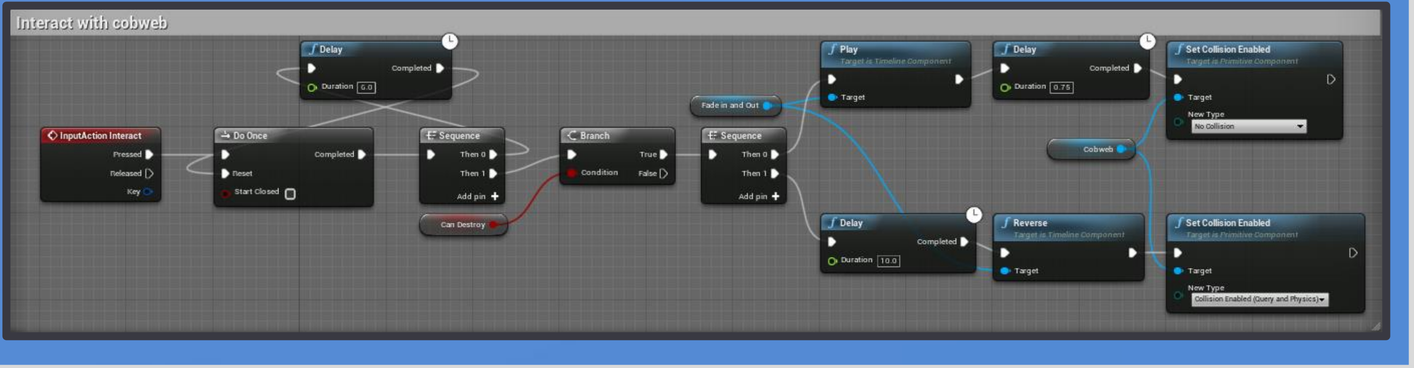
Check range / Make Material



In-Engine Screenshot



Interact with the cobweb when player is in range

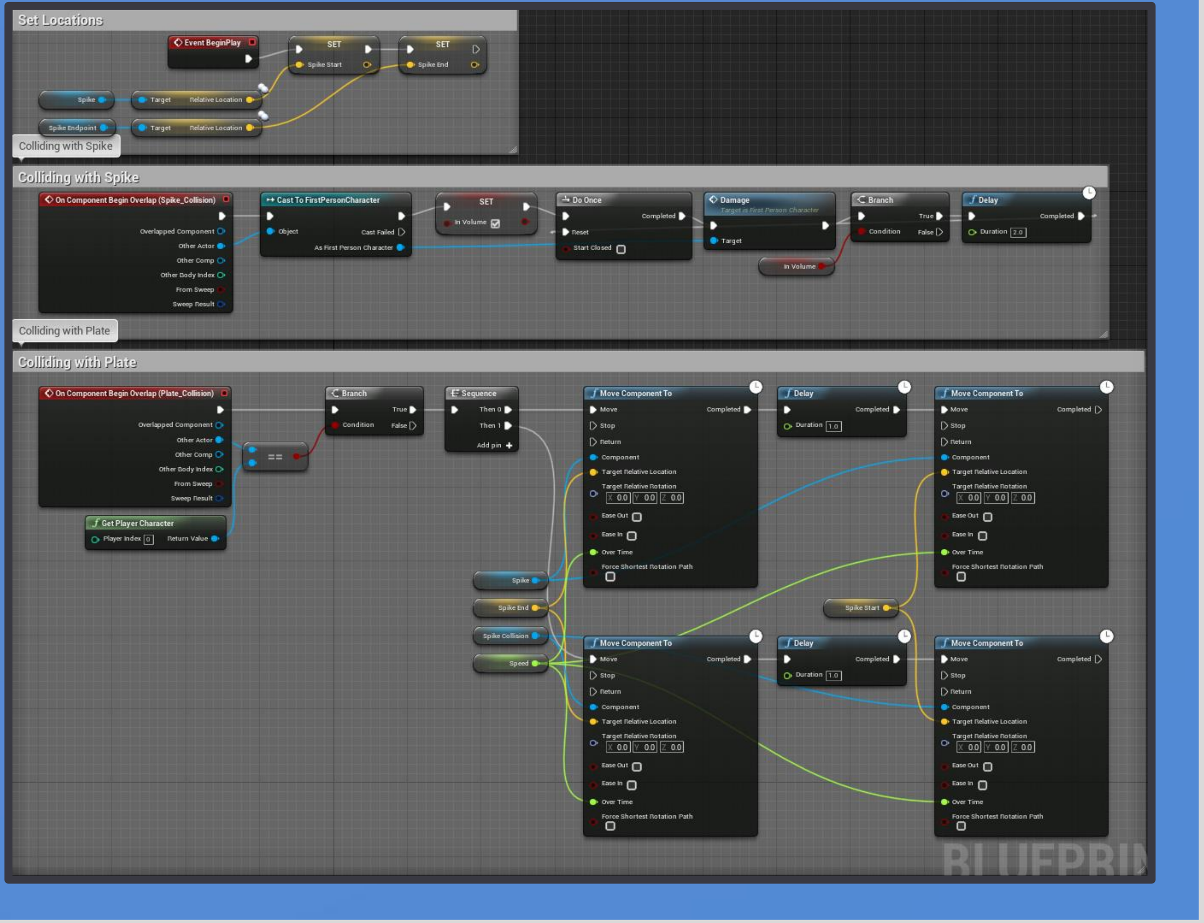


Material Tutorial: https://www.youtube.com/watch?v=aEZeGknl-HU

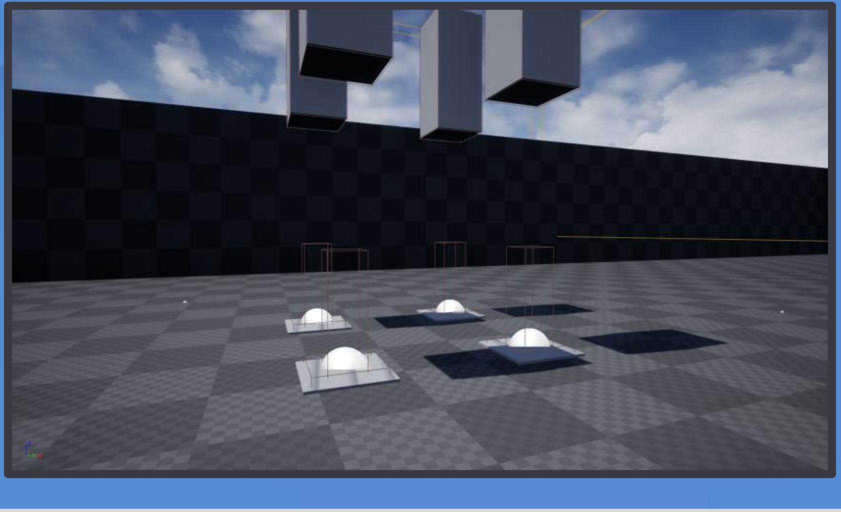
Blueprints | Floor Puzzle

The floor puzzle uses a pressure plate and a crushing spike that comes in from the ceiling. When the player collides with the plate, the spike falls down crushing the player. The blueprint primarily uses move component to nodes to move the spike and collision boxes to appropriate locations. I used a box collision as an end point as this allowed me to move the spike individually if needed without causing issues with movement.

Blueprint Screenshot



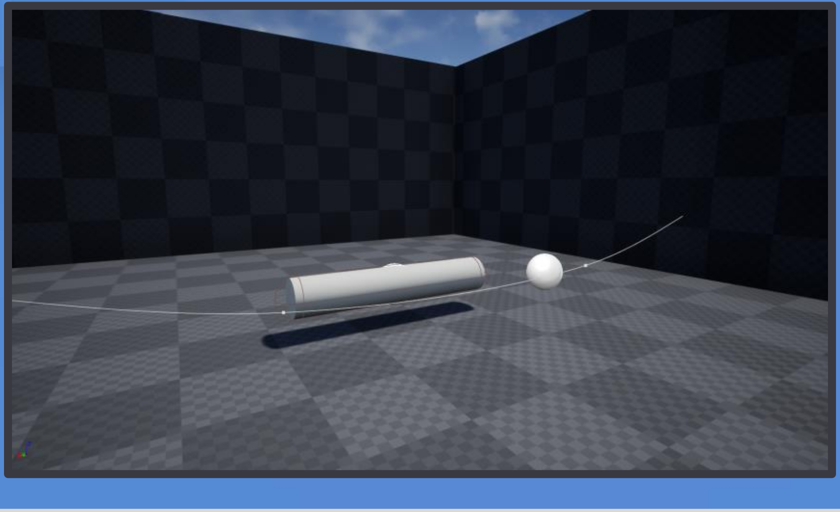
In-Engine Screenshot



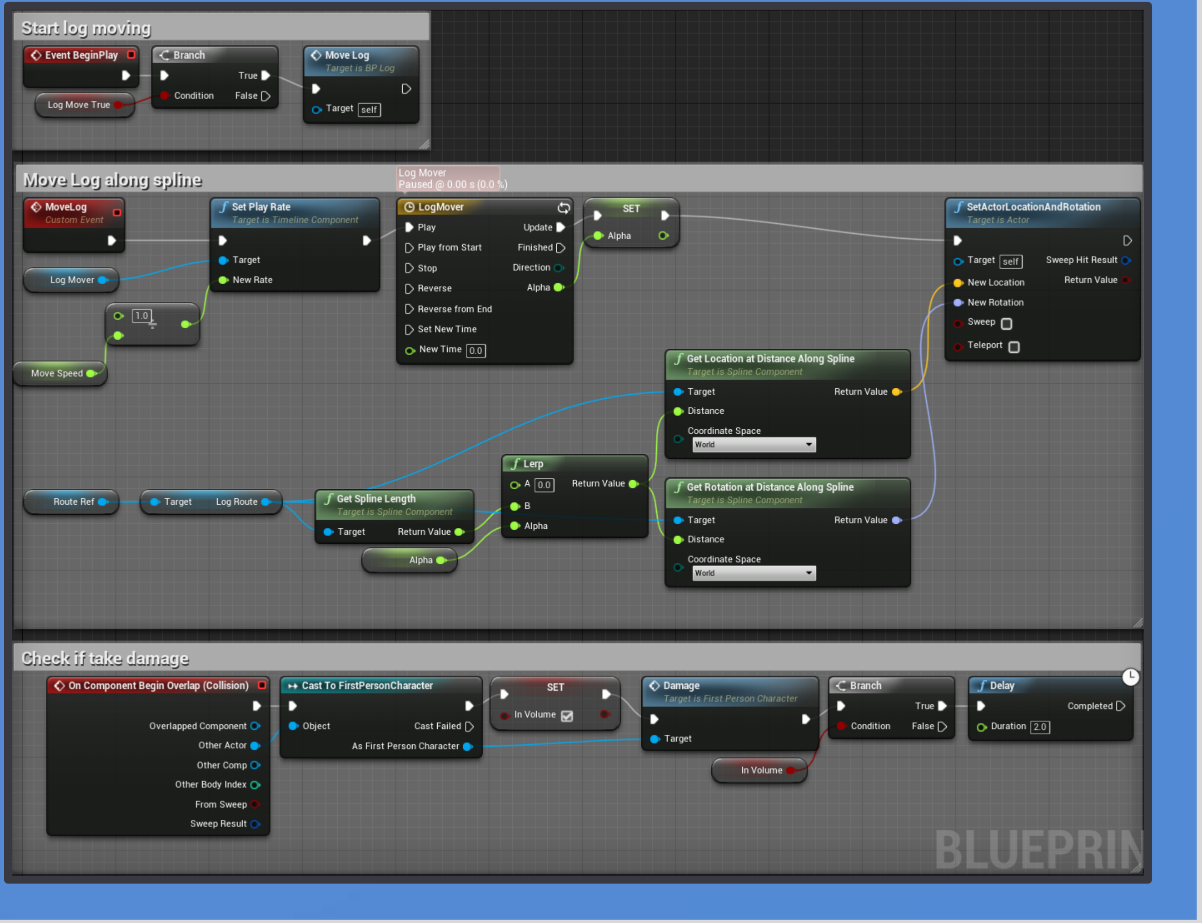
Blueprints | Swinging Logs

The swinging log swings from side to side. If the player collides with the log they take damage. Two blueprints were made, one for the log itself which included the cylinder and a capsule collision and another for the route the log would swing on which included a spline component. The route was created using the spline blueprint and then in the log blueprint the spline was chosen and a timeline was used to control the speed and position of the log along the spline.

In-Engine Screenshot



Blueprint Screenshot

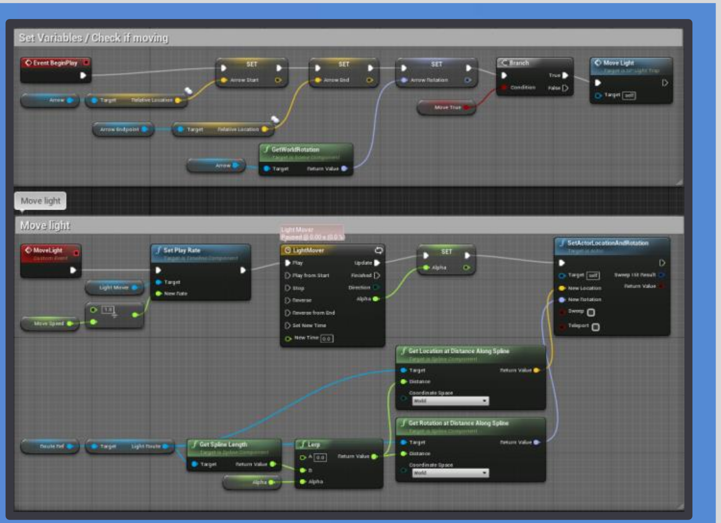


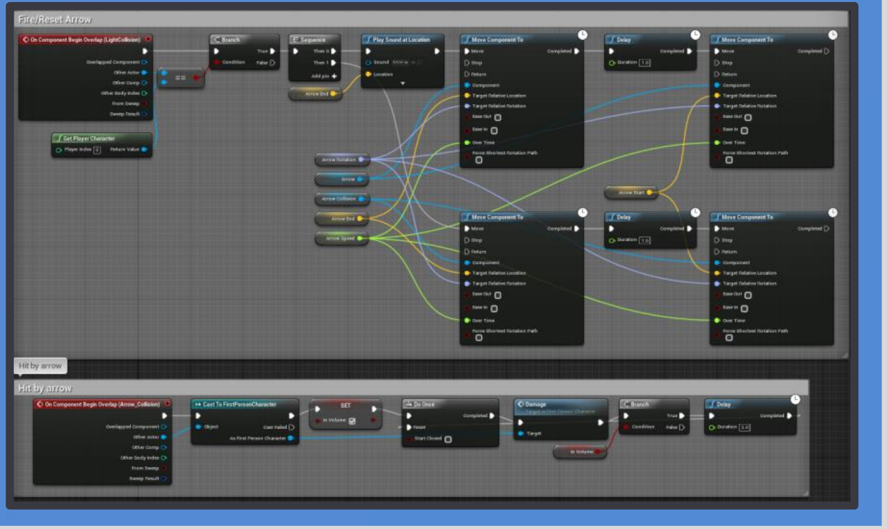
Move objects along a spline: https://www.youtube.com/watch?v=bWXI91FdMtk

Blueprints | Light Trap

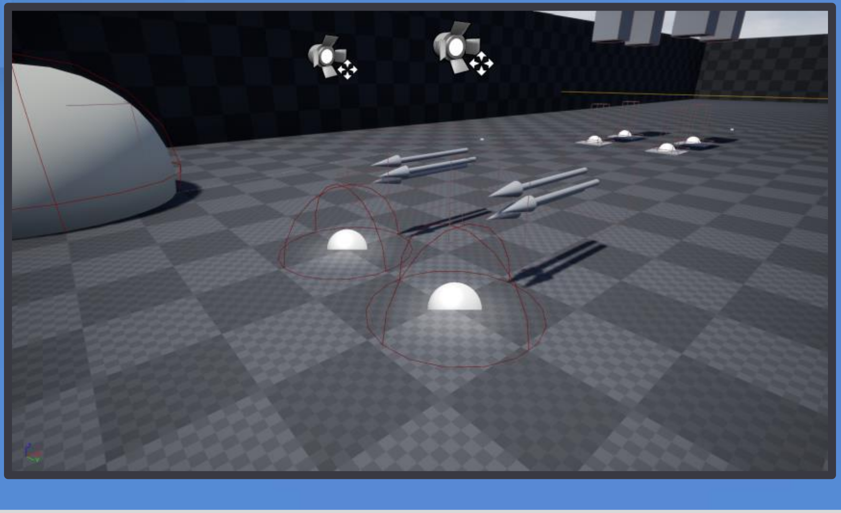
The light trap fires an arrow at the player if the player enters the light spot on the floor. This uses a move component to note similar to the floor puzzle. As part of the develop step of this trap, I coded it to follow a spline if needed. Using a similar mechanic to the swinging logs and adding a toggle variable I could now turn each one on or off individually when adding them to the scene. The moving light traps had the arrows rotated vertically to fire from the ceiling as if they came in from the side they would potentially be seen by the player when they hadn't been fired.

Blueprint Screenshots


Set Variables / Check if moving
Event BeginPlay
SET
Branch
Move Light
Move light
MoveLight
Set Play Rate
LightMover
SetActorLocationAndRotation
Get Location at Distance Along Spline
Get Rotation at Distance Along Spline
Get Spline Length
Lerp



In-Engine Screenshot



Two blueprints were made for this trap, one for the light trap itself and another for the spline that would be used if the trap was set to moveable.

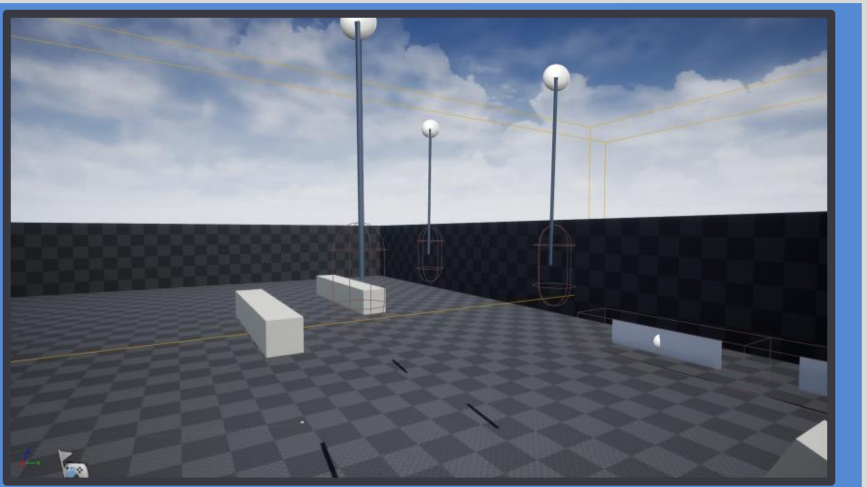
When the arrow is fired, a sound effect is played at the location of the arrow.

Move objects along a spline: https://www.youtube.com/watch?v=bWXI91FdMtk

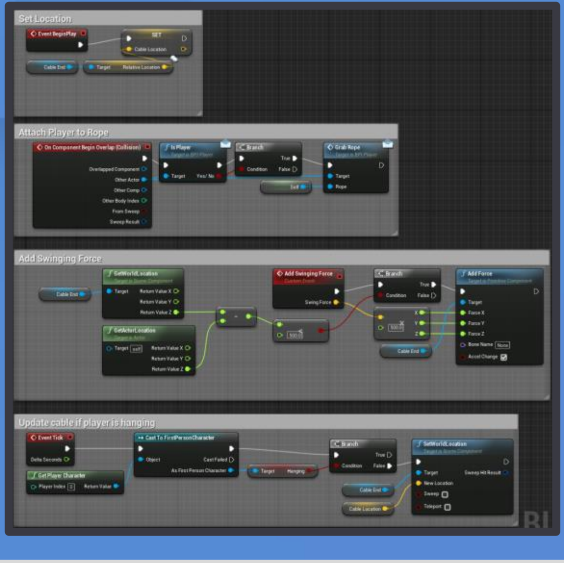
Blueprints | Rope Swing

The rope swing was the hardest trap to create. I followed a tutorial for the most part but had to adapt it to fit my first person character. The rope is made up of a cable component with a sphere attached to the end acting as the players grab point. If the player gets in range of the sphere they are attached to the rope and move with the rope. The player can move the rope using W and S and detach from the rope using SPACE. The rope movement was programmed in the rope blueprint and the player was attached to the rope in the first person character blueprint.

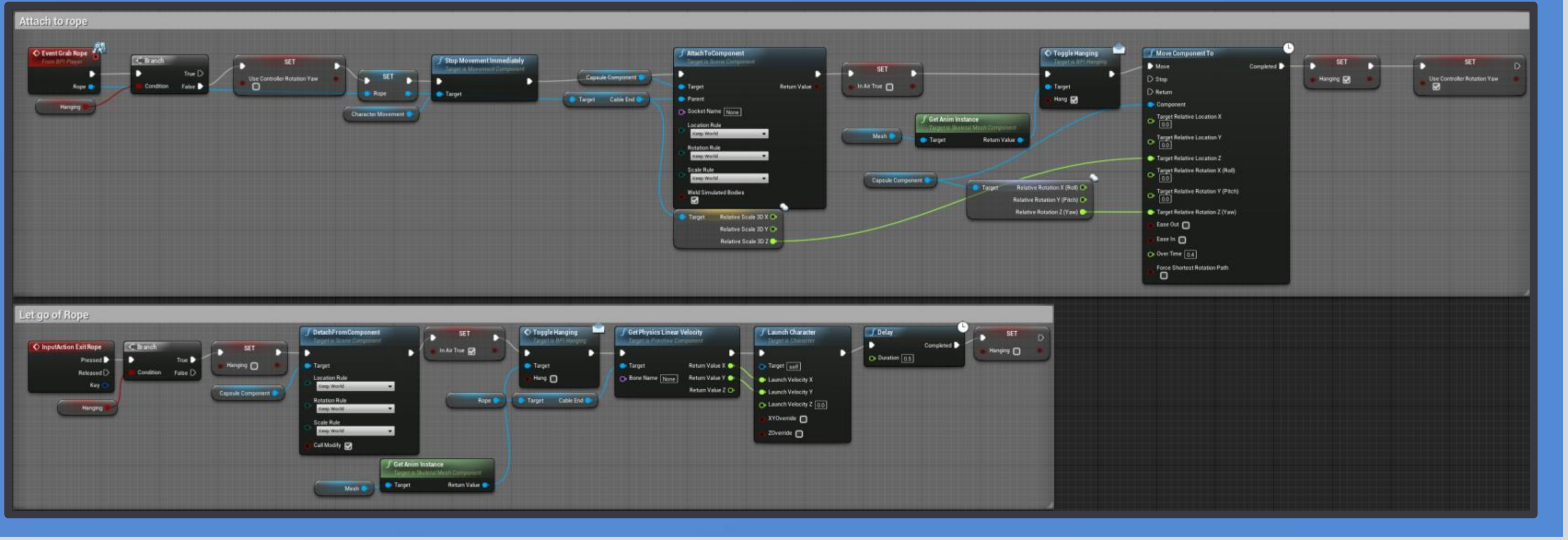
In-Engine Screenshot



Rope Swing Blueprint



Character Blueprint

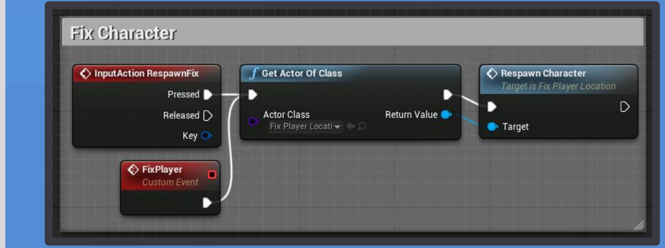


Rope swing tutorial: https://www.youtube.com/watch?v=nmnhIqRXwxA

Blueprints | Rope Swing

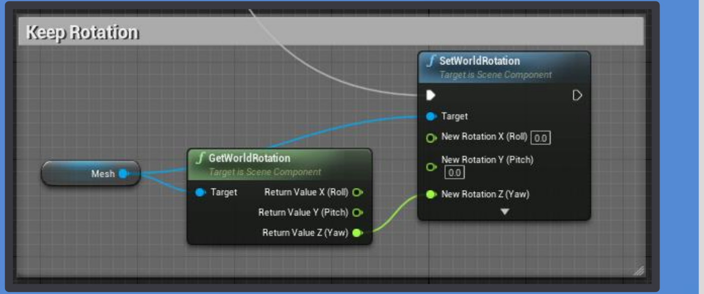
The rope swing caused a lot of issues and many bugs arose in its creation. The biggest was the character being slightly offset after jumping off the rope. I believe this was due to the attach to component node. I tried many workarounds to try and fix this issue however I was unable to stop the offset occurring. To provide a fix of this I created a Reset Player blueprint that would destroy and respawn the player component where the player was standing to reset the offset. This is an awkward workaround but due to time constraints I was unable to continue debugging the rope or recreate it entirely. I had to temporarily store the necessary character variables as these would be lost when respawning the character.

Start Reset

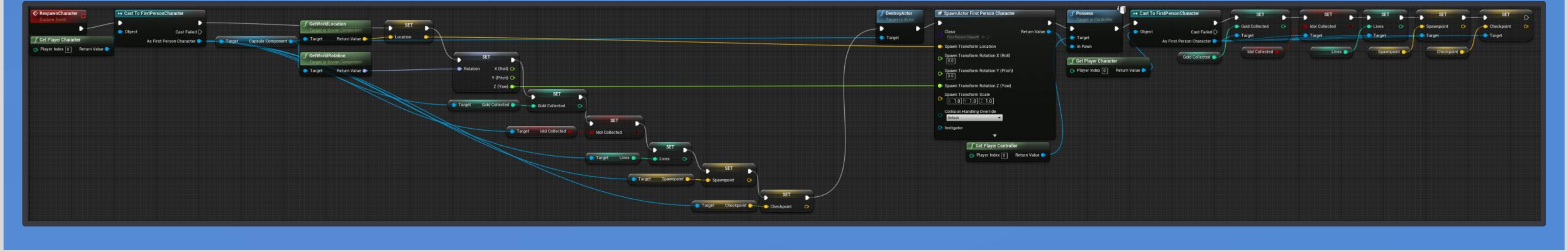


I also had to limit the character swinging side to side as this would often cause the player character to rotate awkwardly, sometimes spinning the player model round without the camera. I had to lock character rotation when hanging as when the player jumped off the rope they could land at an unintended angle.

Lock Rotation



Reset Character

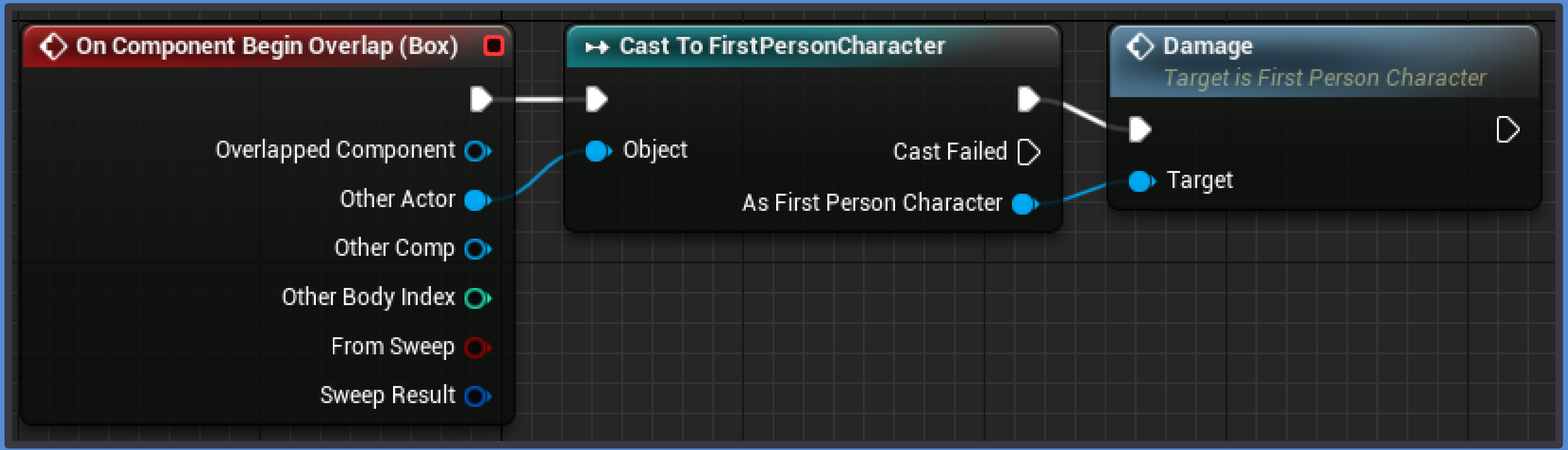


Rope swing tutorial: https://www.youtube.com/watch?v=nmnhIqRXwxA

Blueprints | Damage box

The damage box is used to check if the player has fallen into a pit below a rope swing. It uses a box collision to check when the player overlaps and then casts to the first person character blueprint to run the damage script. It could also be used in other areas if needed to damage the player.

In-Engine Screenshot

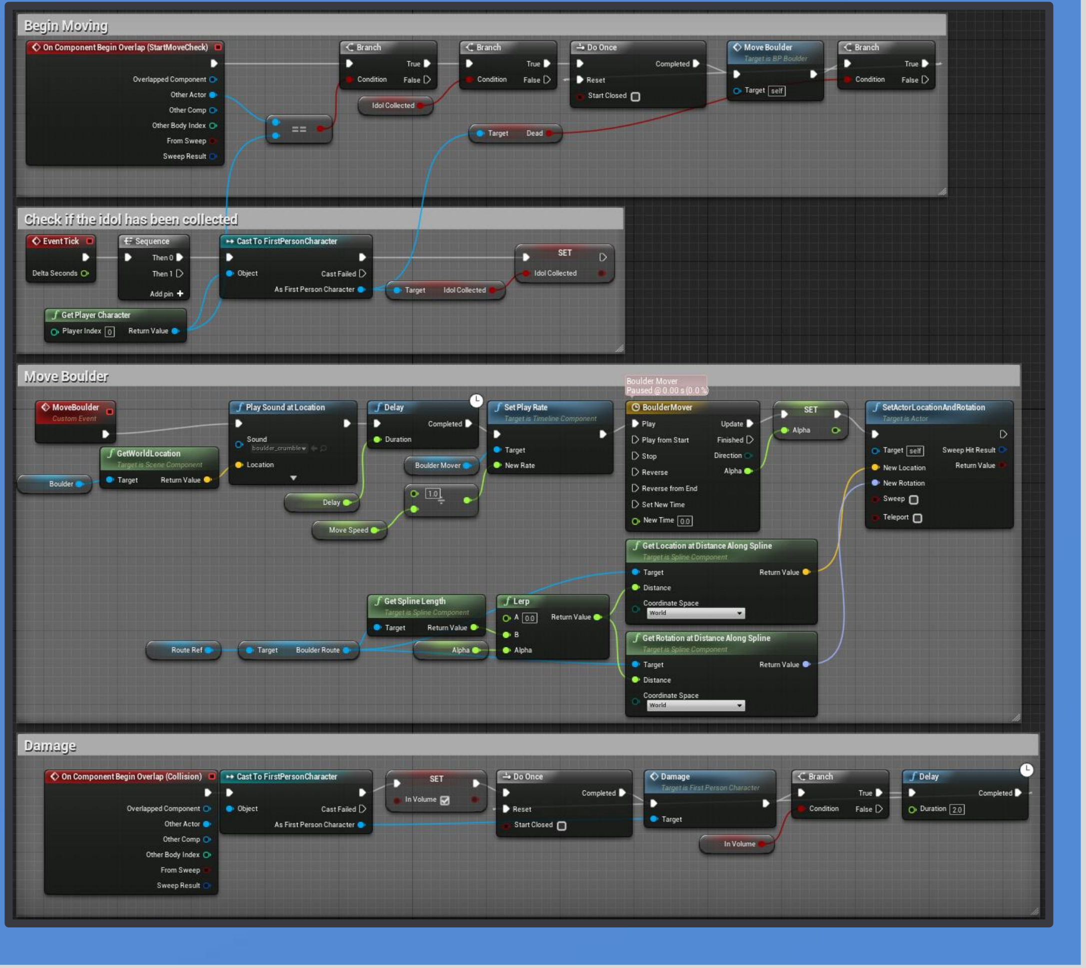


On Component Begin Overlap (Box)
Overlapped Component
Other Actor
Other Comp
Other Body Index
From Sweep
Sweep Result
Cast To FirstPersonCharacter
Object
Cast Failed
As First Person Character
Damage
Target is First Person Character
Target

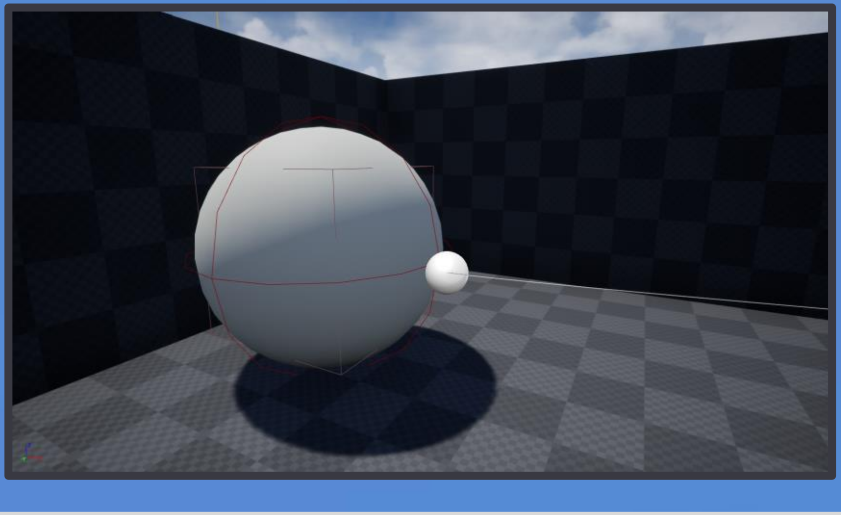
Blueprints | Boulder Chase

I decided to make the boulder move along a spline, similar to the swinging logs, however it would not loop and would follow a much longer path. Again this uses two blueprints, one for the boulder and another for the spline. As the boulder should only move on the way out, I checked whether the player had picked up the idol before starting the boulder. This meant it would only move when the player is leaving the temple. If the player is hit by the boulder they will take damage.

Blueprint Screenshot



In-Engine Screenshot



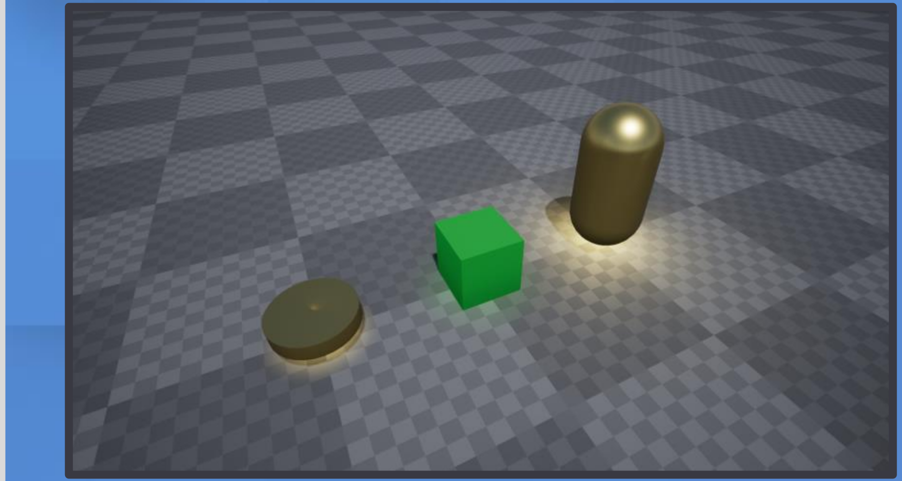
The main problem with this currently is that it only activates once. If the player dies while trying to avoid it, it will not restart its path.

Move objects along a spline: https://www.youtube.com/watch?v=bWXI91FdMtk

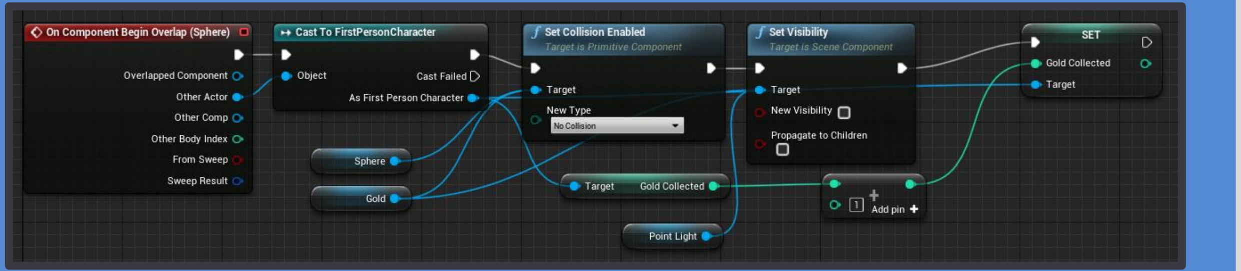
Blueprints | Collectable Items

The pickup items were fairly easy to implement. They each had three components, a mesh, a light and a sphere collision. When the player collided with them each of these items would have their collision changed and visibility set to none. Values are changed when collected such as health or gold collected increased. The health pack had an initial check and could not be picked up if the player had max health. When the idol was collected it would update the HUD subtitle to tell the player the temple is collapsing.

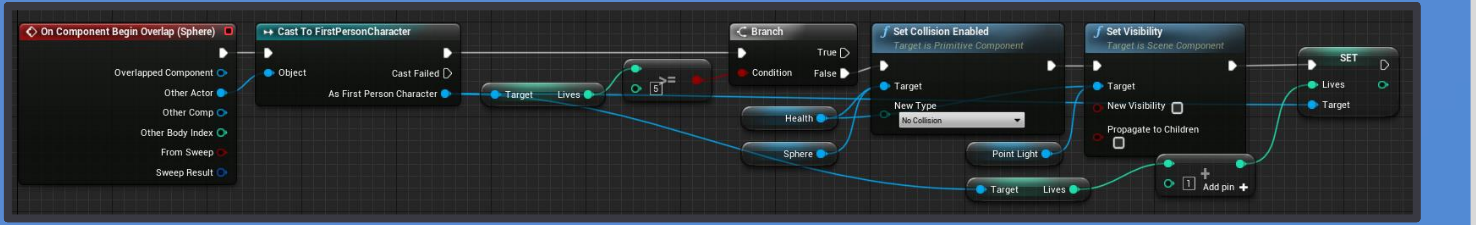
In-Engine Screenshot



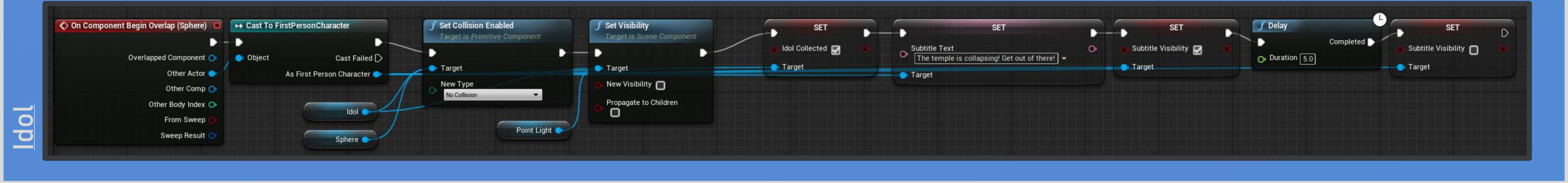
Gold



Health



Idol



Blueprints | Checkpoint

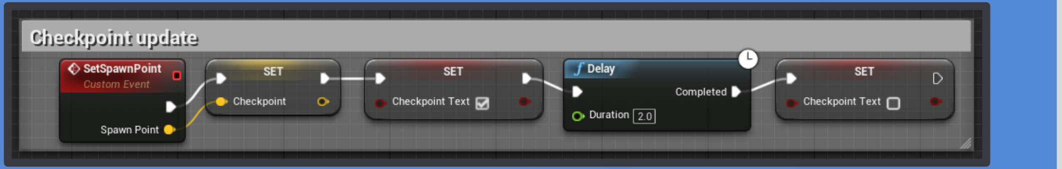
The respawn checkpoint used a box collision to detect when the player entered or left a checkpoint and would update the players spawn point appropriately.

This was done via a custom event that would set the checkpoint by inputting the players current position when run.

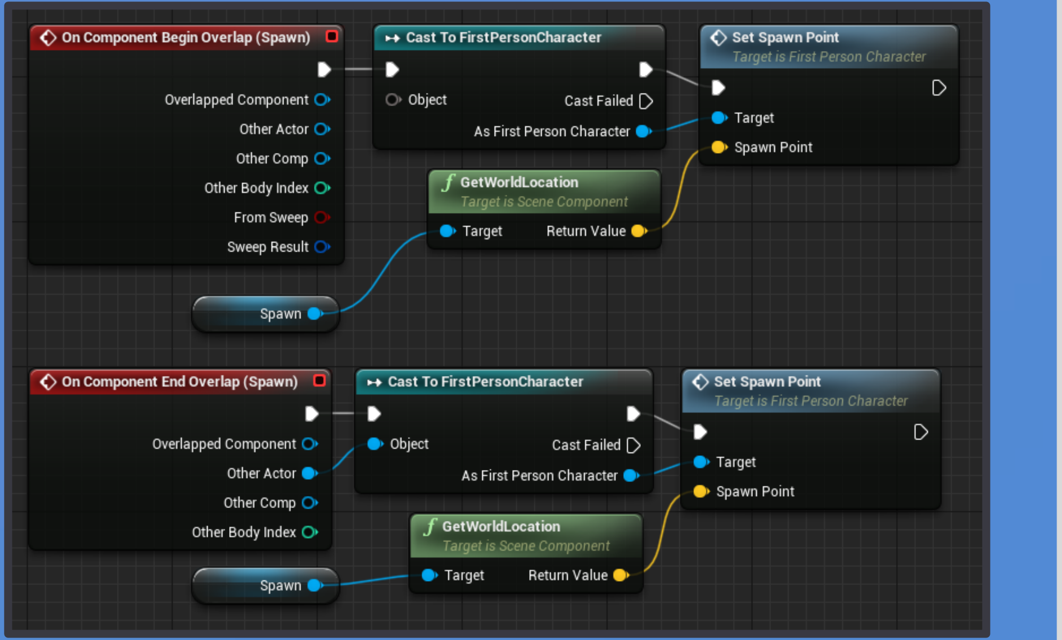
Checkpoint reached text would also be displayed on the hud temporarily.

When testing I noticed that these centre of the checkpoints could not be placed too close to the ground otherwise the player would often respawn in the ground.

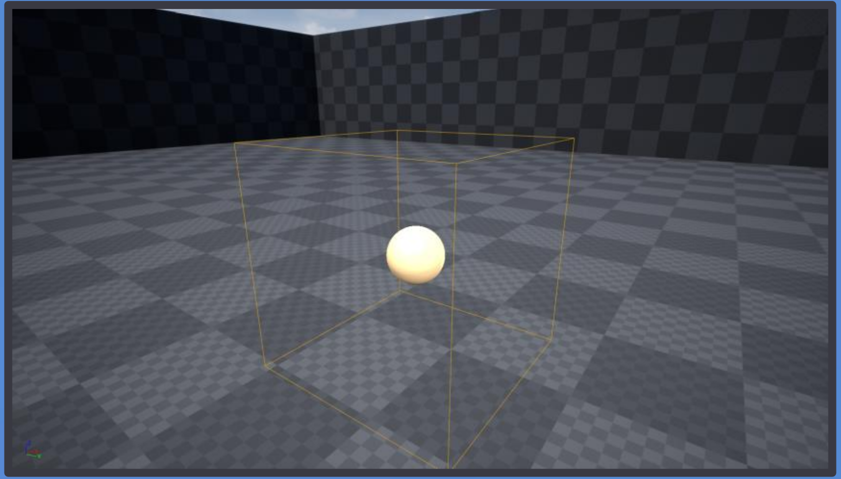
Player Blueprint



Checkpoint Blueprint

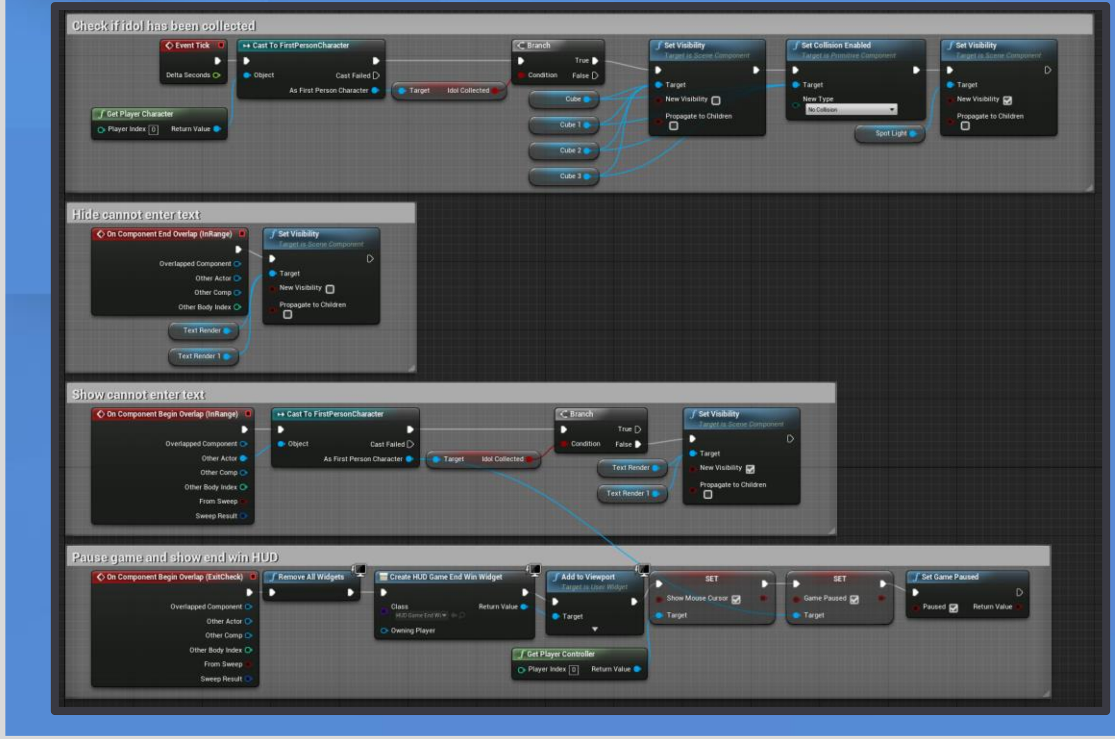


In-Engine Screenshot



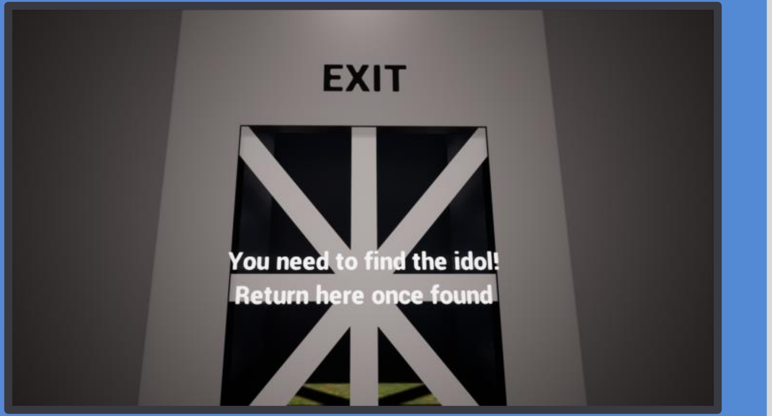
Blueprints | Exit

I needed a way for the player to leave the game after escaping the temple, so I created a small exit barrier that would unlock when the player had collected the idol. When locked if the player gets close text will appear saying the exit is locked until they find the idol. When unlocked a green light appears drawing the player in.

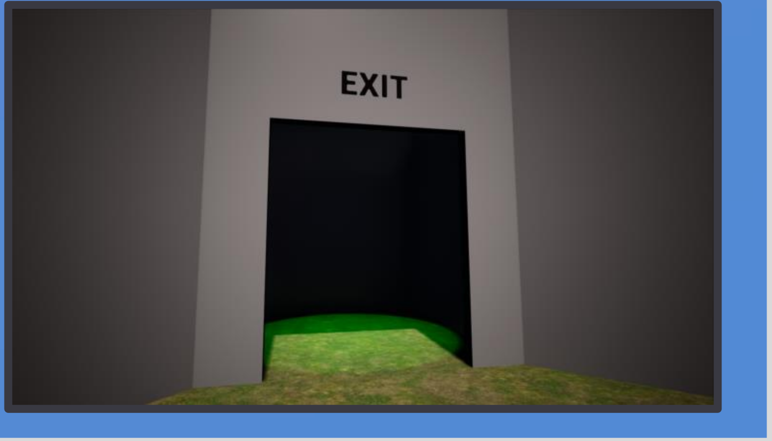


When the player goes into the exit, the win HUD will appear on screen and the game will be paused.

Exit Locked

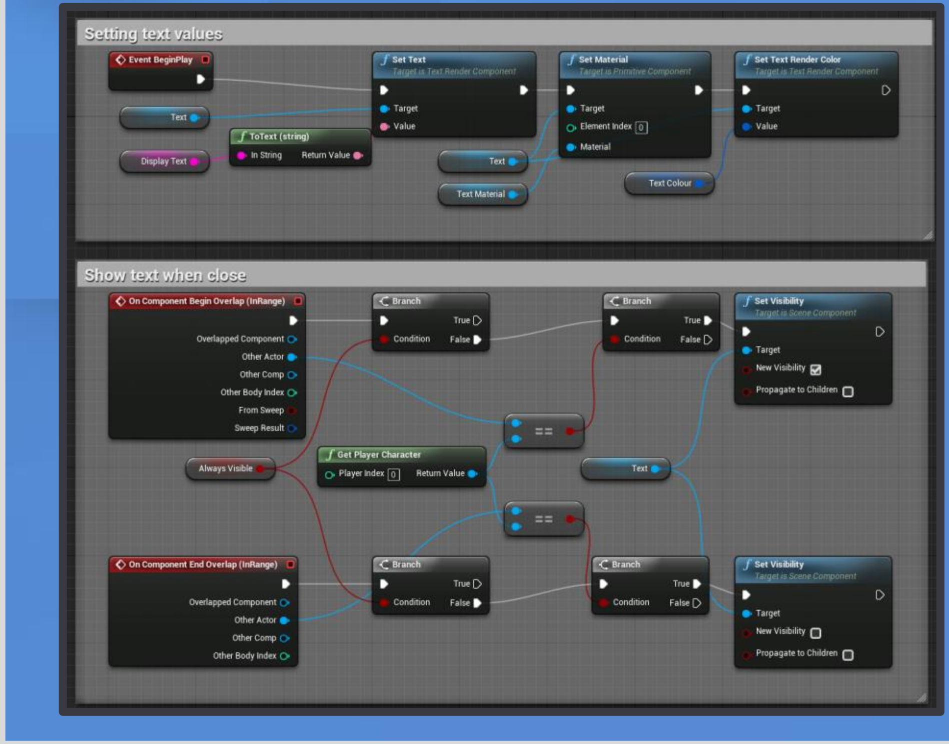


Exit Unlocked



Blueprints | Text

Throughout the level, text will be used to give the player information when needed. I did not want to put all text on the HUD and instead have some in the game world. I decided to create a blueprint for this so that if I wanted to make changes to font etc I could do it in one place.

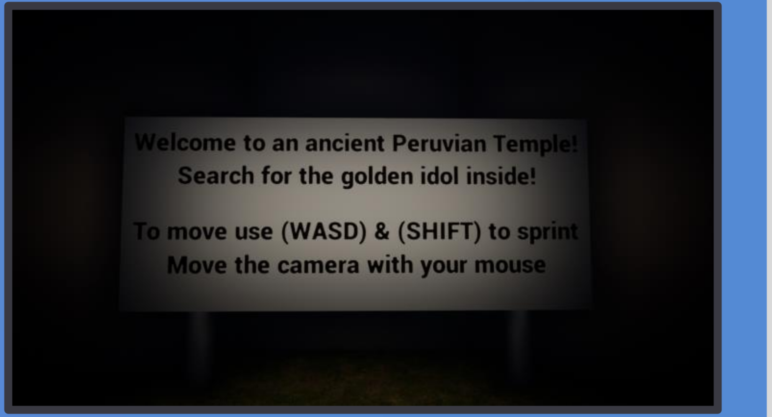


The text values are set at the start through the use of public variables.

If the text is set to always visible, the text will stay up even if the player is not in range.

There are two versions, Standard for outside and lighter areas, and emissive for inside and darker areas where standard may be hard to read.

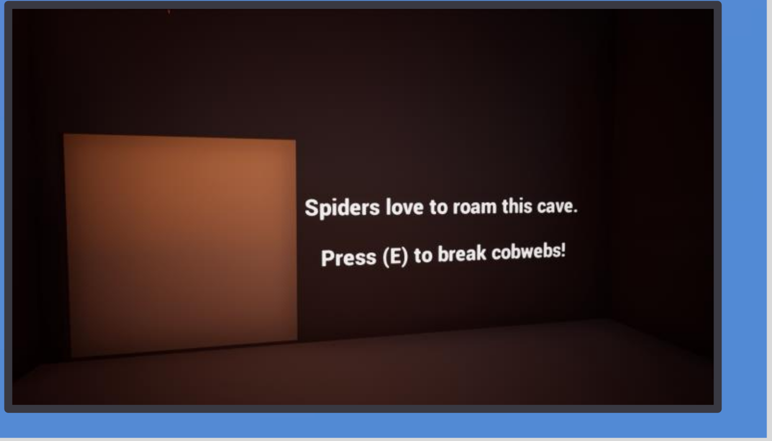
Text (Standard)



Welcome to an ancient Peruvian Temple!
Search for the golden idol inside!

To move use (WASD) & (SHIFT) to sprint
Move the camera with your mouse

Text (Emissive)



Spiders love to roam this cave.

Press (E) to break cobwebs!

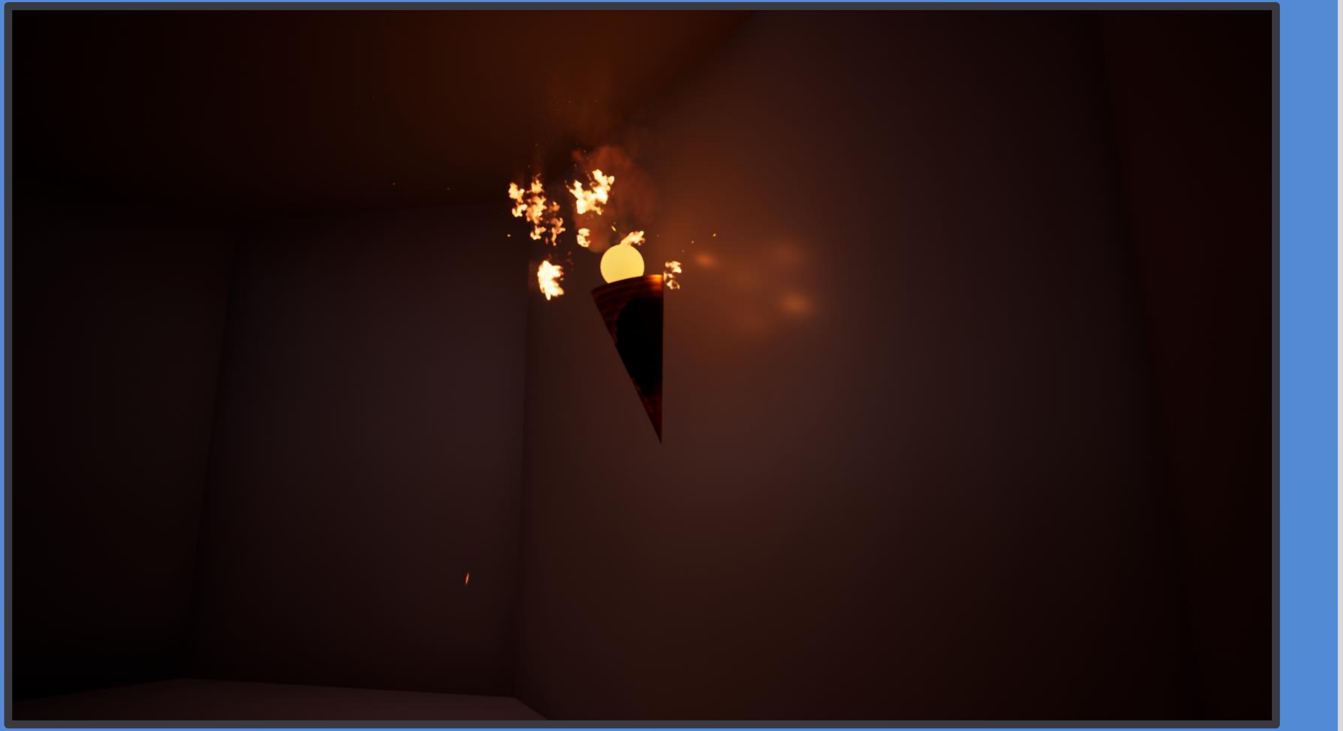
Blueprints | Torch

I needed lighting in the cave so I created a torch blueprint using a cone, a sphere, a point light and a fire particle effect. I chose this torch design as it would help the temple feel more natural.

I also made a fire crackling audio effect play whenever the player gets near a torch to immerse the player into the game.

These torches can be used to guide the player in the intended direction of travel.

In-Engine Screenshot

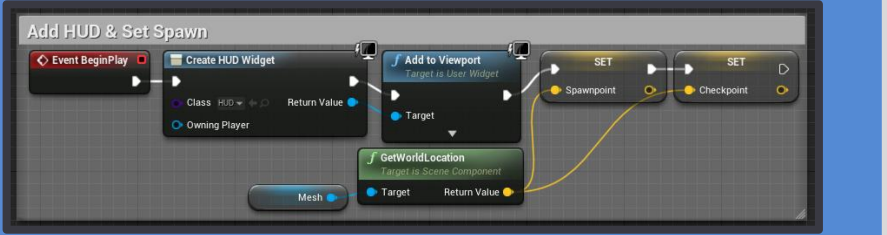


Blueprints | Game HUD

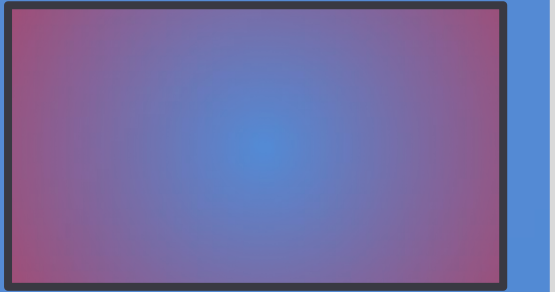
I opted for a minimal game HUD as I didn't want to distract the player from the game experience. The players lives are shown top left and gold collected is top right.

The checkpoint and subtitle text is not always shown and will appear when needed.

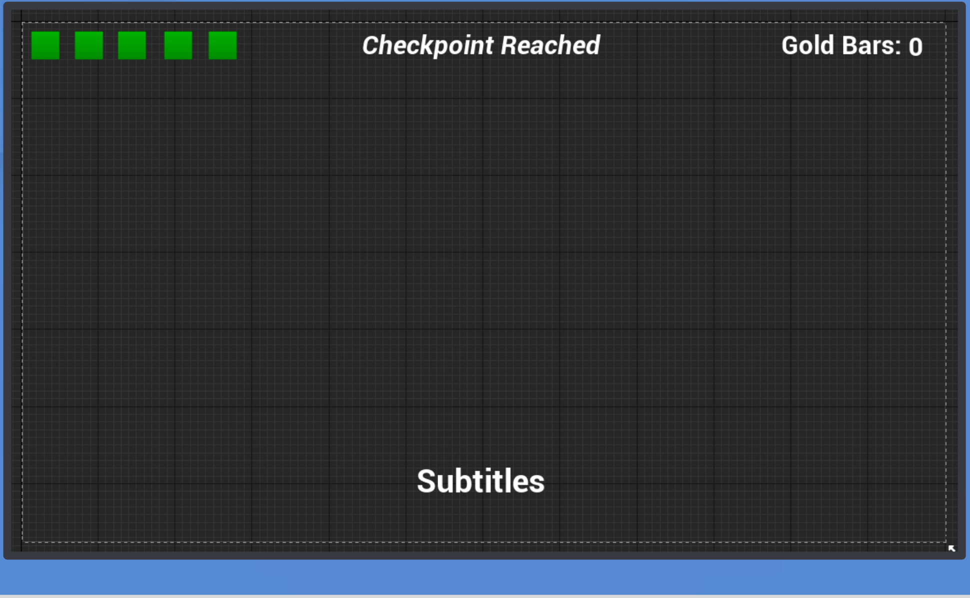
Initialise HUD (First Person Character)



Red Vignette

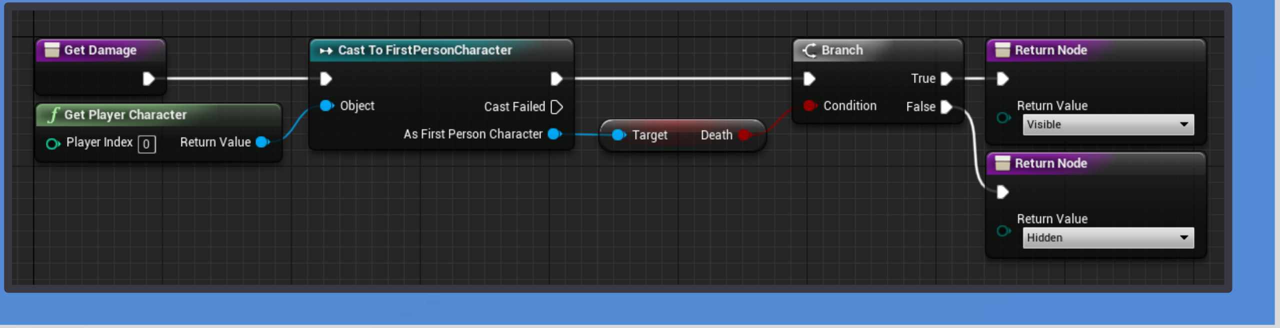


In-Engine Screenshot



Damage Effect

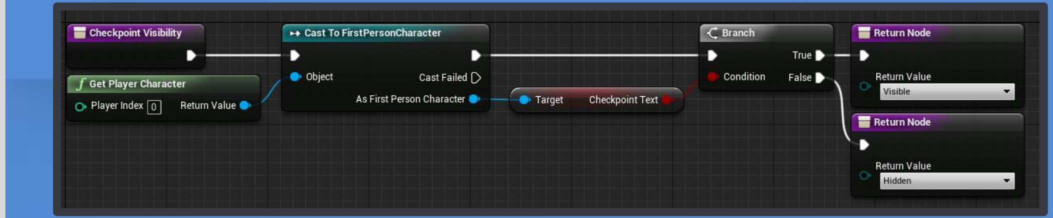
When the player takes damage, a red vignette will be shown on the HUD for a few seconds. This is to show more obviously to the player that they have taken damage.



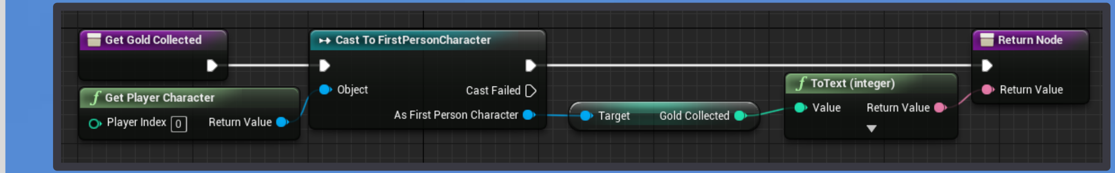
Blueprints | Game HUD

Different parts of the game HUD needed updating at various sections of the game, mainly regarding whether to be visible or not. Whether a HUD component should be visible is controlled via the first person character, so each component casts to the first person character and collects the relevant variable and updates the HUD accordingly.

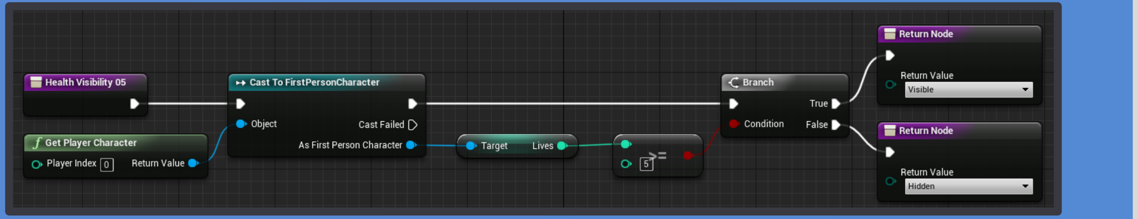
Checkpoint Visibility



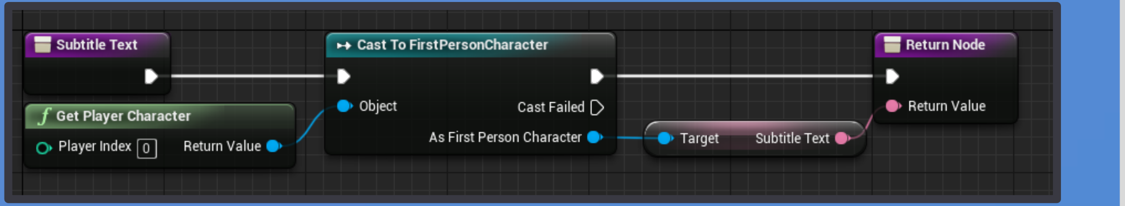
Gold Value



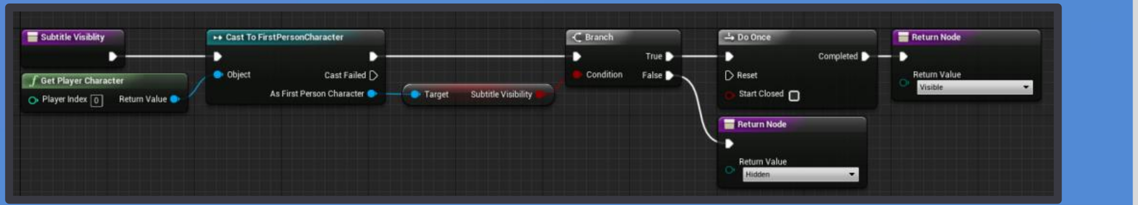
Health Visibility



Subtitle Text Value



Subtitle Visibility



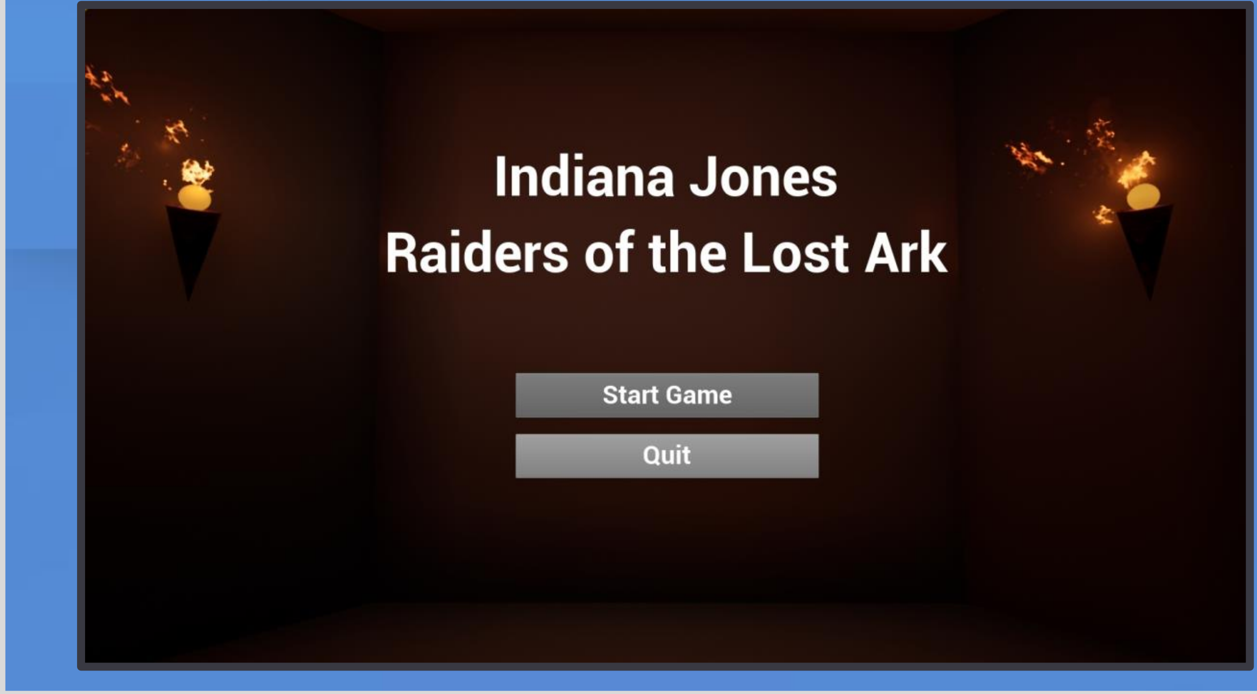
Blueprints | Main Menu

I decided to create a separate level for my main menu and have a cave wall backdrop. There are two buttons on the main menu: Start game and Quit.

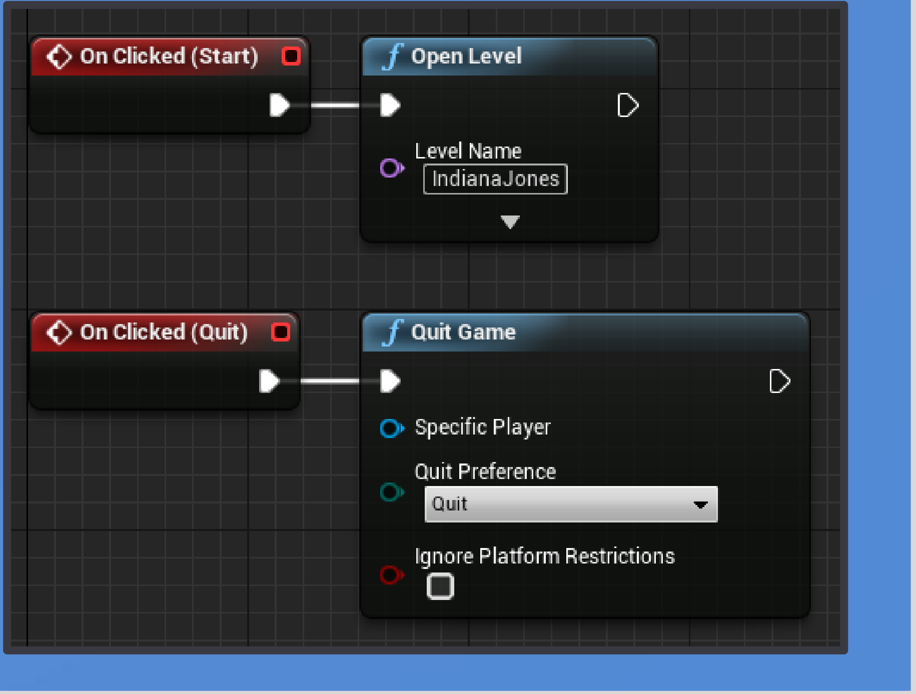
When start game is clicked, the main level is opened which starts the gameplay.

When quit game is clicked, the game will close completely.

In-Game Screenshot



Main Menu Blueprint



On Clicked (Start) → Open Level

Level Name: IndianaJones

On Clicked (Quit) → Quit Game

Specific Player

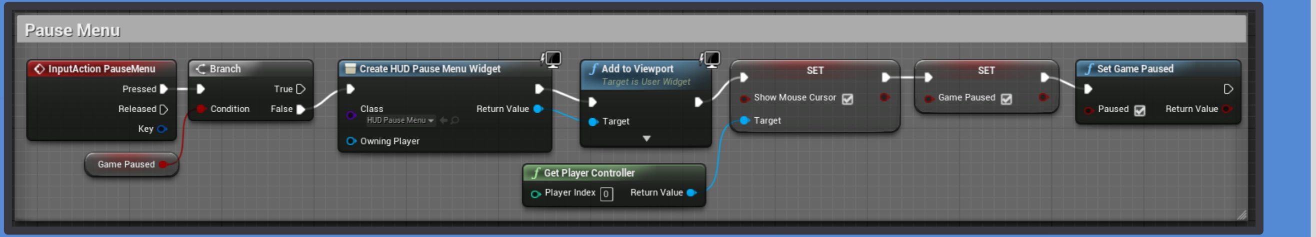
Quit Preference: Quit

Ignore Platform Restrictions: □

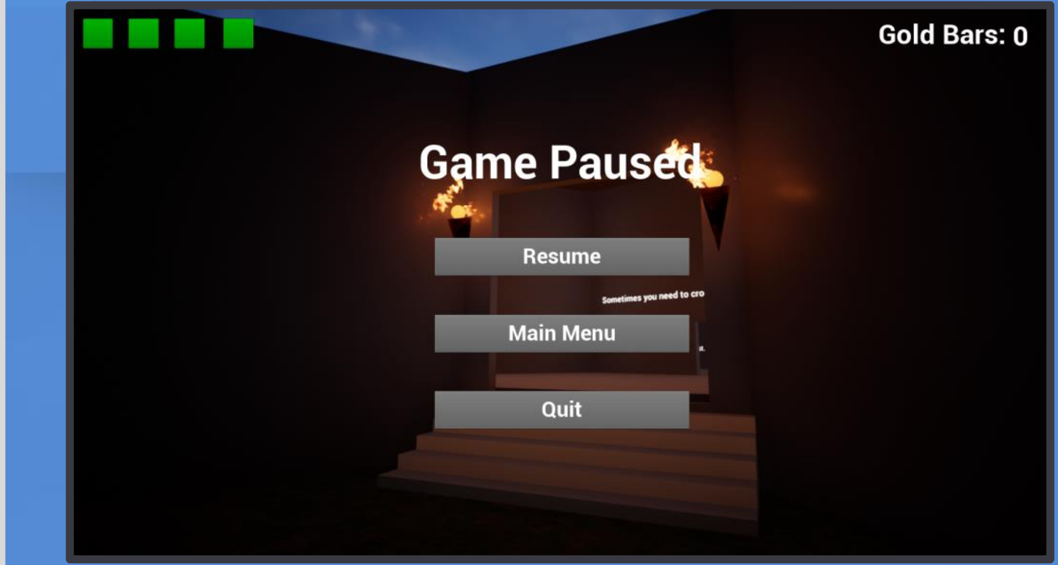
Blueprints | Pause Menu

The pause menu can be accessed by pressing P or ESCAPE (run from character blueprint), the game will be paused so the player can take a break if needed. They can also access the main menu or quit the game from here. There is a resume button for when they are ready to play again.

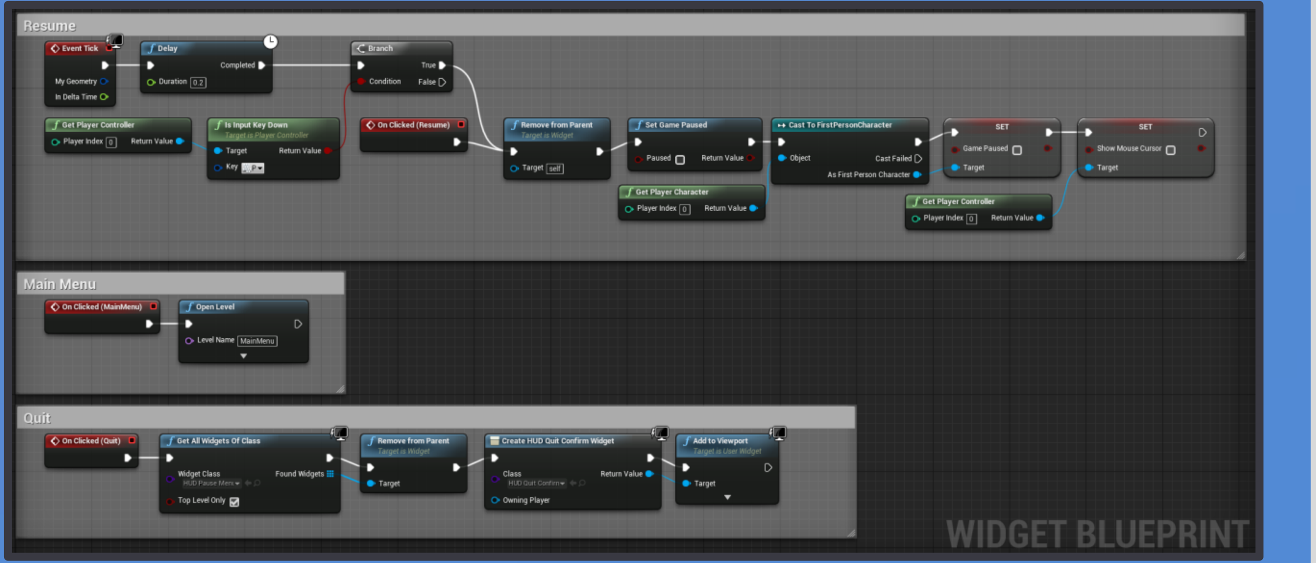
First Person Character Blueprint



In-Game Screenshot



Blueprint Screenshot

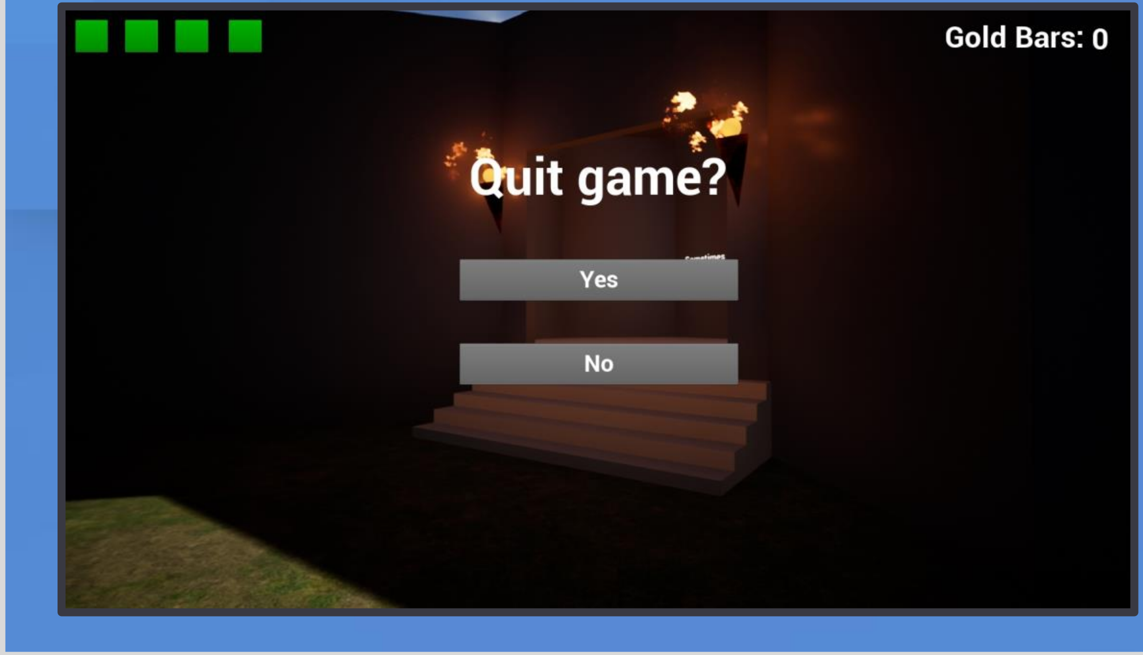


Blueprints | Quit Menu

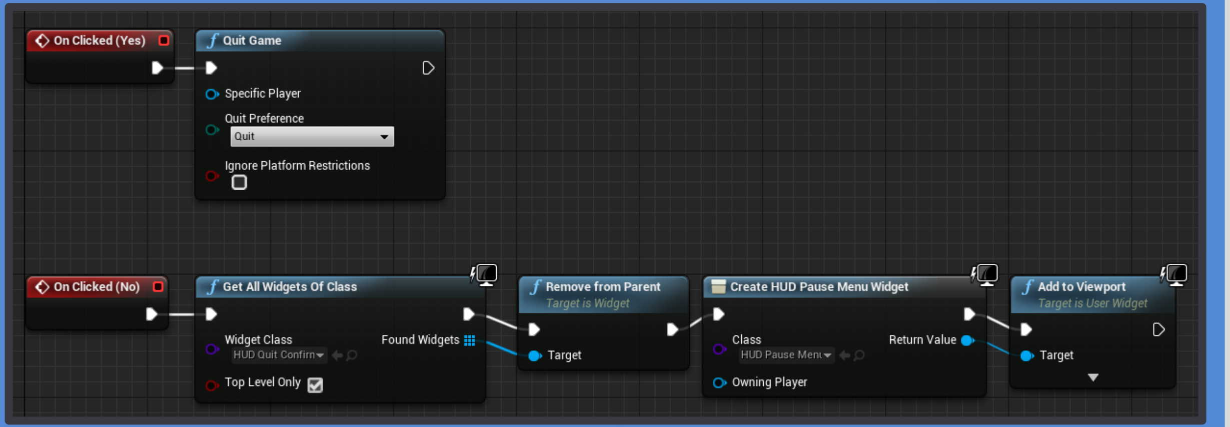
When quitting the game I added an option to confirm whether you wanted to quit so that the player does not close the game by accident.

This was to stop the player accidentally quitting the game halfway through a playthrough.

In-Game Screenshot



Blueprint Screenshot

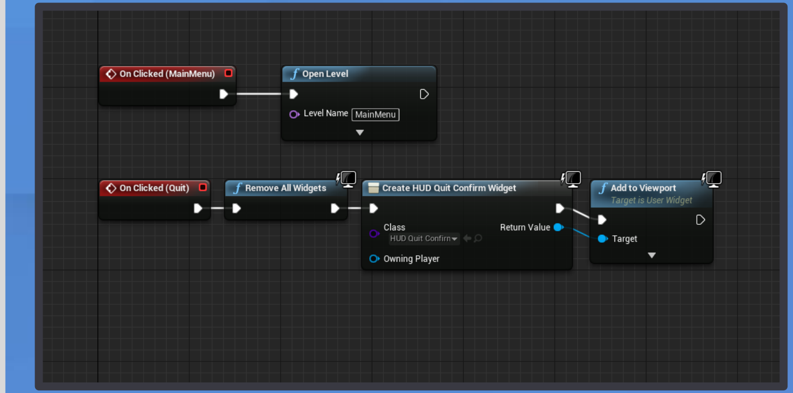


Blueprints | Ending Menus

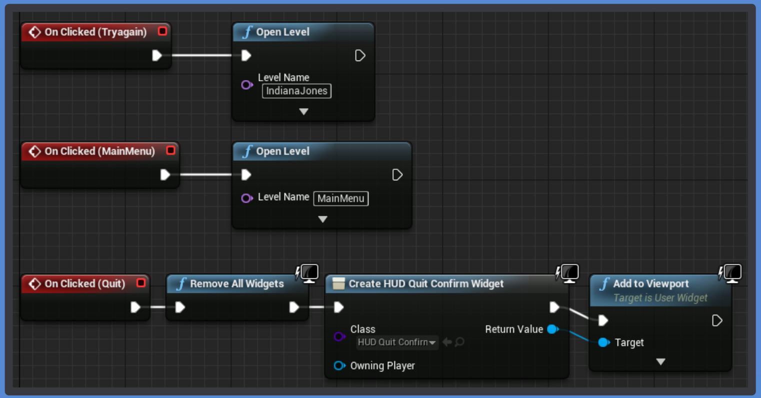
I created two menus for the ending scenarios, win and lose. Both menus will have a main menu and quit option, but the lose menu will have a retry button which will restart the level from the beginning if the player wishes to try again.

The menu will also show how many gold pieces the player has collected. This will act as the players score and they can try the level again to improve their score.

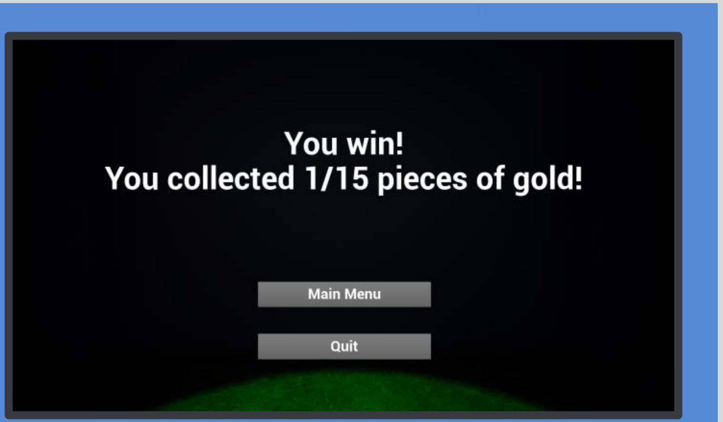
Win Menu Blueprint



Lose Menu Blueprint

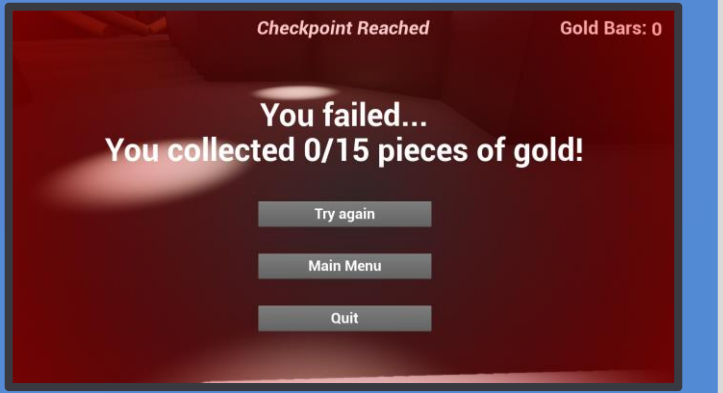


Win Menu



You win!
You collected 1/15 pieces of gold!

Lose Menu

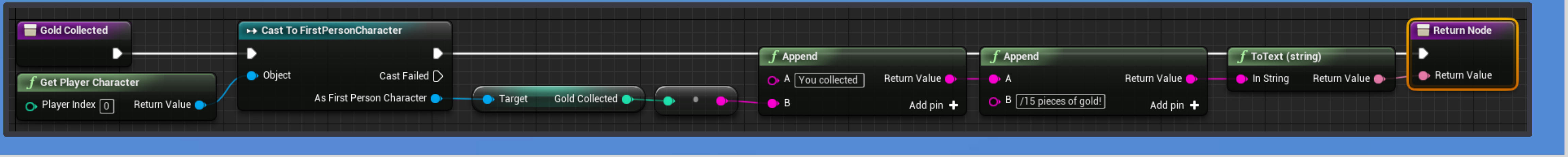


Checkpoint Reached

Gold Bars: 0

You failed...
You collected 0/15 pieces of gold!

Gold Value

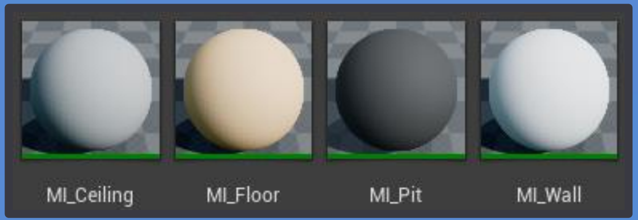


Red Vignette Effect: https://www.pngkey.com/maxpic/u2y3q8q8r5u2w7r5/

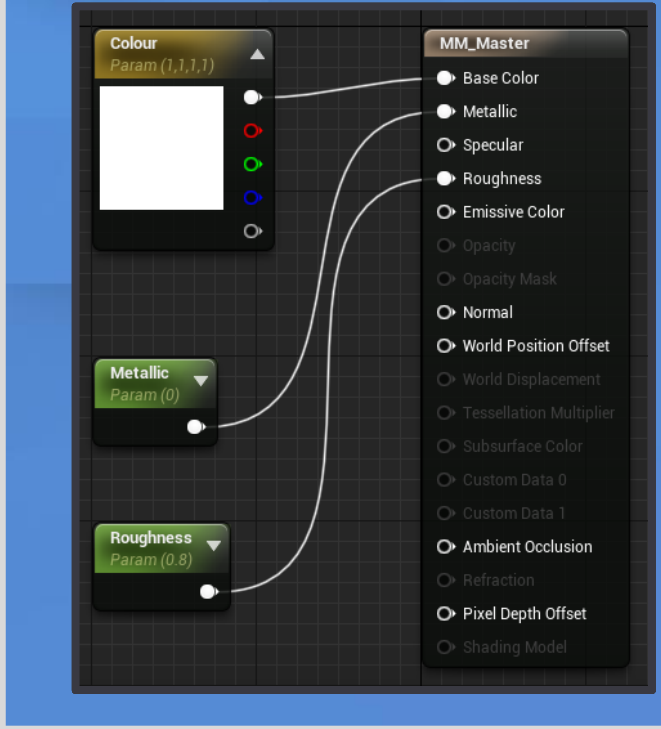
Materials | Geometry

I created a master material for my geometry as this would allow me to quickly create each material for floor, walls and ceiling as well as any others I needed in the project.

Material Instances



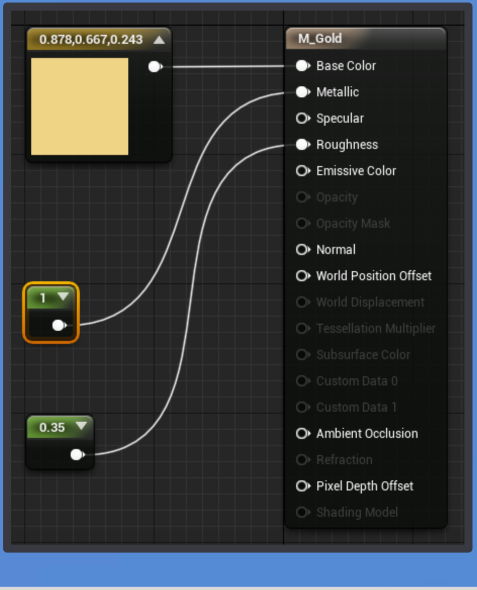
Master Material



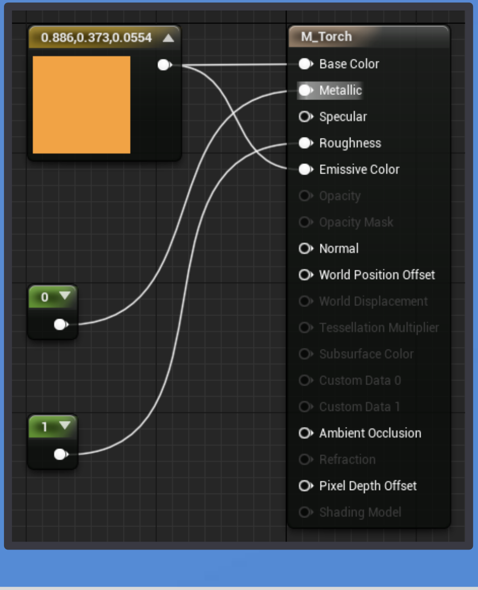
Other Materials

I also created some other materials to use for health packs, gold and the torch. These needed more differing values so I decided against using a material instance for these. The torch and health material used emissive values to give a glow to the objects with the material.

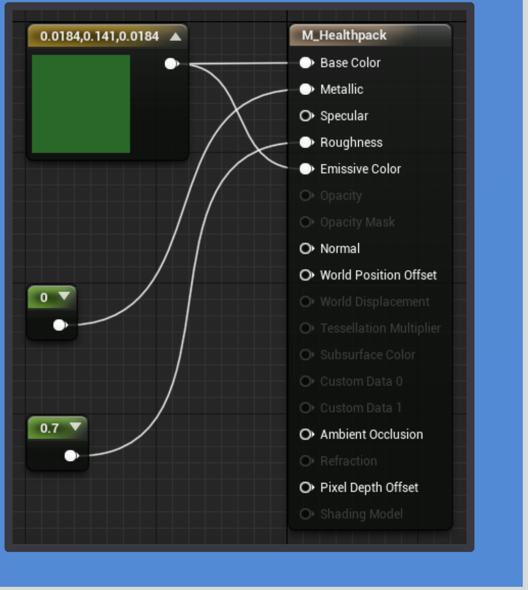
Gold Material



Torch Material



Health Material

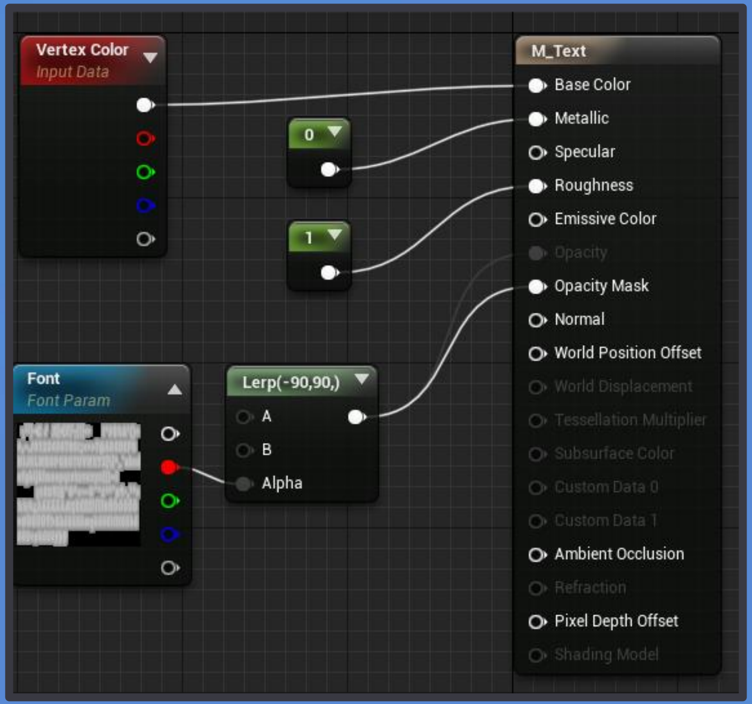


Materials | Text Variations

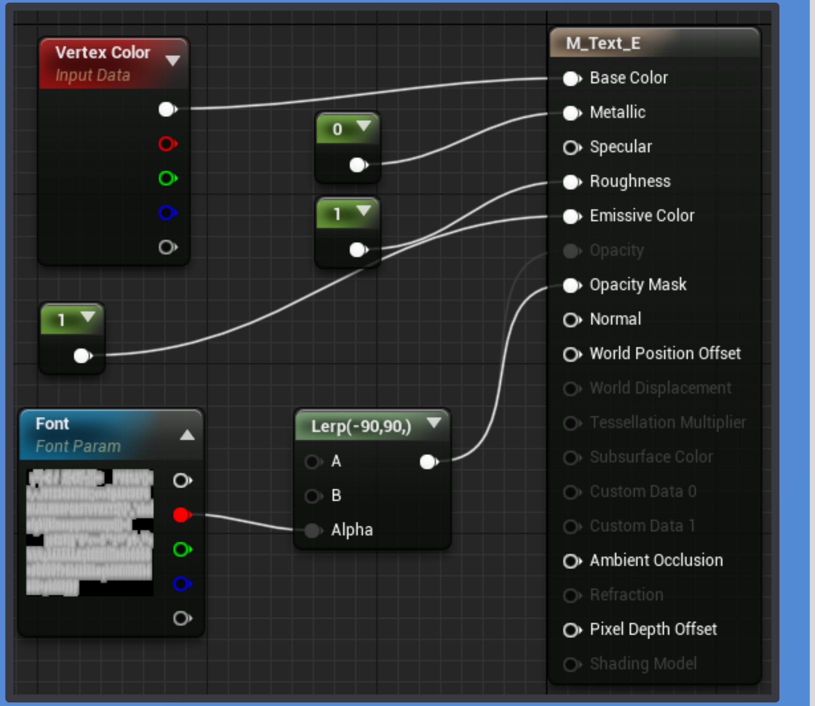
Text Variants

When I added text to my levels, I noticed that when in the cave the text was difficult to read in some areas. One solution I thought about was adding lighting around the text, however this often looked artificial and didn't fit the theme of the cave. I decided instead to make an emissive version of the text material to use inside the cave for better readability.

To do this I created copies of the standard text material from Unreal Engine and gave one version an emissive value of 1 to light it up.



Text - Standard

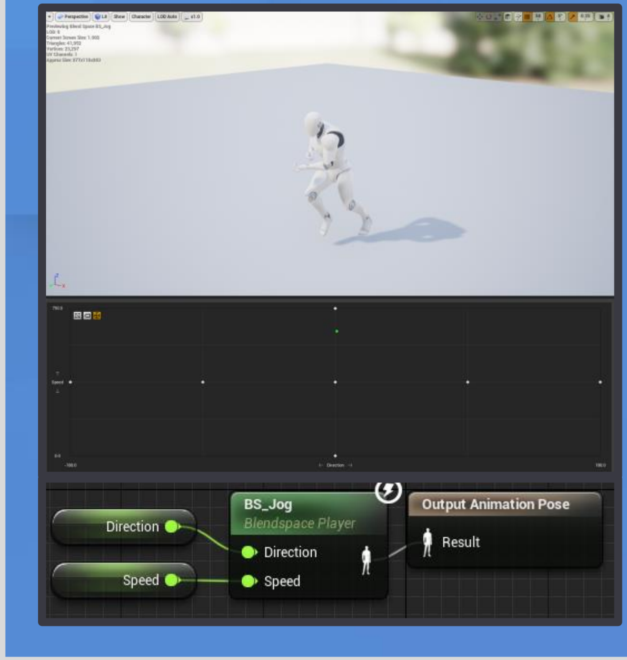


Text - Emissive

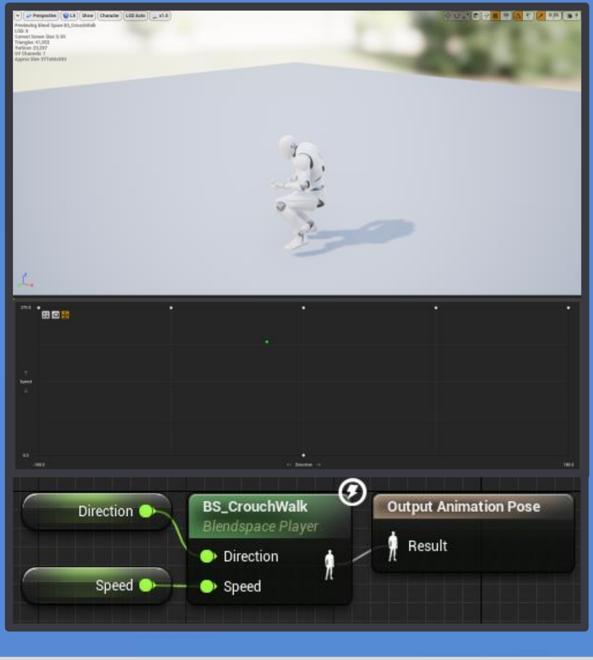
Animation | Player

I needed some basic animations for my player for idle, walking, running and crouching. To do this I downloaded Unreal Engines animation starter pack to use. I created three blend spaces for the moving animations and made transitions between the different states. While the pack did not have a dedicated hanging animation, so I used a pistol idle animation that worked for the character holding the rope. This was activated whenever the player was hanging off the rope by checking the hanging boolean in the character blueprint.

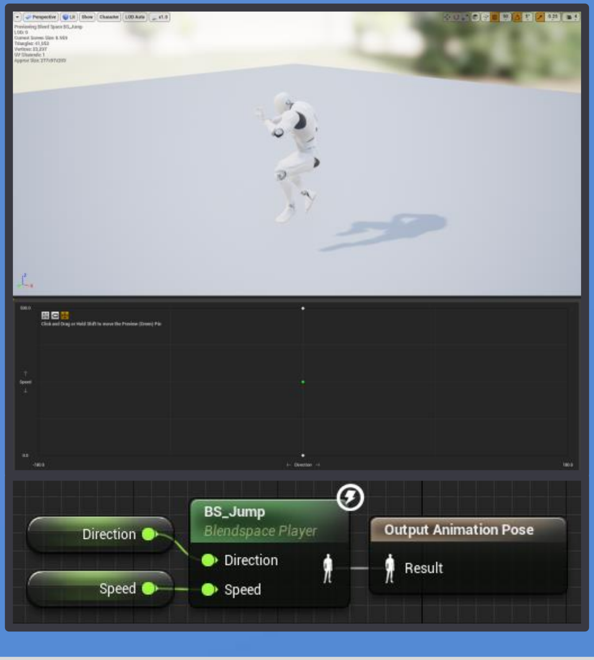
Walk/Run Blend space



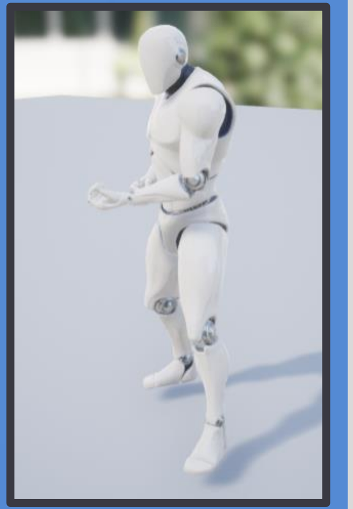
Crouch Walk Blend space



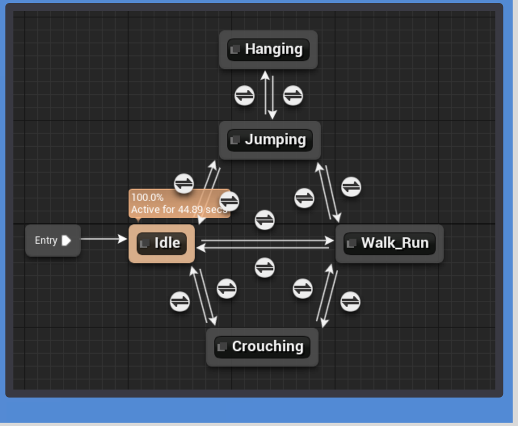
Jump Blend space



Idle Stance



Animation Transitions



Feedback

As I had a mostly complete version of my game ready for playtesting I built the project and began asking my classmates and house mates for feedback to help me improve the game. The next few pages will go through some of this feedback and the key changes I made to improve my level as a whole.

The most common feedback was that the game overall was very difficult to complete. While iterating on the project, I plan to make it easier for players to progress as many testers could not finish the game in its current state. This had been difficult to test myself as I had played through it many times and gotten used to the difficulty over time.

Another area that needed improvement was the layout of the temple, as there were some sections where players were getting confused with orientation and the intended direction of travel. Some of the gold locations did not work well either and needed to be placed in areas requiring more player skill to reach, while keeping the main path easier to complete. Some players did find it confusing which way was the correct way, however I did want the player to explore the temple in order to find the gold. However it did make finding the idol room challenging as it was easy to get lost in the temple. Players did mention that the level did give off the feeling of a temple very well.

Due to the layout and difficulty of the level needing improvements, I decided to recreate the layout of the temple to better fit these. The main improvements were:

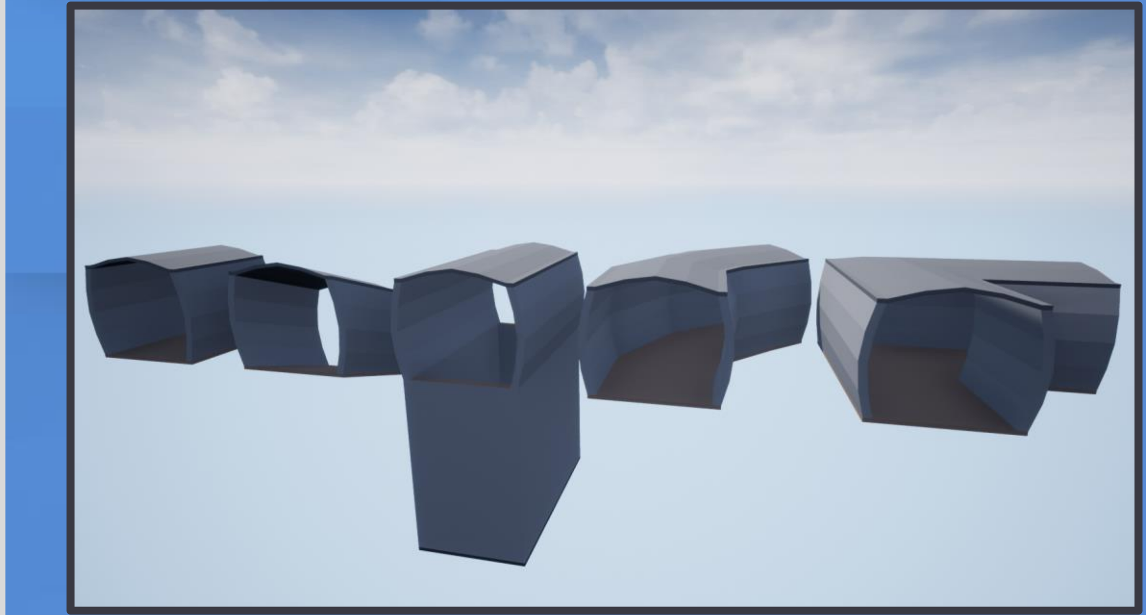
- More obvious and easier main path to navigate.
- Gold and health packs in challenging locations, to allow the main path to be easier.

With these two key points in mind, I returned to Unreal Engine to improve the level as well as implement other smaller changes brought up during feedback.

Iterations | Layout v4.0

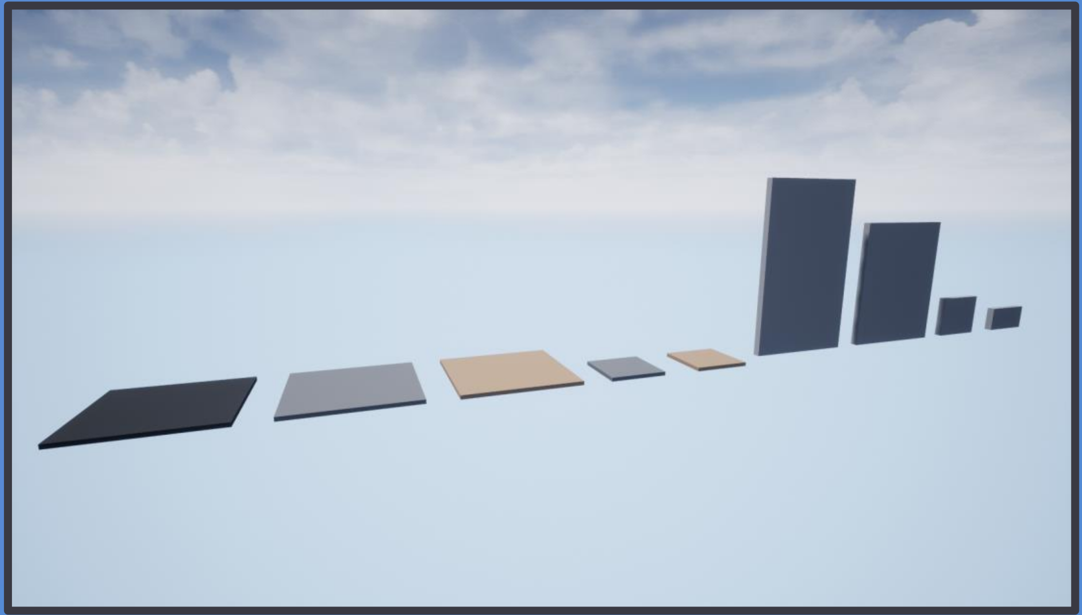
As I was planning to rework the layout, I redesigned some of my block out meshes to better fit the temple theme. One piece of feedback I had received was that it was difficult to differentiate between regular and crouch doors. I have made the regular doors larger to make it more obvious. I created the corridors using Blender and imported them into Unreal engine. These meshes fit together well and overall felt much more natural than the previous layout.

Layout v4.0 (In-Engine: Corridor Block Out)



Left -> Right: Corridor, Slope, Pit, 45° turn, 90° turn

Layout v4.0 (In-Engine: Room Block Out)

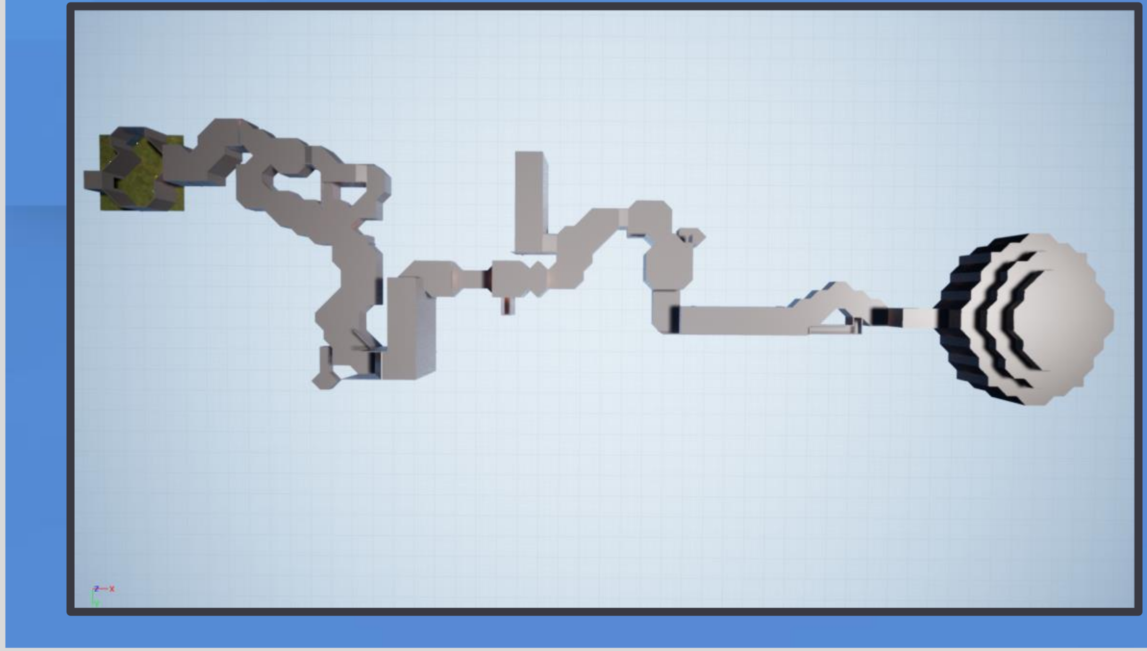


Left -> Right: Pit Floor, Ceiling, Floor, Ceiling (Small), Floor (Small), Wall, Wall (Medium), Wall (Small), Wall (Crouch)

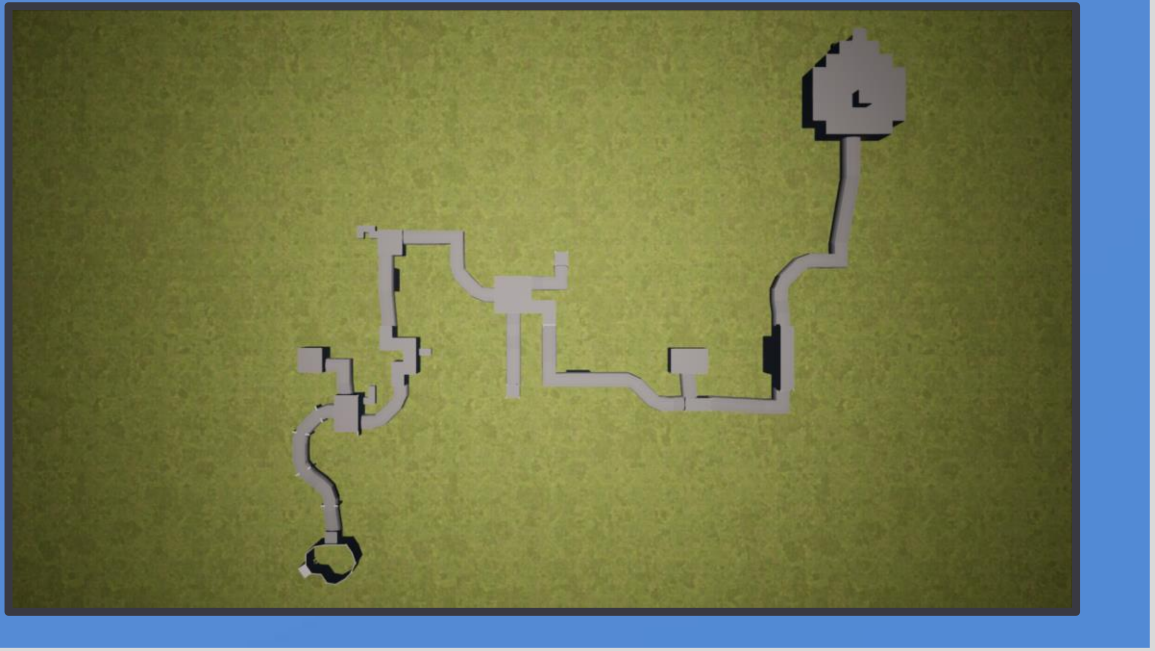
Iterations | Layout v3.0 / v4.0

I then set about making a layout for v4.0, taking a more linear approach with a main path using the corridor meshes with paths off to the side for the player to explore if desired. This made the main path more obvious and each route had a reason to explore them. The layout did change a fair amount but for the most part the traps remained in the same order as the previous layout, however I removed one of the boulders and relocated the other to the idol room where it will activate when the player leaves. It now rolls to the next pit where it falls in and stops chasing the player.

Layout v3.0 (In-Engine: Top Down)



Layout v4.0 (In-Engine: Top Down)



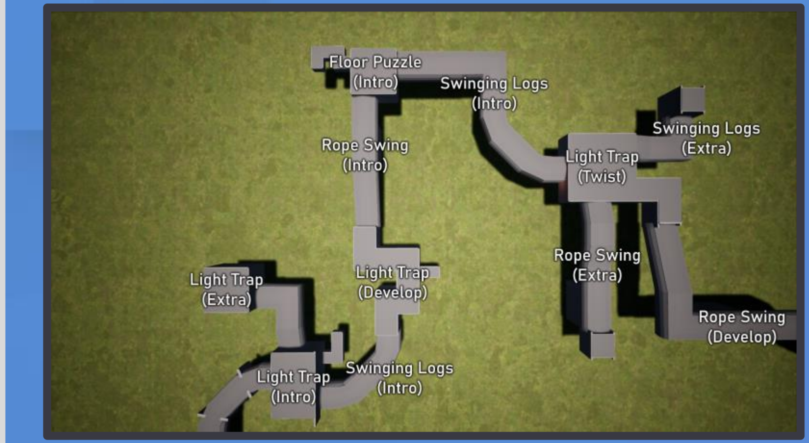
Iterations | Layout v4.0 (Traps)

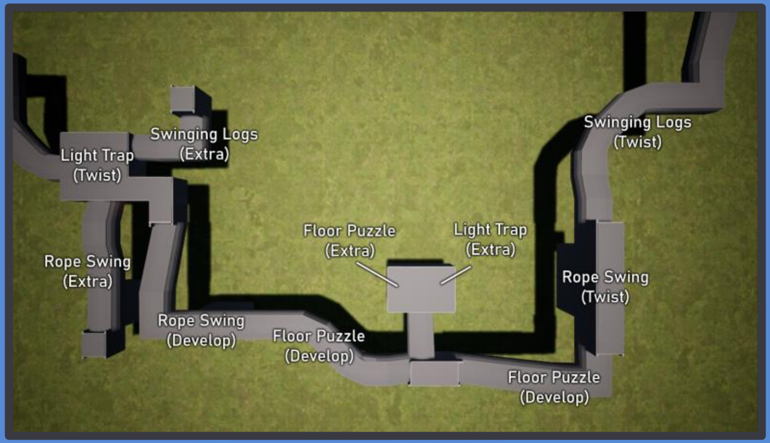
The traps followed mostly the same order as v3.0, however they were spaced differently due to the new layout.

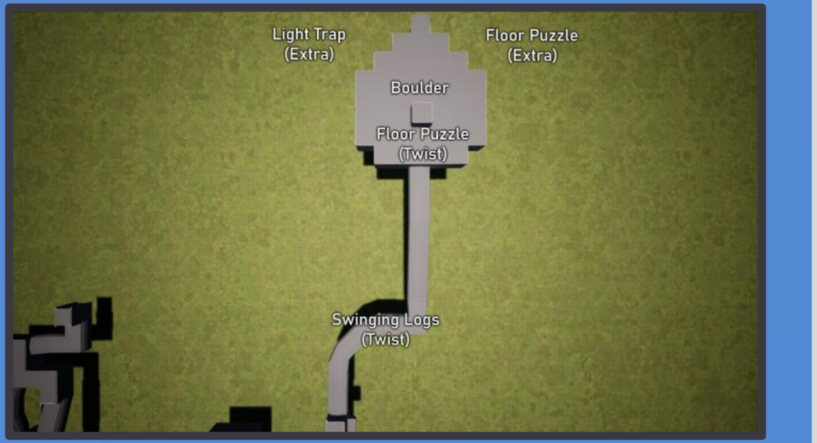
I also added an extra rope swing to the main path to force the player to develop their skills.

One boulder was removed and placed in the idol room at the end of the level as it felt better placed there than anywhere else.

Extra versions of the swinging logs, light traps and floor puzzles were scattered about the level in areas of the main path to challenge the player to collect gold or health packs.





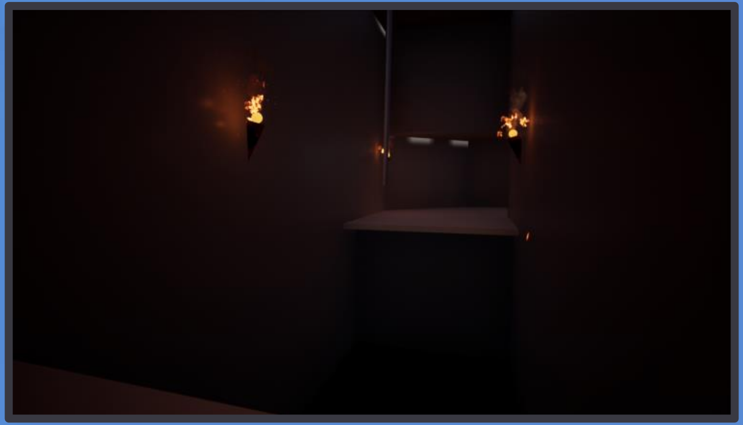


Iterations | Rope Swing

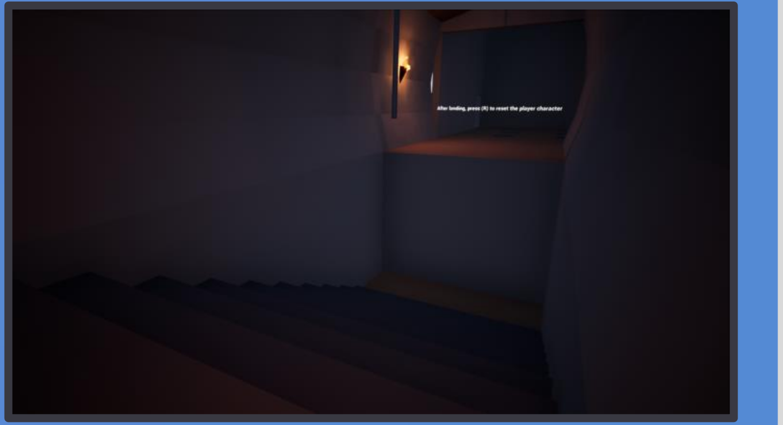
An area of feedback I received related to the rope swing and how it was not explained how to swing on the rope. It took the player some guess work to figure out how to swing.

The rope swinging was also a hard mechanic for players to get used to and the first time the player approaches it they have little room for error.

Before Feedback



After Feedback (Stairs)



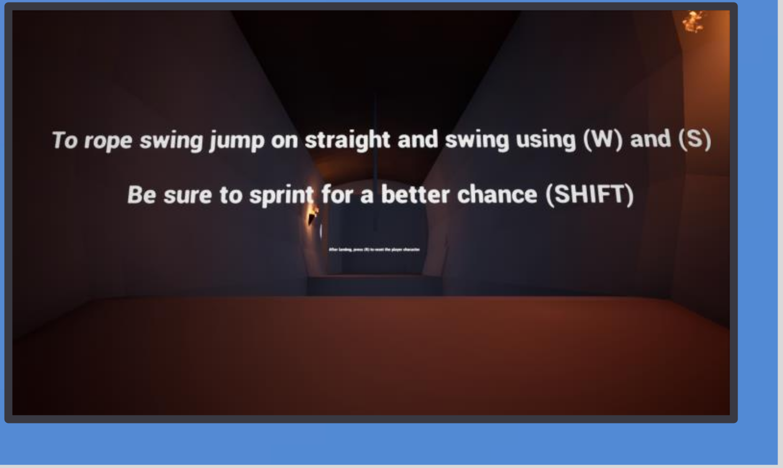
After landing, press (R) to reset the player character

To help players find more success with the rope swing I first added a stairway back up to the first rope swing to allow the player to retry as many times as necessary.

Text was also added before the rope swing telling the player the controls when on the swing and also a tip to sprint before jumping to have a better chance of catching the rope. I also added text after recommending to reset the character due to the rope swing offset issue mentioned earlier.

Overall the rope swing has been more appropriately introduced to the player and is easier for them to pick up the mechanics.

After Feedback (Text)



To rope swing jump on straight and swing using (W) and (S)

Be sure to sprint for a better chance (SHIFT)

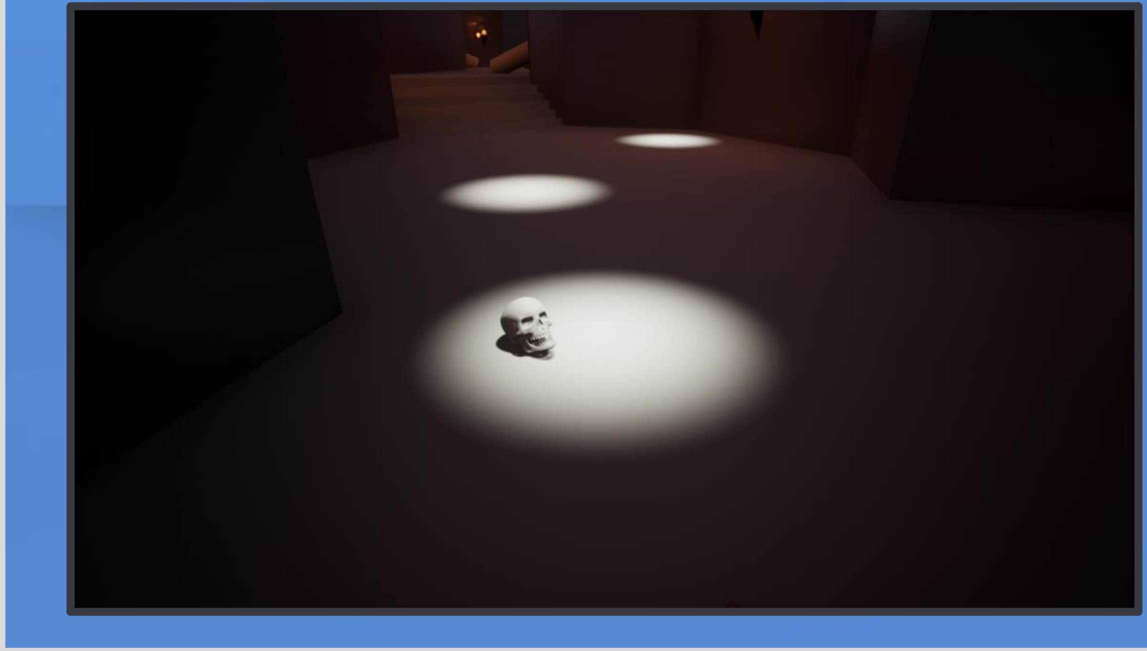
After landing, press (R) to reset the player character

Iterations | Light Trap

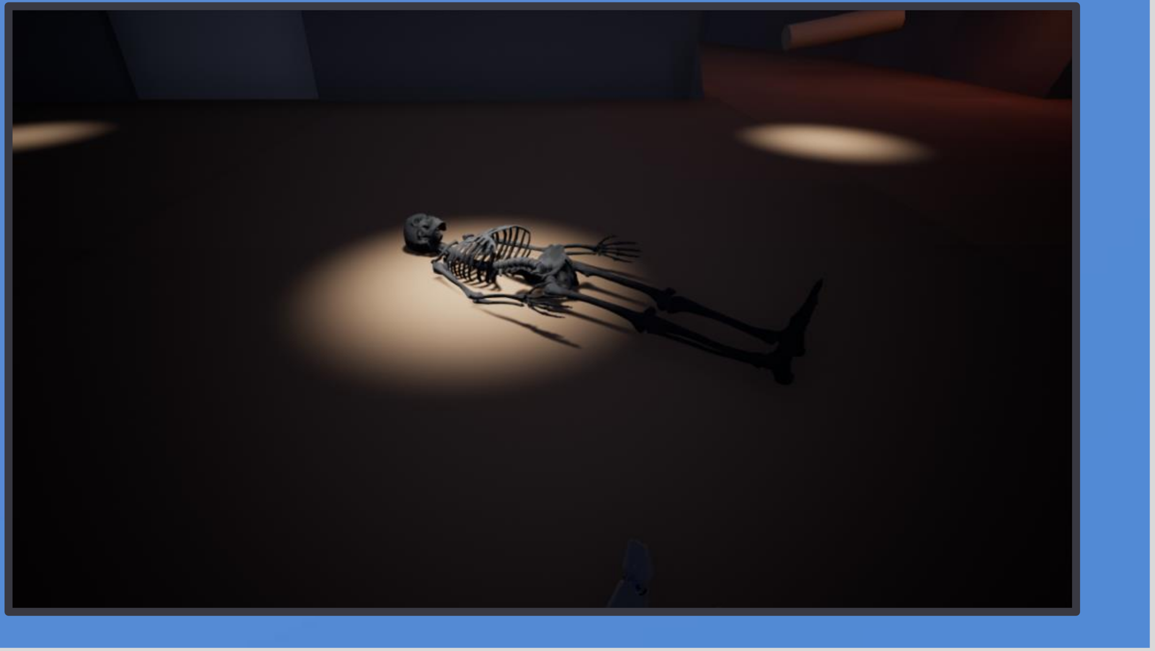
The skull in the first light trap was noted to look artificially placed and recommended to replace the asset with a skeleton for a better look and help introduce the player to the light trap.

I have changed the asset to a skeleton as suggested and left it half in and half out of the light. This does appear more obvious to the player and they should not miss the hint that light should be avoided.

Before Feedback



After Feedback

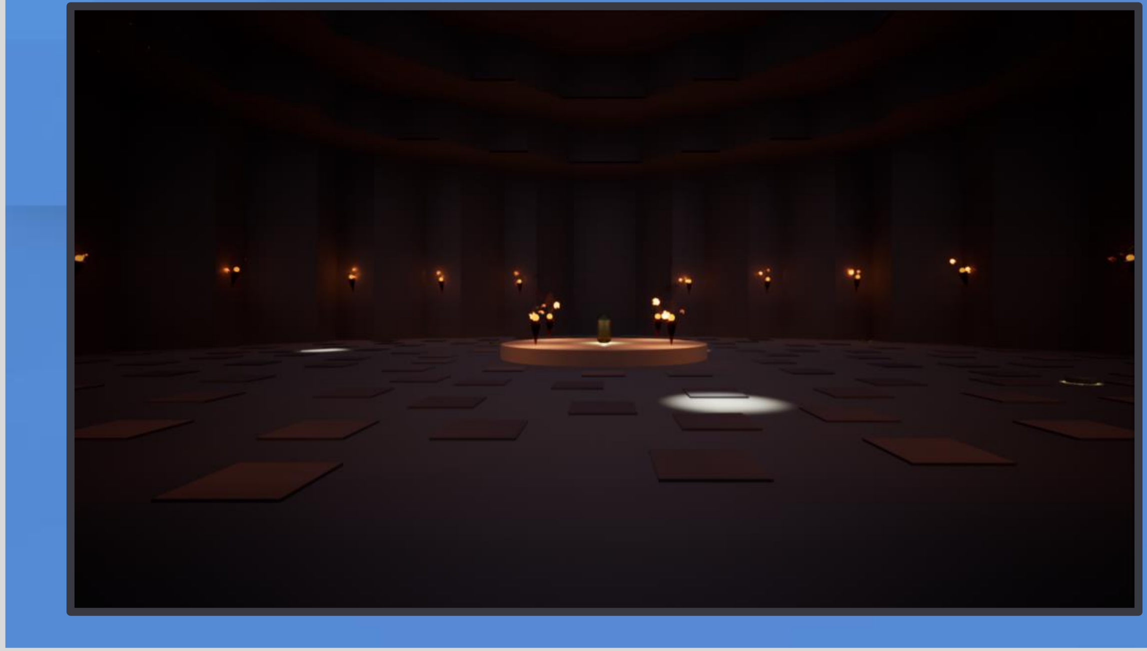


Iterations | Idol Room

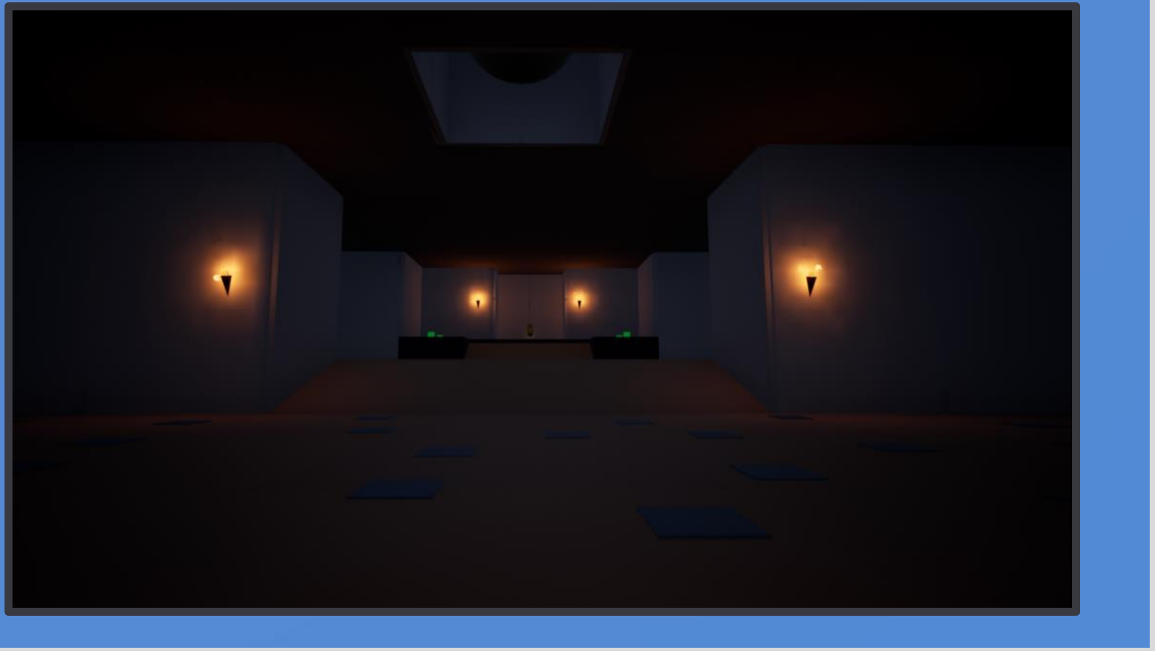
Feedback told me that the idol room was too symmetrical and lacked landmarks needed to orientate the player. Some players felt lost at points in the room and took a bit to find the exit after collecting the idol.

To fix this I redesigned the idol room to have some landmarks such as a back wall where the idol sits along with many health packs to reset the players health. The boulder is also seen in the ceiling for those who look out for it. The player can find various bits of gold hidden around the idol room for extra points. Overall the idol room feels much more like a temples grand vault.

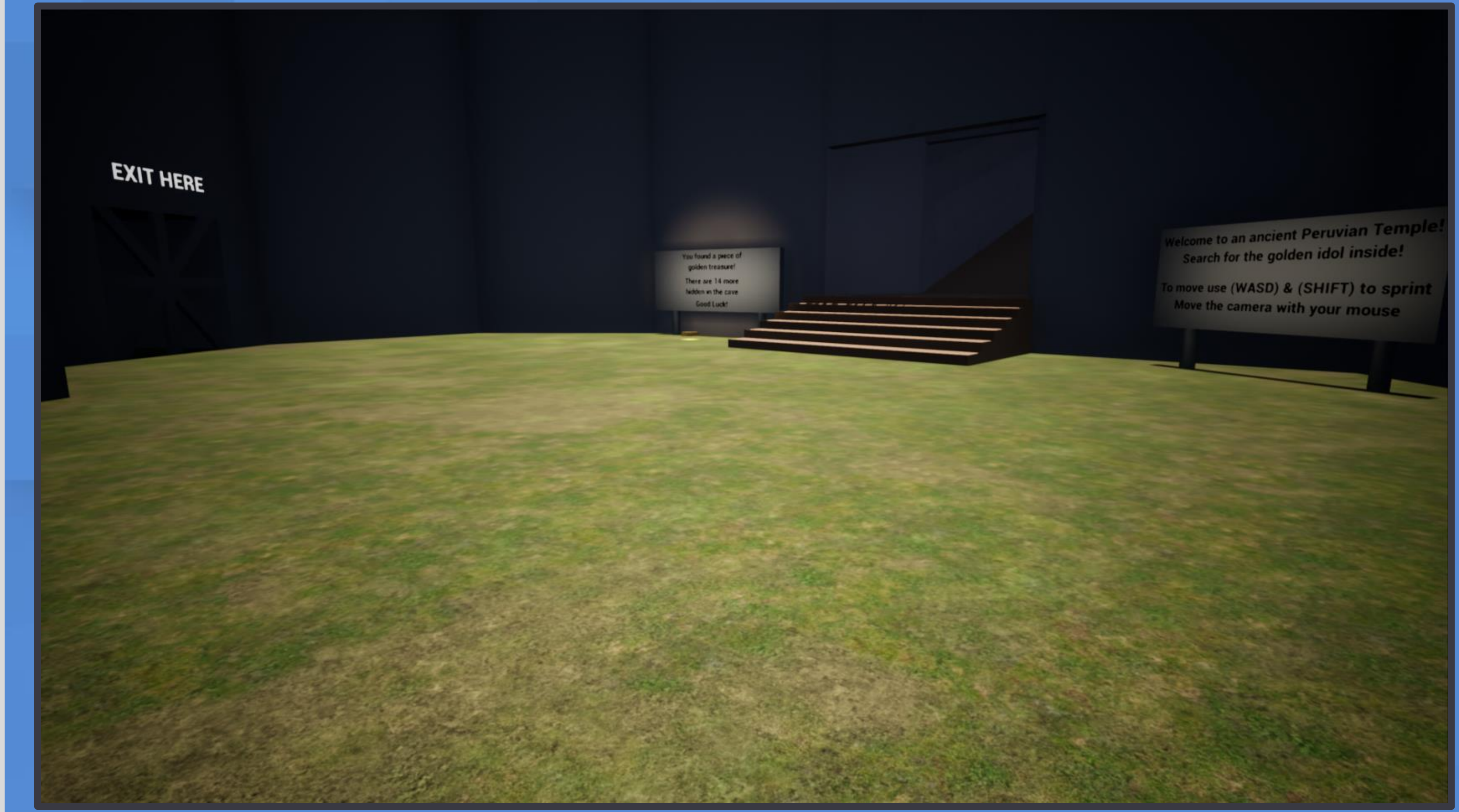
Before Feedback



After Feedback



Final Renders | Temple Entrance


EXIT HERE
You found a piece of golden treasure!
There are 14 more hidden in the cave
Good Luck!
Welcome to an ancient Peruvian Temple!
Search for the golden idol inside!
To move use (WASD) & (SHIFT) to sprint
Move the camera with your mouse

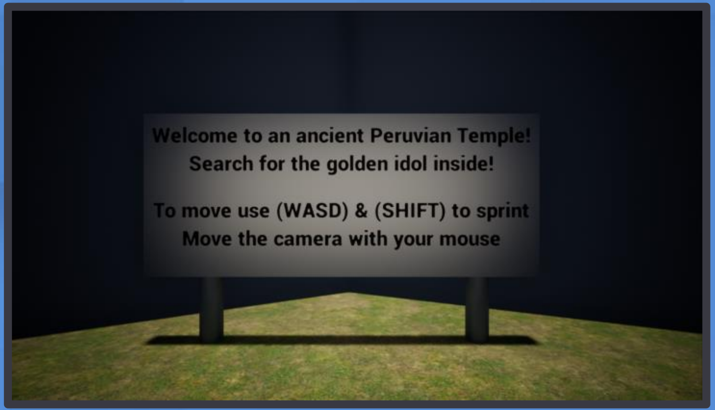
Gold board

You found a piece of golden treasure!
There are 14 more hidden in the cave
Good Luck!

Exit point

Final Renders | Tutorial

Basic Movement

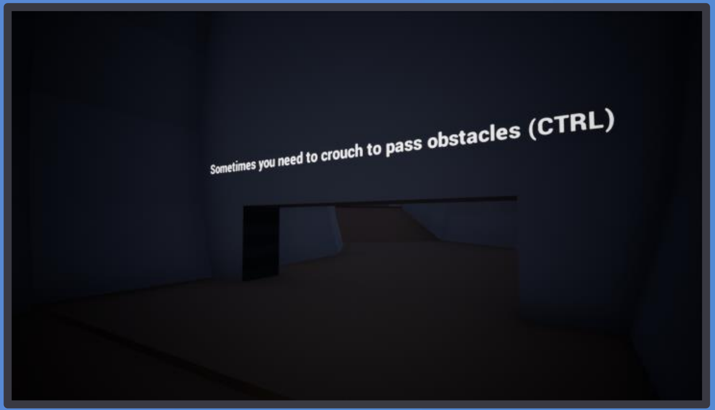


Welcome to an ancient Peruvian Temple!
Search for the golden idol inside!

To move use (WASD) & (SHIFT) to sprint
Move the camera with your mouse

Tutorial Entrance

Crouching Tutorial

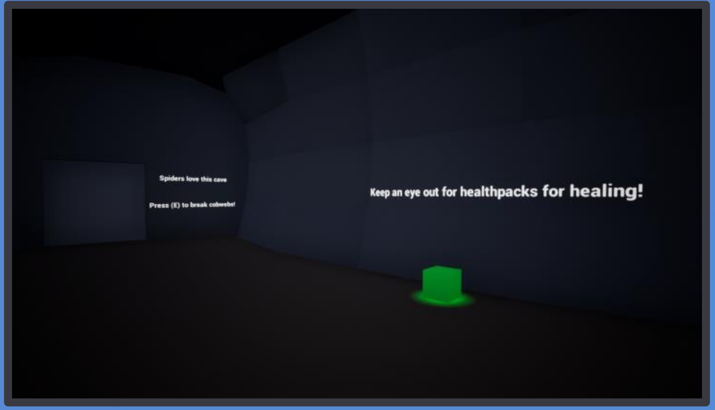


Sometimes you need to crouch to pass obstacles (CTRL)

Jumping Tutorial

Other times you need to jump over (SPACE)

Health pack and cobweb help



Spiders love this cave

Press (E) to break cobwebs!

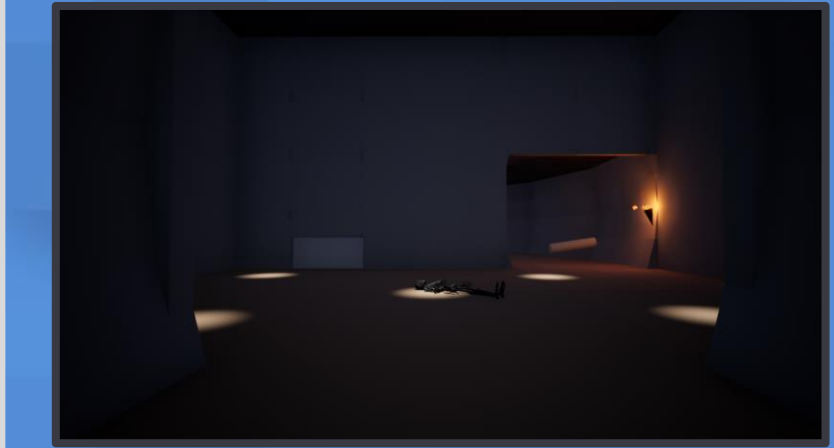
Keep an eye out for healthpacks for healing!

Torch light tip

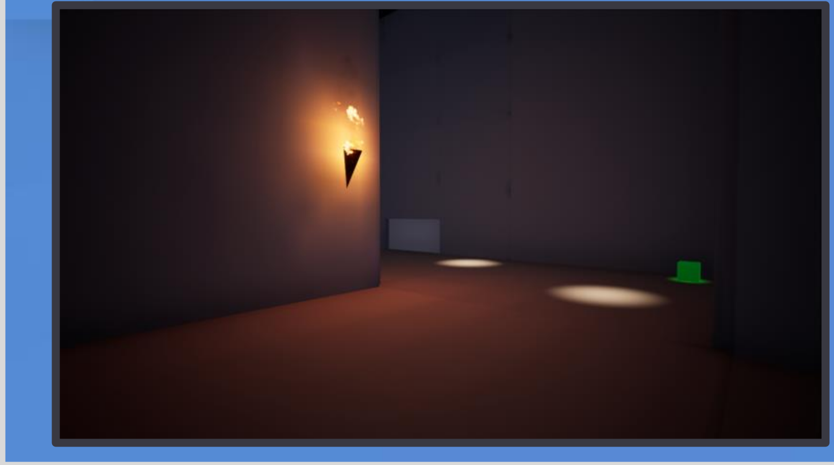
Keep an eye out for torches leading the way!

Final Renders | Light Traps

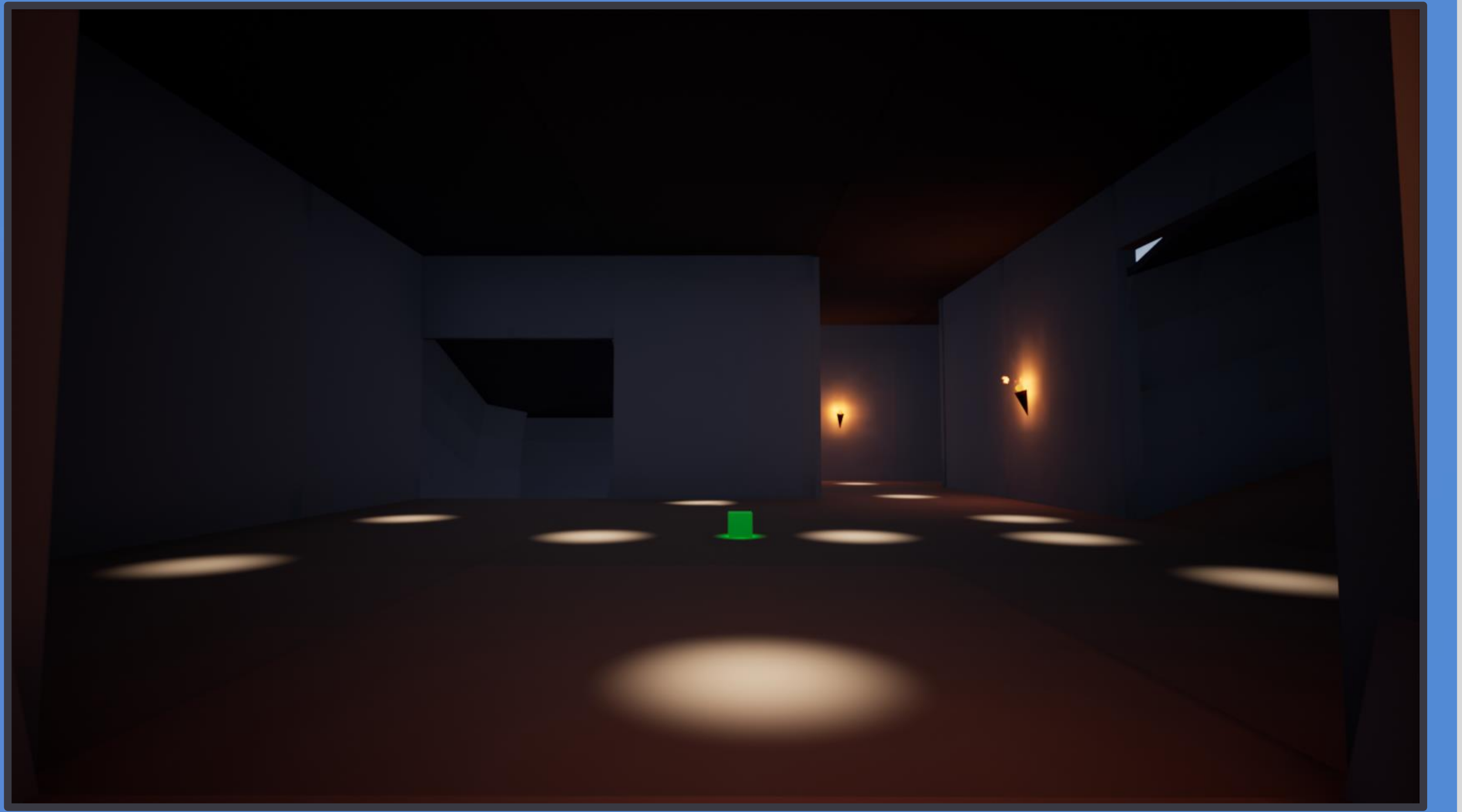
Introduce



Develop

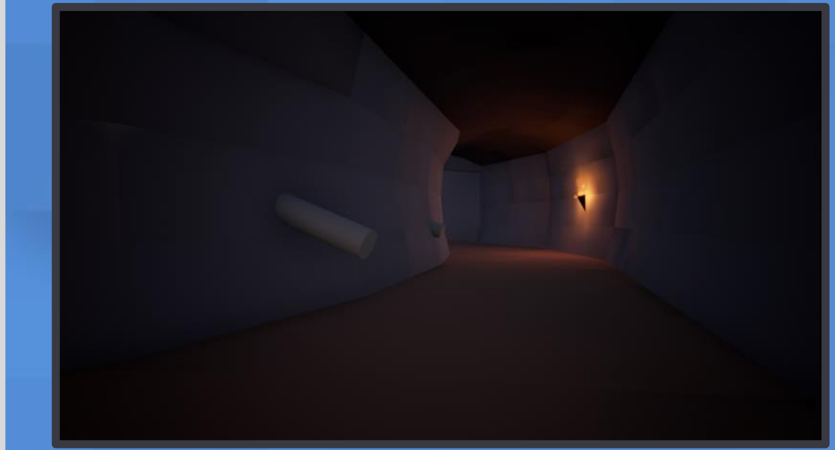


Twist

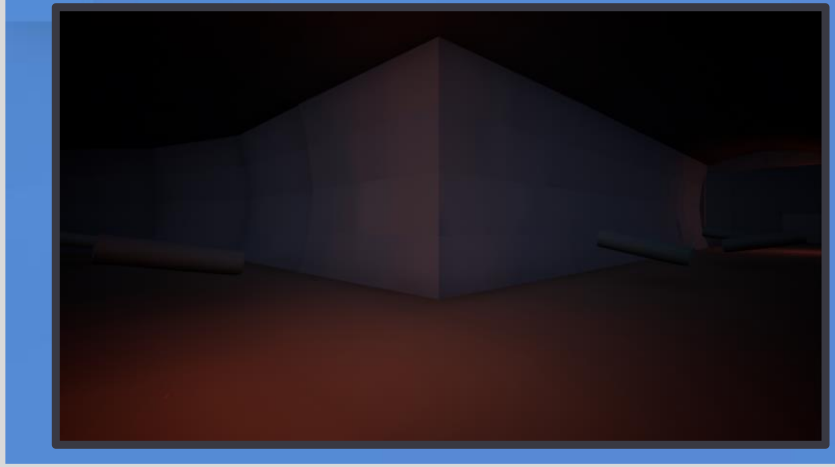


Final Renders | Swinging Logs

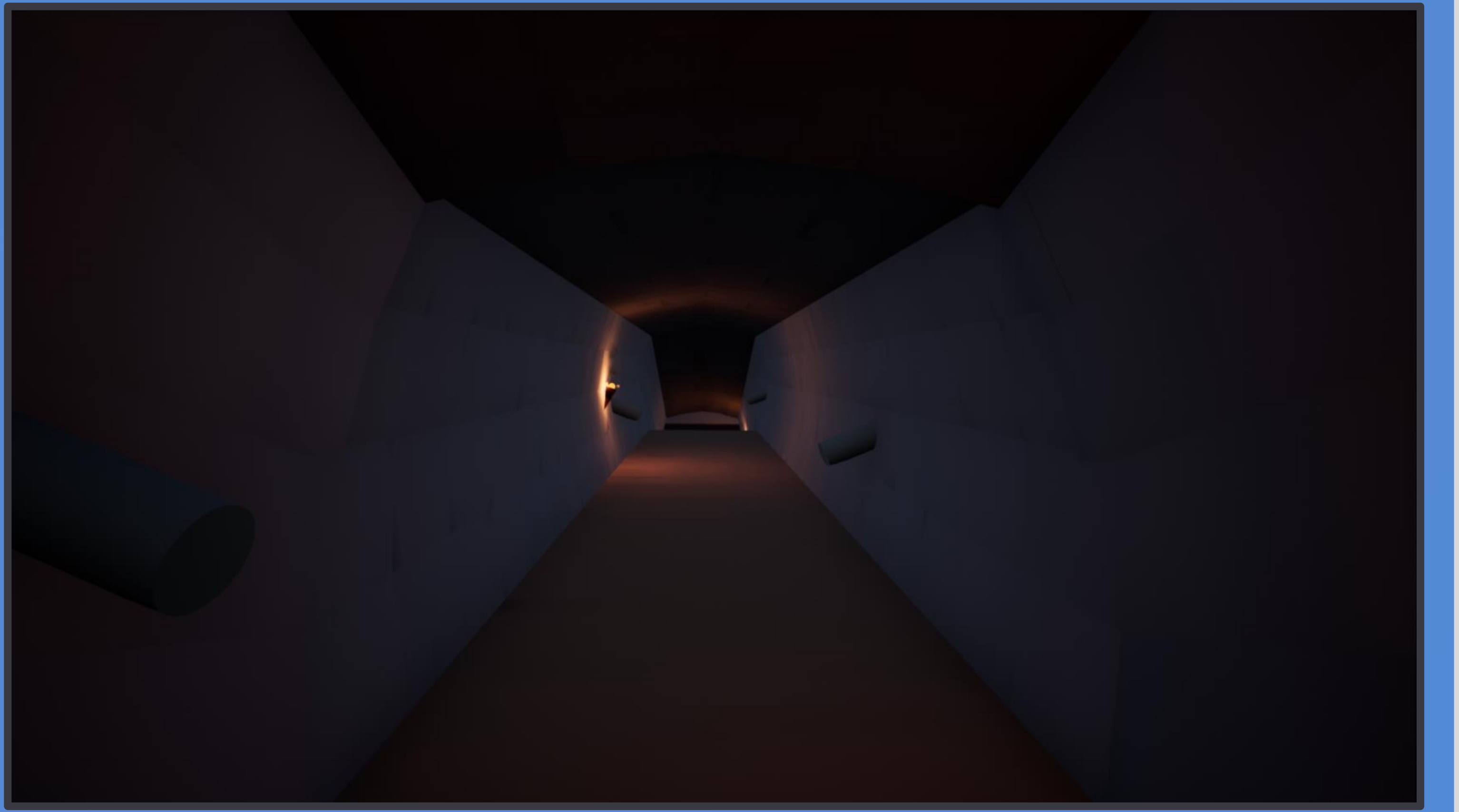
Introduce



Develop

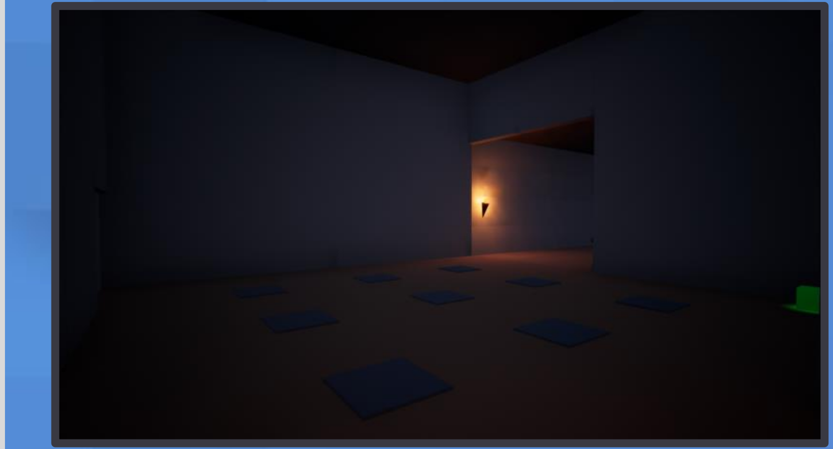


Twist

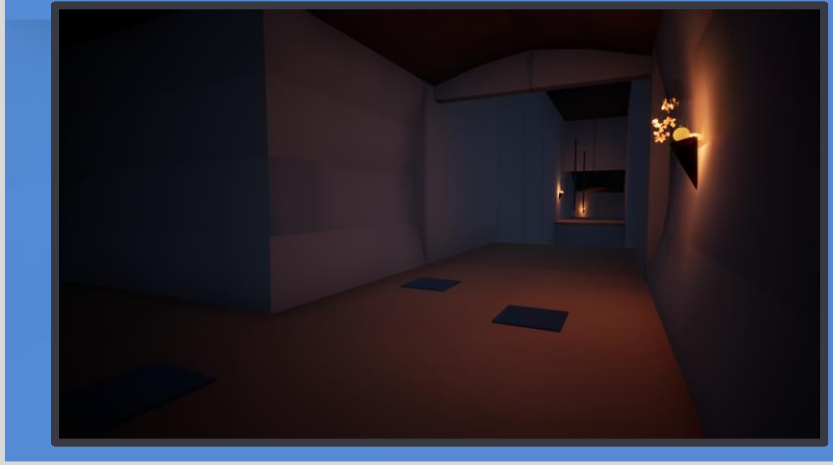


Final Renders | Floor Puzzles

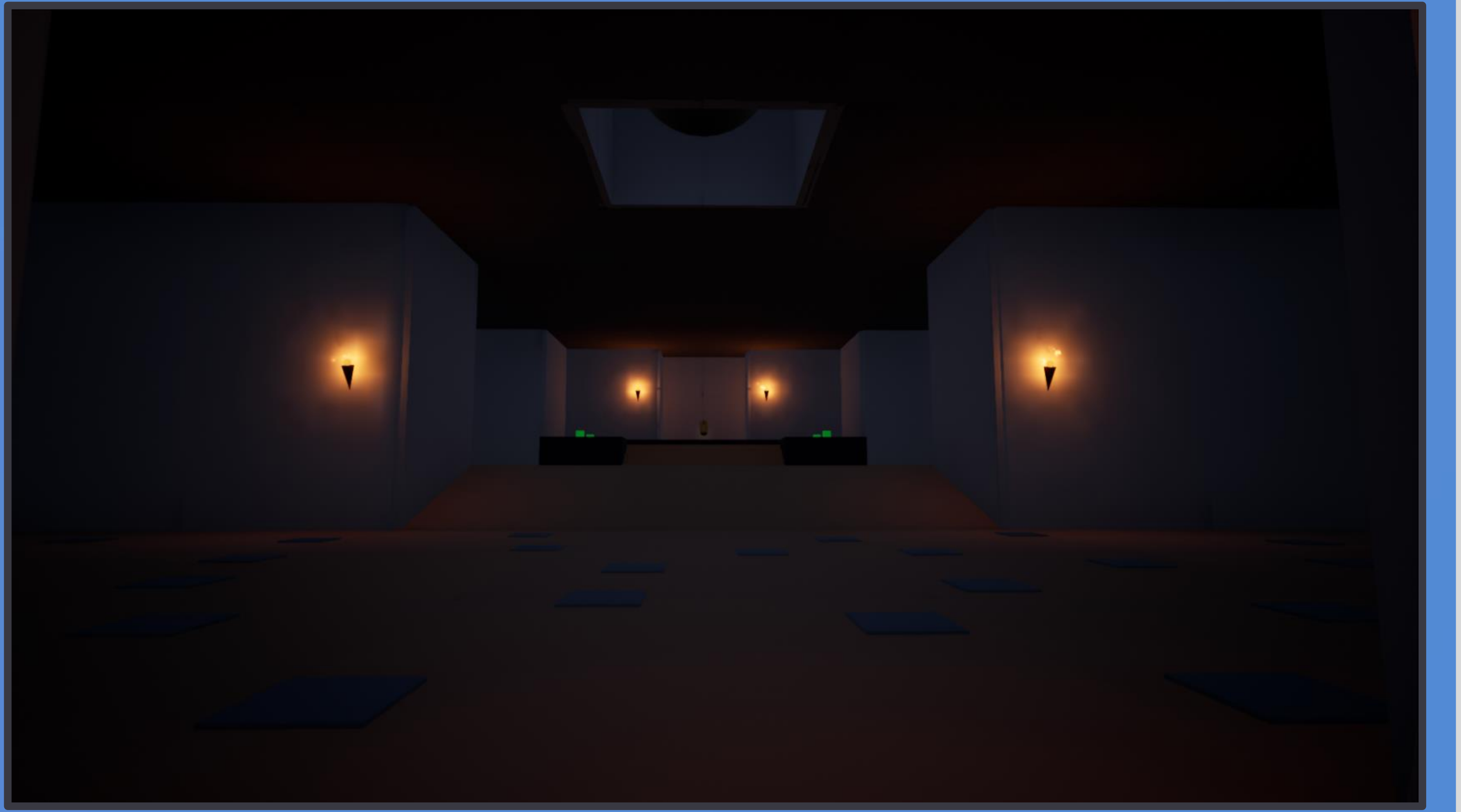
Introduce



Develop



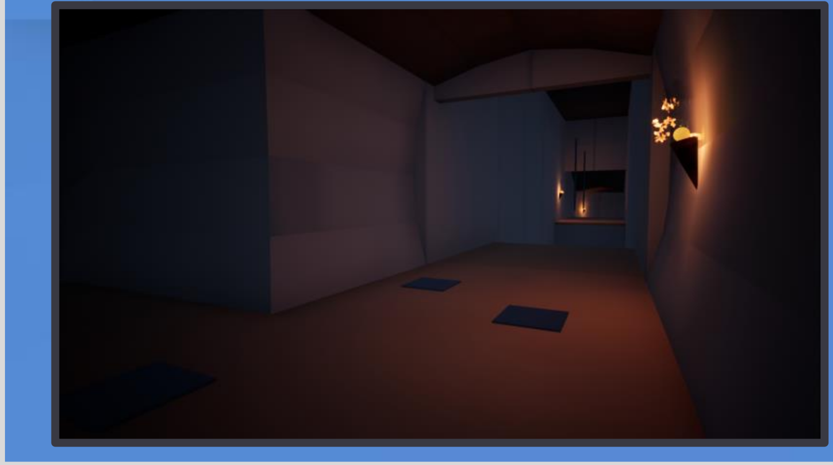
Twist



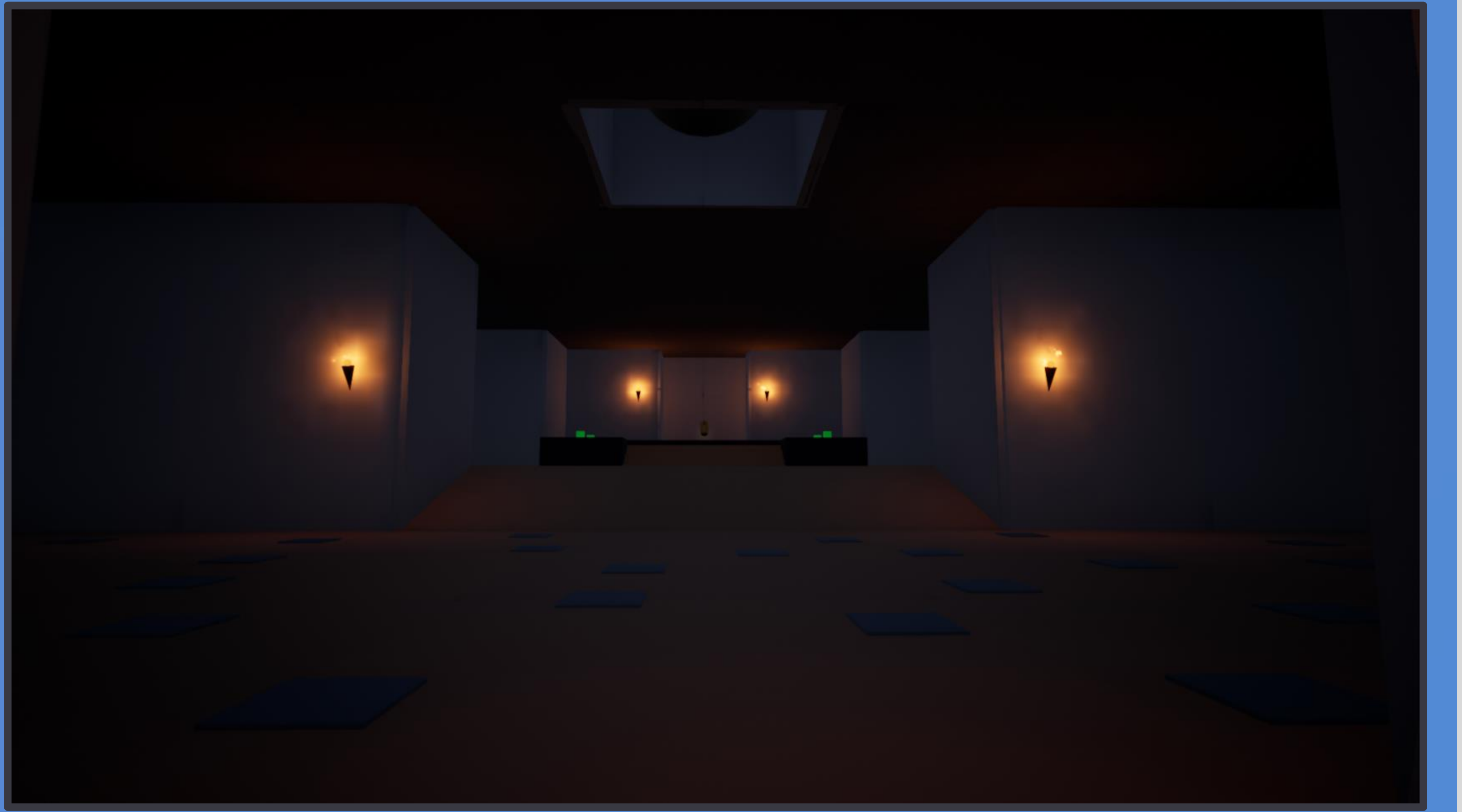
Final Renders | Extra Traps

Introduce

Develop



Twist



Final Renders | Rope Swings

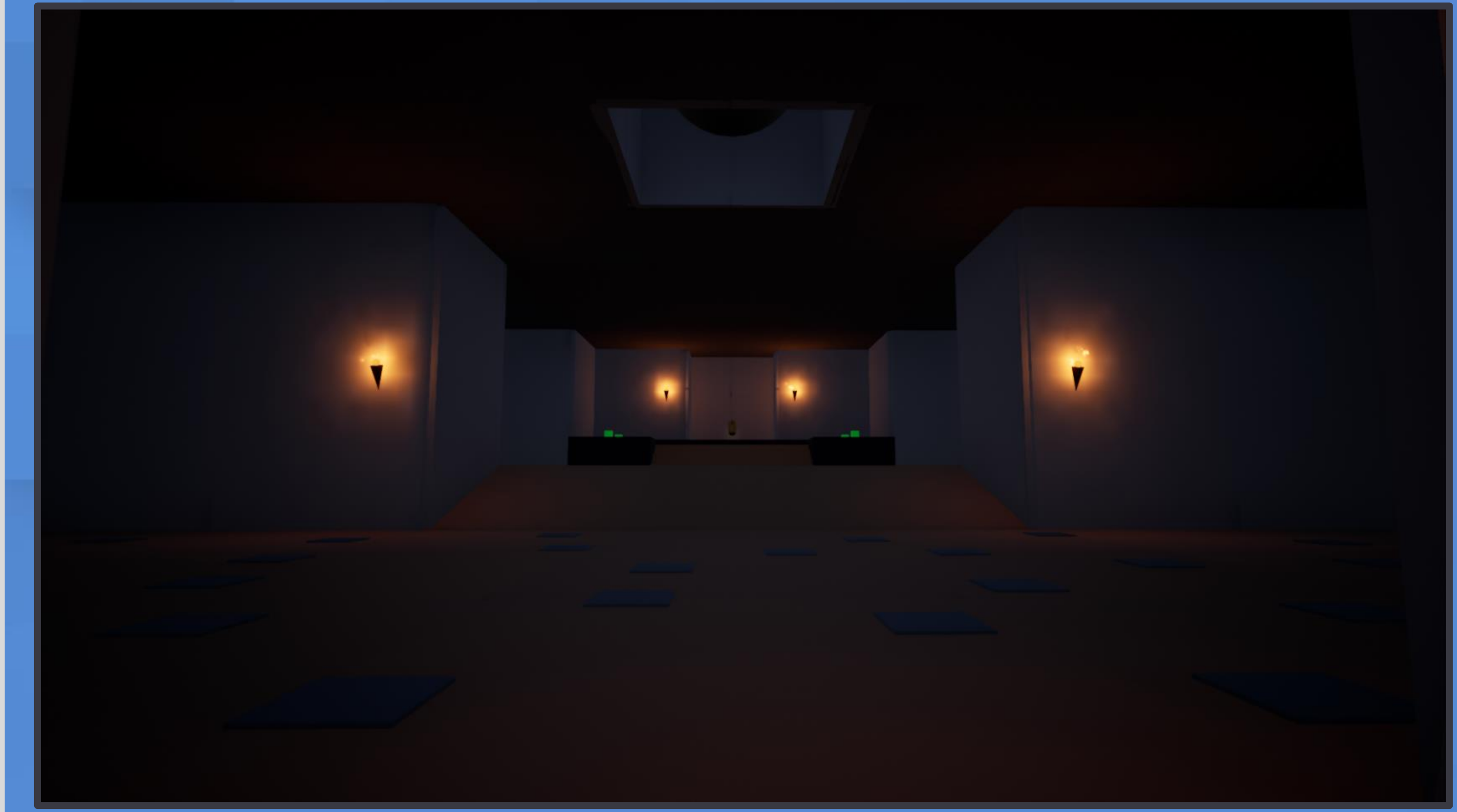
Introduce

After landing, press (R) to set the player character

Develop

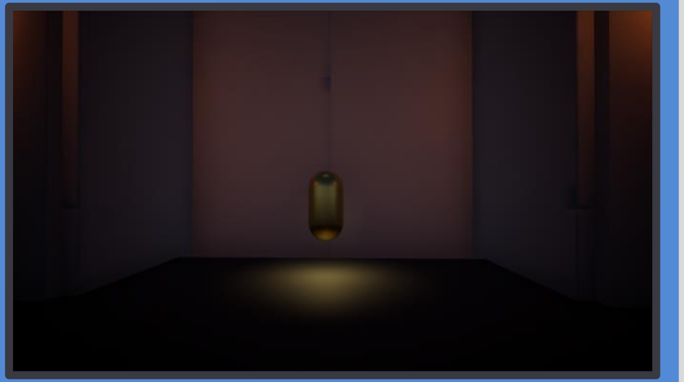
Twist

Final Renders | Idol Room



Idol Pedestal

Idol close up



Feedback (2)

After completing the final version of the game I gathered feedback from some more players to see how the changes have effected gameplay and the level as a whole. Some key points from this feedback were:

Players enjoyed the new layout and appearance of the temple saying it felt more natural and realistic. This was due to the new assets I made in Blender. The torches leading the correct path was helpful for navigating the temple.

While the previous design was too difficult for most players, the new layout has flipped and become a bit too easy. Some players found the old layout challenging but rewarding and enjoyed the difficulty, but had trouble completing the game. This layout has become too easy and these players are getting through the level with minimal difficulty, losing very few lives. Other players however had a better time with the levels lowered difficulty, managing to complete the game when they couldn't before.

I believe the levels difficulty needs to be in the middle ground between the two to provide a challenge but not make it almost impossible for the average player. The gold coin locations while in more challenging areas, were not too difficult to collect that players had to pass up on the opportunity, meaning the majority of players gathered all 15 coins.

The ending boulder was much better than before, was intense and added fear when escaping the temple. Some kept looking back to see if more were coming.

The rope swing had positive feedback with players saying it was much easier and more reliable than before. This may have been because of minor changes I made to the mechanic itself, or the fact I added instructions and overall made the rope swings slightly easier throughout the level.

Evaluation

I believe overall the project went well, I managed to stick to the deadline and complete the project on time. While there were some struggles, for the most part it was a positive experience. One area I believe worked extremely well was deciding to create a mechanics level where I could develop the traps in a clean environment without any level constraints. This allowed me to build up the traps and mechanics I wanted to include in the level, test how they worked alone and in conjunction with other mechanics. Doing this before taking the main level further gave me valuable details about the trap that made me realise that the main level needed redesigning to better accommodate the traps due how they functioned. This was discovered early, which meant I did not waste time on a level that would not work in the end.

My pre production stage allowed me to see how the traps were going to function from the beginning which allowed me to design a level around them early on. While I did have to rework the level after building the traps, the order the traps appeared in didn't need major changes. The four step structure I followed of introduce, develop, twist and conclude worked well to help the player get used to the traps and build up their skills when approaching them.

My initial layout design v1.0 was beneficial to my later level design as it set up the order of the traps early on. This layout was based fully on the film research I had done, so I knew it would not be perfect for a game level but helped me learn many of the key ideas I wanted to bring later into the final game design. This was later changed multiple times and eventually drifted away from the film accurate design however this was for the better as I believe it made a better game level.

The latest level (layout v4.0) needed an increase in difficulty as I lowered the overall challenge of the level compared layout v3.0. This did make the level easier for players to complete, meaning they could experience the full level, however it became too easy and some players missed the challenge. To improve this I would randomise the traps up more, possibly combining them together more often with multiple traps working together to try stop the player. Depending on the changed difficulty, I may lower the max health of the player, or remove some health packs from the level.

Evaluation

Blueprint wise, I believe the majority were created well, I have improved my skills greatly and now feel much more confident at creating mechanics for games. The main issue was the rope swing which was very buggy which I improved by limiting rotation when on the rope, however this did make it less moveable for the player, but the benefits outweighed the negatives for me at this stage. I would love to go back and rework the rope swing from scratch now I know more about blueprinting and unreal engine mechanics, however during the project due to time constraints, I did not have the chance to rework the rope swing to work without bugs.

I believe pre production overall was successful, however if I went back, I would have done more sketches both on paper and digitally. I would have added a layout sketch after developing my trap blueprints which may have made a level that fits the traps better as I had a couple of redesigns in unreal engine which may have been avoided if I had planned them out more.

I also would have liked to make better block out meshes earlier on to make the temple feel more accurate. When I made some quick block out tools in blender, the layout I made with those felt more like a temple than the previous layout to me. In the third layout I used geometry tools in unreal engine which was fine to begin with but began causing performance issues when moving them around as the level grew. This issue was not present when I used the blender meshes.

Another minor area I would have improved on would be adding more audio effects to the game, such as ambient noise and footsteps.

Overall, the project and module went well, I have learned a lot of valuable skills in how to approach level design. My skills in pre production and designing layouts have improved and I have developed my unreal engine skills, now feeling confident in creating game mechanics, animations, materials and other key game features.

References

Research

- *Raiders of the Lost Ark*. 1981. [film] Directed by S. Spielberg. United States: Paramount Pictures.
- Game Makers Toolkit, 2015. *Super Mario 3D World's 4 Step Level Design | Platformer Level Design*. [online] Youtube.com. Available at: <https://www.youtube.com/watch?v=dBmIkEvEBtA> [Accessed 9 January 2021].
- Game Makers Toolkit, 2016. *How Games Do Health | Game Maker's Toolkit*. [online] Youtube.com. Available at: <https://www.youtube.com/watch?v=4AEKbBF3URE> [Accessed 9 January 2021].
- Game Makers Toolkit, 2018. *How To Keep Players Engaged (Without Being Evil) | Game Maker's Toolkit*. [online] Youtube.com. Available at: <https://www.youtube.com/watch?v=hbzGO_Qonu0> [Accessed 9 January 2021].
- Flusher, 2019. *USE OF COLORS IN GAME DESIGN AND LEVEL DESIGN*. [online] Youtube.com. Available at: <https://www.youtube.com/watch?v=d-hwSG4o8xg> [Accessed 9 January 2021].
- Game Developer Conference, 2018. *Level Design Workshop: Blockmesh And Lighting Tips*. [online] Youtube.com. Available at: <https://www.youtube.com/watch?v=09r1B9cVEQY> [Accessed 9 January 2021].

References

Assets and Audio

- pngkey.com. 2021. *Download Red Vignette Background - Peach PNG Image With No Backgroud - Pngkey.Com*. [online] Available at: <https://www.pngkey.com/maxpic/u2y3q8q8r5u2w7r5/> [Accessed 9 January 2021].
- Free3d.com. 2021. *Skeleton Free 3D Model - .3Ds .Obj .Sldprt - Free3d*. [online] Available at: <https://free3d.com/3d-model/skeleton-81335.html> [Accessed 9 January 2021].
- Freesound. 2021. *Rumbling Rocky Sound By Opsaaaaa*. [online] Available at: <https://freesound.org/people/Opsaaaaa/sounds/335337/> [Accessed 9 January 2021].
- Freesound. 2021. *Arrow Impact By Twisted_Euphoria*. [online] Available at: <https://freesound.org/people/Twisted_Euphoria/sounds/205938/> [Accessed 9 January 2021].
- Film Concerts Live!. 2021. *Raiders Of The Lost Ark In Concert – Film Concerts Live!*. [online] Available at: <https://filmconcertslive.com/movies/raiders-of-the-lost-ark/> [Accessed 9 January 2021].
- Unreal Engine. 2021. *Animation Starter Pack In Epic Content - UE Marketplace*. [online] Available at: <https://www.unrealengine.com/marketplace/en-US/product/animation-starter-pack> [Accessed 9 January 2021].

Portfolio Links

Level Walkthrough: https://youtu.be/-lKzjXON67g
Trap Showcase: https://youtu.be/tujCRMxqpSA
Game Download: https://drive.google.com/file/d/1rxfeRWdgNJce0foxVXnwdHgBw1Ku3wAT/view