Puzzle Tavern

Estimated Project Plan

Week 1 – Pre-production + initial Design

Week 2 – Continued Design and Production + Puzzle Design

Week 3 – In-engine mechanic implementation + Puzzle blockout

Week 4 – Puzzle Implementation + Playtesting Feedback + implementation

Week 5 – Project polishing + Showcase Video creation

Contents

Week 1

Weekly goals:
- Establish the concept outline
- Establish the games design pillars and goals
- Talk through and diagram the gameplay loop as well as key mechanics.
- Establish win and loss conditions of the game as well as what features from each genre I wish to merge into this prototype.

Conception

My two concepts and reasons for choosing them

So, my two concepts I am combing are rogue-lite and puzzle. In terms of pacing and design these two genres of games are so wildly different I thought combing them could make for some really interesting design decisions, with rogue-lites generally being quite fast paced were puzzle games are generally the opposite, so merging them could lead to some interesting gameplay. As well as this I think it would really allow me to stretch my creative thinking to make the best design decisions possible.

Elevator Pitch

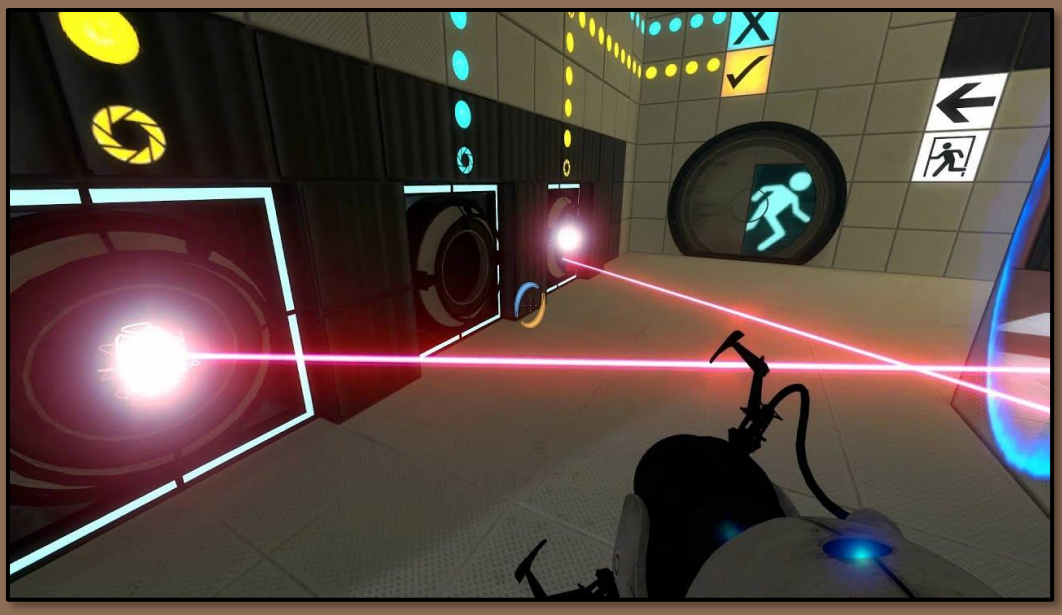
Puzzle Tavern is a 3D rogue-lite puzzle game, the player will have to venture through a series of randomized puzzles in order to get to the end of each 'run' to find the wizards crystals.

Game outline

As Puzzle Tavern is a Rogue-lite puzzle game, I first had to consider what elements of each genre I wanted to incorporate into the gameplay. The key element here for rogue-lite is the randomization on the puzzle as this will directly impact the core gameplay loop. However, as most rogue-lite systems are background systems, these will impact more of the progression of the game as opposed to the initial gameplay loop, other than randomization.

Ideation

Before I decide to send it entirely one idea, I wanted to look at a few different games that had features I think would fit/mesh well together. So, my initial research had me looking at puzzles games, and my first ideas were to look at mechanics from games such as Portal or Portal 2 as almost the 'go-to' puzzle game in gaming. While puzzle had quite a few mechanics I could take inspiration for, one that



particularly stuck out to me was the lasers that portal use to solve several puzzles throughout the game.

While portal did have the portal mechanic to make these more interesting, I still felt like this was something I could work with during my conception period and make some interesting puzzles with throughout my game.

In terms of my rogue-lite inspirations I looked at the game Hades. While Hades is a combat game during its core gameplay loop it had a lot of features I enjoyed and would like to take from, mainly tied into the progression of the game.

Throughout each run of the game the player will pick up 'Darkness', a currency that can be used to buy upgrades that will benefit the player during their run. I quite liked this idea and was certain I could apply it to a puzzle game instead of a combat scenario.

Executive Summary

Puzzle Tavern is a fast-paced puzzle rogue-lite game where the player has been tasked by the wizard Razmuth to break back into his tavern and beat his security system to retrieve his broken crystal.

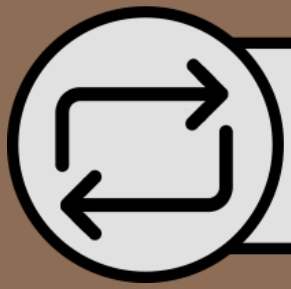
The player will have to engage and interact with a complex series of rooms and guide the laser using redirection cubes and reflect walls to open the door to the next room. The player shan't fear failure as they will find crystal fragments in each room to buy upgrades, to improve their chances on their next attempt of getting through to the end of his tavern.

By thinking quick on their feet, the player will be able to swiftly navigate this maze of rooms and reach the end before the security system notices their presence and kicks them out for good.

Design Pillars

Quick Thinking
The game should encourage the player to think quickly. To solve puzzles with speed, so they can progress faster and avoid reaching a loss state. However, some systems will be implemented to help circumvent loss due to this.

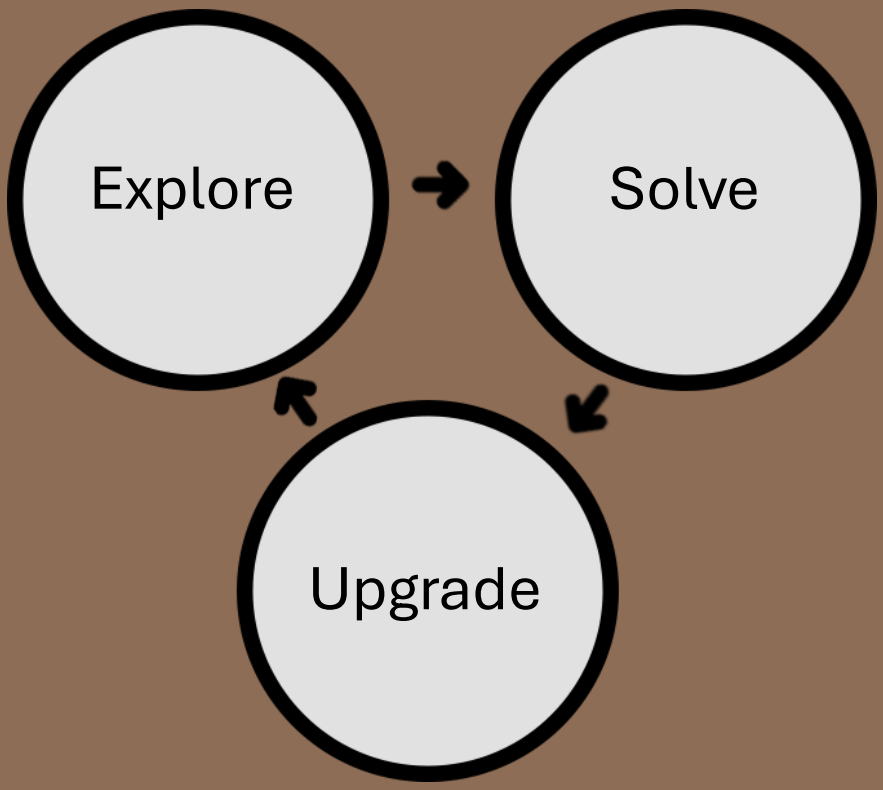
Magic
The game should give the player a feeling they are in some sort of fantasy realm, somewhere that science perhaps does not exist. Most of the mechanics here should be powered or involve magic and even magical events should likely happen around the player.



Repetition
As the game is a rogue-lite, it should encourage the player to do multiple runs of the game. The mechanics should help encourage this by letting the player do new things or allow them to progress somehow between runs.

The main reason I have decided to go with these design pillars is to keep me on track while developing mechanics for the game, and to make sure all the mechanics positively contribute to gameplay in a reliable and consistent way.

Core gameplay loop



Explore

Throughout the gameplay, the player will be given a variety of different rooms and puzzles they can explore, hopefully leading to seeing something new every time the player does a run during the game.

Solve

The player will be required to solve puzzles throughout the game, consisting most of the core gameplay loop. Not successfully being able to solve a puzzle will eventually lead to the player failing and resetting the gameplay loop. Slightly involved with explore, but the player will have to solve new puzzles every time they come across one meaning they will need to apply learnt skills throughout the game in different ways.

Upgrade

After they have solved puzzles, the player will be able to acquire upgrades, these upgrades will then help them throughout the rest of the gameplay by providing them with buffs that will persist throughout their game play.

Win and fail states

There are two main win states throughout the game, however there is only really one main fail state.

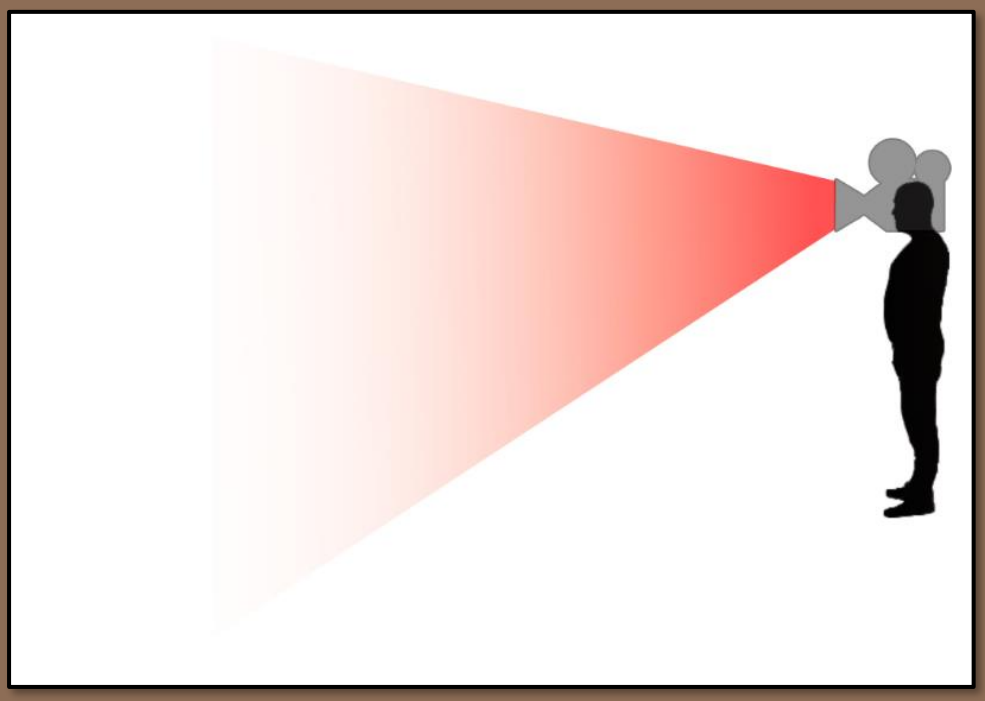
The two main win states in the game, will consist of the player solving each puzzle room and therefore delaying their progress to a fail state, and the second win state is the completion of a run in the game. However, as the game's primary focus on the game is playing through all of the puzzles and acquiring upgrades, so there is not a big emphasis placed on losing throughout the game.

The main and only lose state here is the player not successfully solving a puzzle causing them to fail their run.

Mechanics

Player Camera

The camera will be placed in a first-person view, and the player will be incapable of seeing their own body. As well as this the camera will be controlled using the mouse and will allow the player to look around in 360 degrees left or right, as well as 180 degrees up or down.

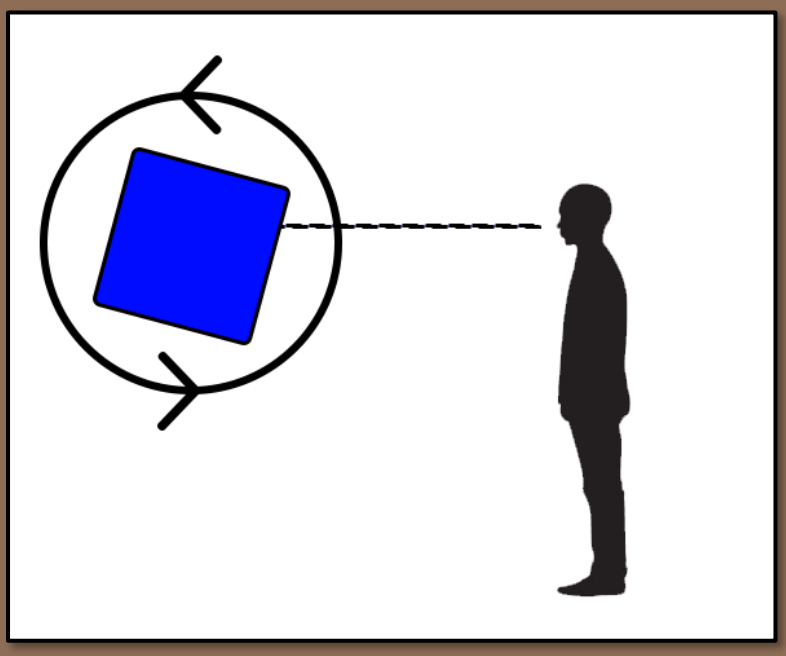


Player movement and interaction

The movement in this game will emulate what is expected of real life, physics, and movement.
The player will be able to move around and explore the environment using **WASD**. As well as this they

will be able to jump up or down using **Space**. Movement in this game will not have a big emphasis as they will not contribute to the core gameplay loop, other than navigation.

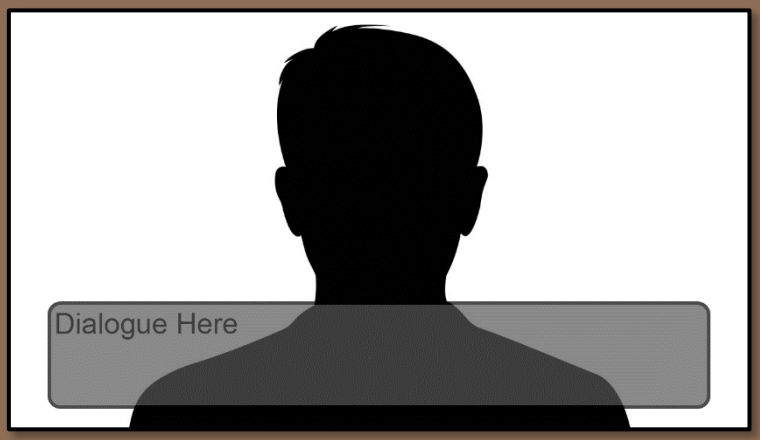
As well as movement, the player will be able to interact with various things around them in the environment. In terms of picking up objects the if the player press the **E** key which will be the button to



interact with most things, the object will position itself in front of the player and stay there until the player presses **E** again to drop it.

As well as this the player will be able to also interact with doors, that will open 90 degrees. As well as interact with certain objects to open dialogue boxes.

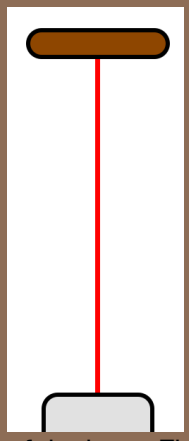
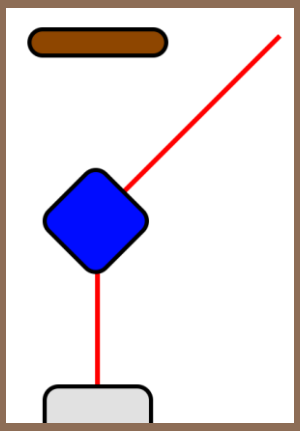
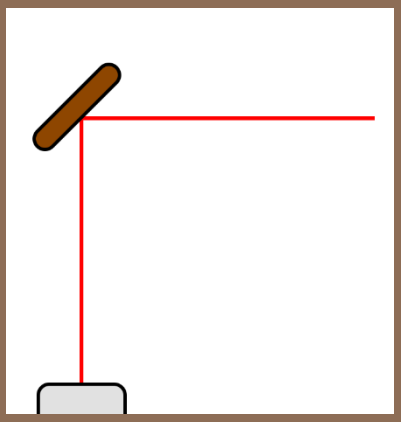
To continue onto the dialogue boxes, when you interact with an object that provides dialogue, the player will cease all movement and be unable to move until they have finished dialogue with whatever is causing it. Here you can see also that the dialogue box will spawn at the bottom middle of the screen and take up most of its width.



Puzzle Mechanics

Throughout the game the player will be able to interact with different kinds of puzzle mechanics. The most prevalent puzzle mechanics will revolve around the laser being used to activate and solve the puzzle, as well as interacting with other elements that will change the direction of the laser.

In terms of just the laser, it will shoot forward out of a block and go straight forward until it hits a solid object. However, as stated earlier there will also be a block and a wall that change affect the direction

of the laser. The block will be an object that the player can pick, move, and rotate and this will help the player solve puzzles by manipulation the direction of the puzzle.

As well as this block there will also be a 'reflect wall'. This wall will not be interactable directly however, the player will be able to press a button to rotate it 45 degrees, to cause the laser to reflect off of the wall and hit somewhere according to the direction of the 'reflect wall'.

There will also be a 'target block' This block when hit will solve the puzzle and allow the player to progress. However, there will be a second variation of the target block which will be able to change the environment once, to increase the variety of the puzzles.

A secondary mechanic that will be used throughout the puzzles, is a pressure plate. The main function of the pressure plate will be when it is pushed down it will cause some aspect of the environment or puzzle to move, rotate or change locations while the plate has something on top of it to keep it pressed down.

Week 1 Reflection

With the first week gone by, I am already quite excited about developing this project further. I think I have started with a solid base and am ready to progress onto developing my ideas and creating a fun prototype to play. I feel I have decided a nice set of genres to combine for my prototype as neither me or my peers could think of an example of a game that fit this combination and because of this I feel like it could lead to some interesting interactions and gameplay. I feel like my combination really opens up the genre of rogue-lites to an audience of people who would not usually play them by changing combat out, which is what usually is encompassed in many systems in a rogue-lite game.

In the next section I will go into more detail about what elements I am taking from roguelikes and how I will make them fit into my game. As well as this I will be developing my puzzles that I will eventually implement into my game.

Week 2

Weekly Goals:
- Further develop rogue-lite mechanics
- Document and Develop puzzles
- Develop additional support mechanics.

Rogue-lite

Randomization

So one big important feature that is very prevalent in a rogue-lite game is the randomization of the gameplay. While it would be difficult and potentially lead to some badly designed puzzles I decided not to go the route of the puzzles themselves being randomly generated, but more the actual order of the puzzles being generated. As well as this there will be more puzzles in the game than can be completed in one run so the player will always have something new to go back to.

As well as this, because I do not want the player getting bored from repeating the puzzles, I decided it would be a good idea to make it so the puzzles cannot be repeated in a single run, allowing the player to always experience something new on each run.

Fast paced progression

One key staple of a rogue-lite is that they are meant to be quite fast paced. However, this does counteract my additional combination of a puzzle genre, I feel there is a good compromise to allow the player to problem solve in a fast-paced manner.

I believe the implementation of a timer during each run, that will count down and cause the player to lose if it hits 0. As well as this, when a puzzle is solved the player will receive a 30 second increase in the timer to give themselves, simultaneously a reward and increase their chances of finishing the run.

Meta-progression

Randomization being the main rogue-lite feature is great however, it is very much a background system and does not add much to the apparent gameplay, so I have decided to introduce meta-progression into the game, via a store that the player is able to access after they have completed at least one puzzle, this will allow the player to feel like they are still progressing on runs that they do not get to the end of.

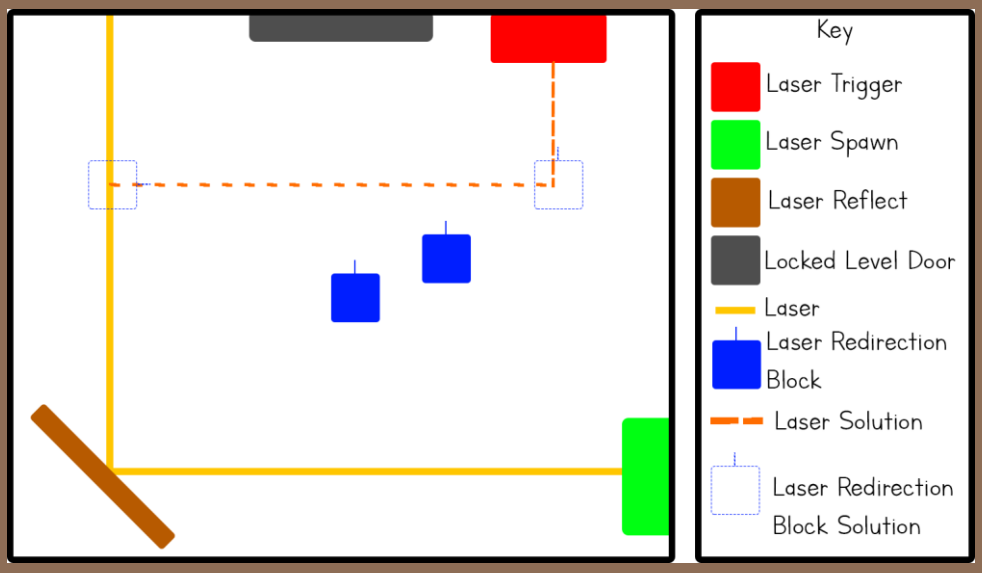
My current ideas for upgrades that could be purchased in this ship will give the player the ability to drastically slow down time for a short amount of time and every subsequent purchase of this will increase the amount of time the player has slowed time down for. As well as this, one of the upgrades will allow the player to increase the amount of time they have when they start their run.

One thing to note is that neither of these upgrades will be free to purchase. When a player beats a puzzle, they will gain 1 crystal and if a player finishes a run they will gain 5 crystals. With these crystals the player will be able to purchase these upgrades.

Crystals	5	10	15	20	25
Slow Time	Slow time unlock	+0.5 seconds	+0.5 seconds	+0.5 seconds	+0.5 seconds
Increase Timer	+5 seconds	+ 10 seconds	+15 seconds	+ 20 seconds	+30 seconds

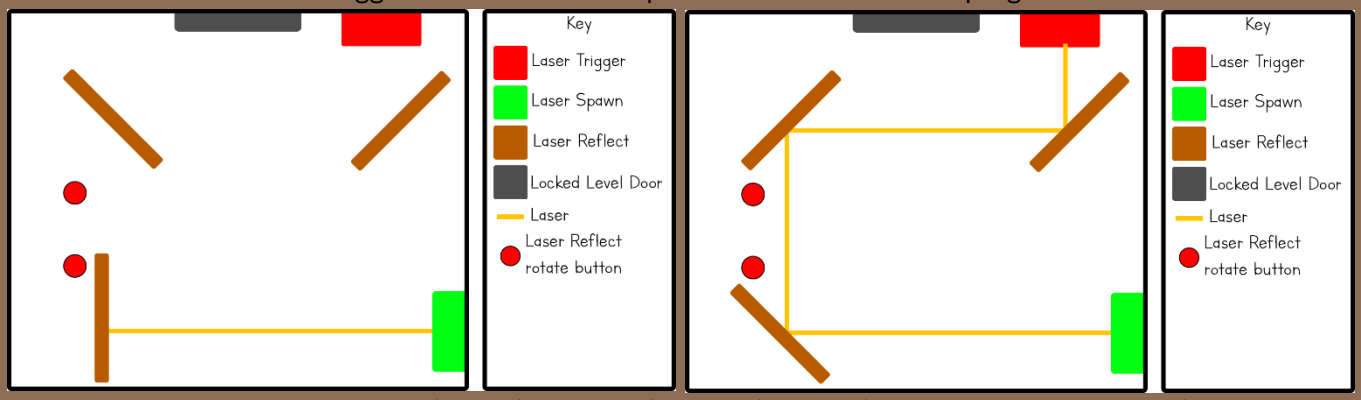
Puzzle Designs

While I am going to have an intended solution to these puzzles, I am aiming to allow the player to experiment and if any solutions are found that I did not intend, that is fine in scope with the design.



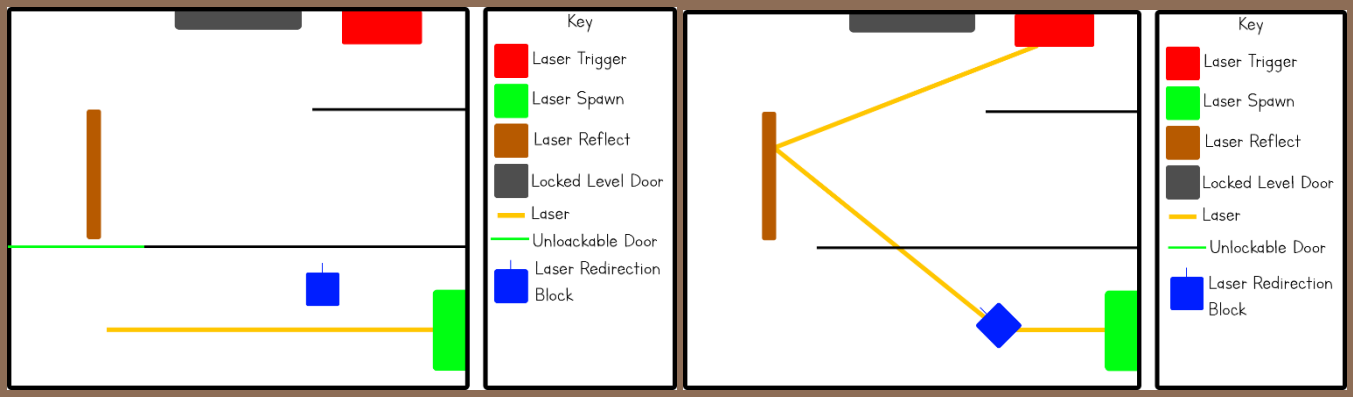
This is the first of many puzzles I am intending on having the player solve during the gameplay. All of the core mechanics are introduced in this puzzle but only have the player use them in a very simple way.

In order to solve this puzzle, the player must use the laser re-direction block to redirect the laser into the trigger. This will solve the puzzle and allow them to progress to the next one.

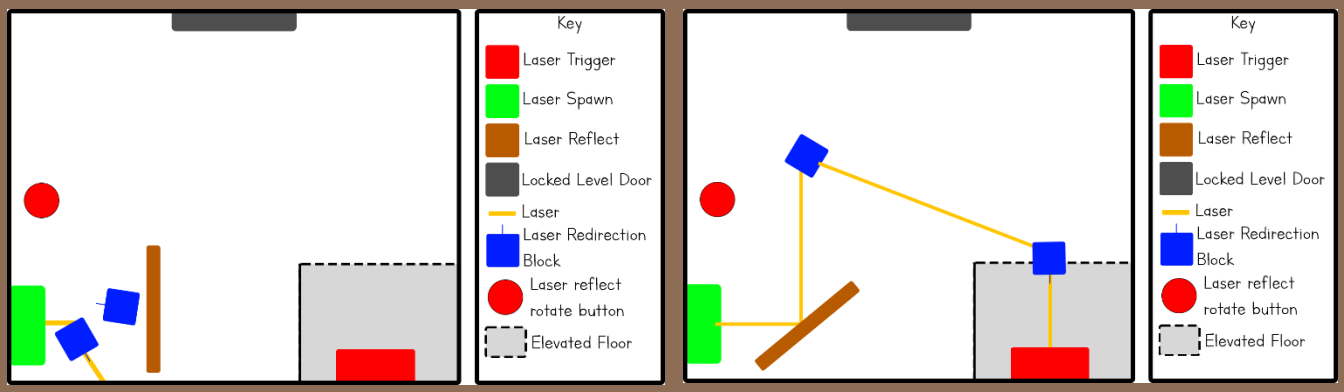


From here onwards, the puzzles will be displayed with two diagrams in order to show the solution more obviously.

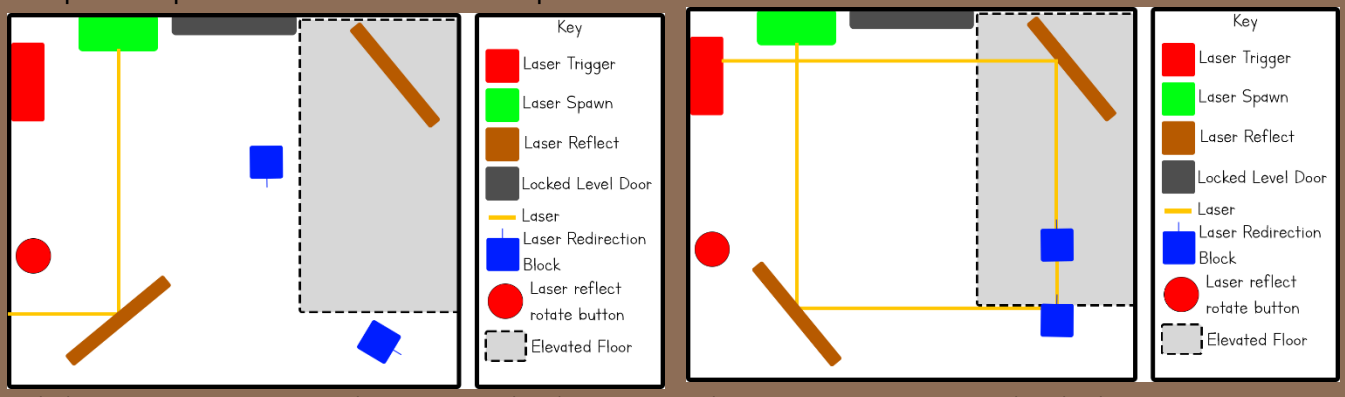
This second puzzle is also going to be on the easier side to solve and will have the player rotate the laser reflect walls into the trigger. The main aim of this puzzle is to teach the player exactly how the laser reflect walls work.



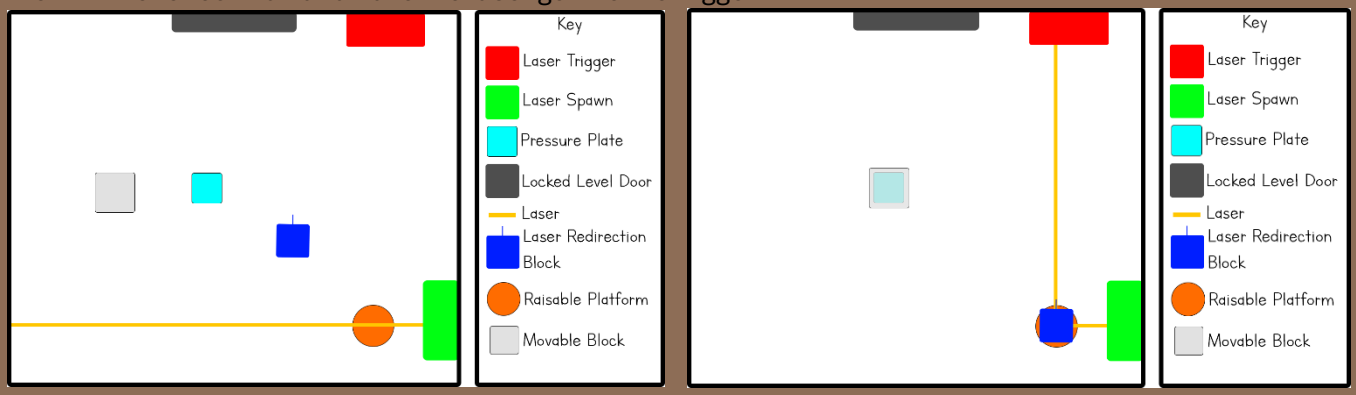
This next puzzle is intended to show the player that they can also rotate the blocks to aim the laser into the trigger, as they are blocked off from moving through the room until the puzzle is solved. Here they need to reflect the redirection block at the wall in order for it to reflect into the trigger.



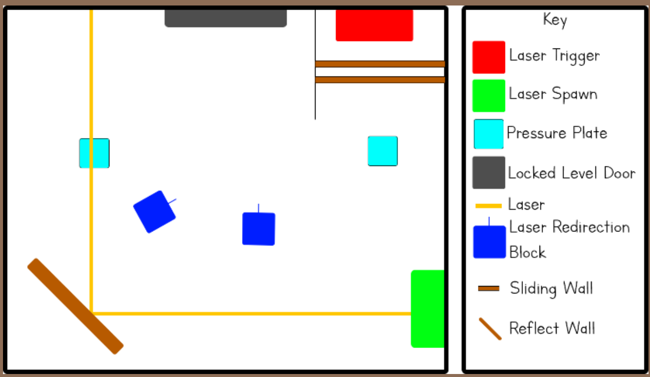
This puzzle introduces some verticality into the room in order to get the player to use the redirection blocks in ways that they have not been able to in other puzzles. Here they have to place a redirection block on the stairs that lead up to the elevated floor to angle the laser, so it hits the trigger. The main goal of this puzzle is to introduce verticality into the puzzles however, I think this also serves as a more complicated puzzle as the solution is not quite as obvious.

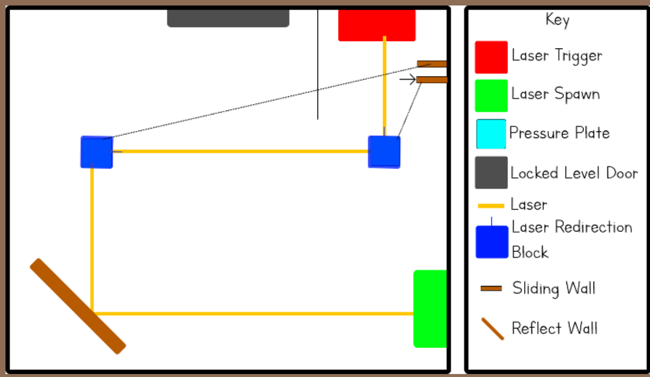


This is another puzzles that involves verticality, however it has more elements to it this time. The player here is required to use the redirection blocks to guide the laser up to the elevated path, which they can then hit the reflect wall and have the laser go into the trigger.

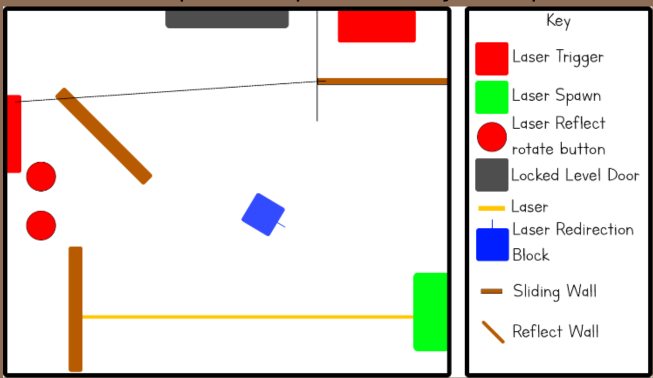


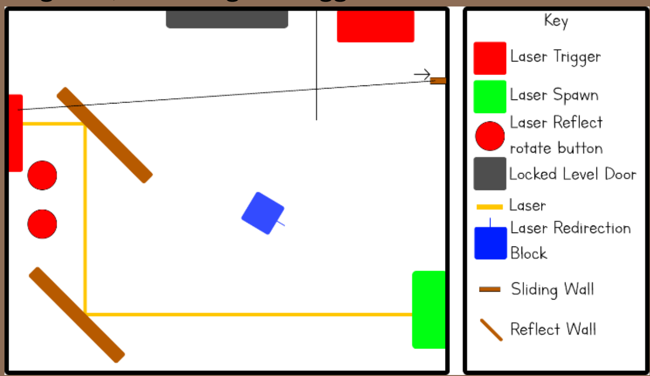
The sixth puzzle is the first puzzle to introduce the use of a pressure plate. This pressure plate when pressed down either by the player or a block, will life up the reusable platform. The player in this instance will need to move the moveable block onto the pressure plate and then redirect the laser into the trigger now that the platform is at an appropriate height.

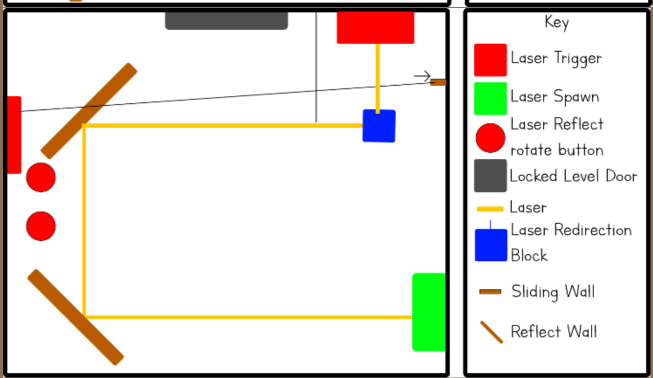




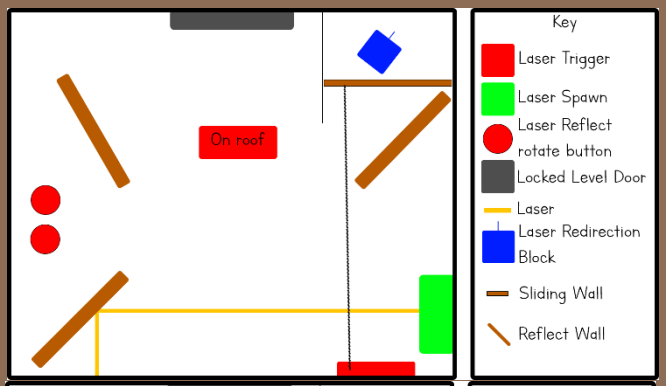
The seventh puzzle is where the puzzle will start to get more complicated and require more time to complete. This puzzle uses all elements of the game that have been previously introduced in other puzzles. The player here must simultaneously redirect the laser to the laser trigger whilst placing the blocks on the pressure plate so they can open the sliding wall, revealing the trigger.

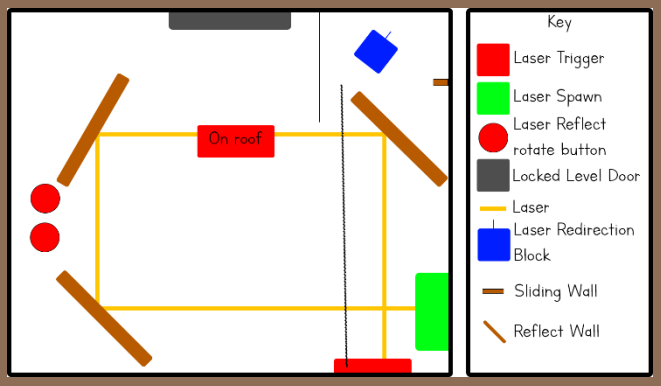


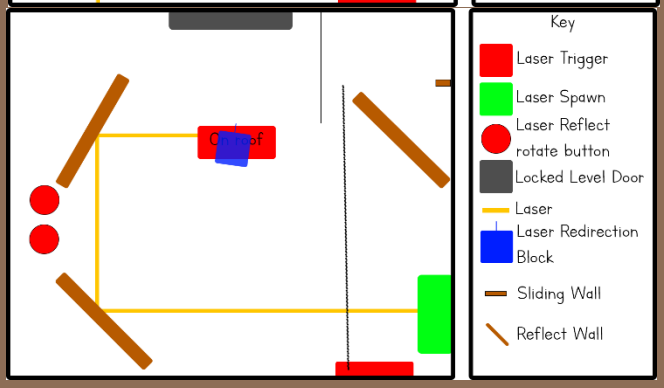




As the next two puzzles are quite complicated, they require more than two diagrams to demonstrate the solution. Here the player must use the reflect wall to hit the first trigger made present in the level. This trigger will permanently open the sliding wall so then the player can use the redirection block in tandem with the reflect wall to direct the laser into the second trigger, in order to solve the puzzle.







In order to solve this puzzle the player must first use the reflect walls to hit the trigger on the wall. This will open up the sliding wall revealing a redirection cube. Once this has been done the player can then use the redirection cube to direct the laser onto the trigger that is on the roof the room to solve the puzzle.

Week 2 reflection

So during this week I got a lot of work done, mainly in the form of documenting and designing the puzzles that will be present throughout the game but also in further designing the core mechanics of the game, specifically relating the rogue-lite elements I will be incorporating. I believe I have taken a good number of elements from both rogue-lite and puzzles game and the design is really coming together. I did have a lot more trouble than expected while designing the puzzles as it took a lot of time to make puzzles, I was happy with.

In the next section I am going to discuss the art and UI design as well as my development in engine.

Week 3

Weekly Goals:
- Design the UI and HUD for the game.
- Mechanic Implementation
- Asset Implementation

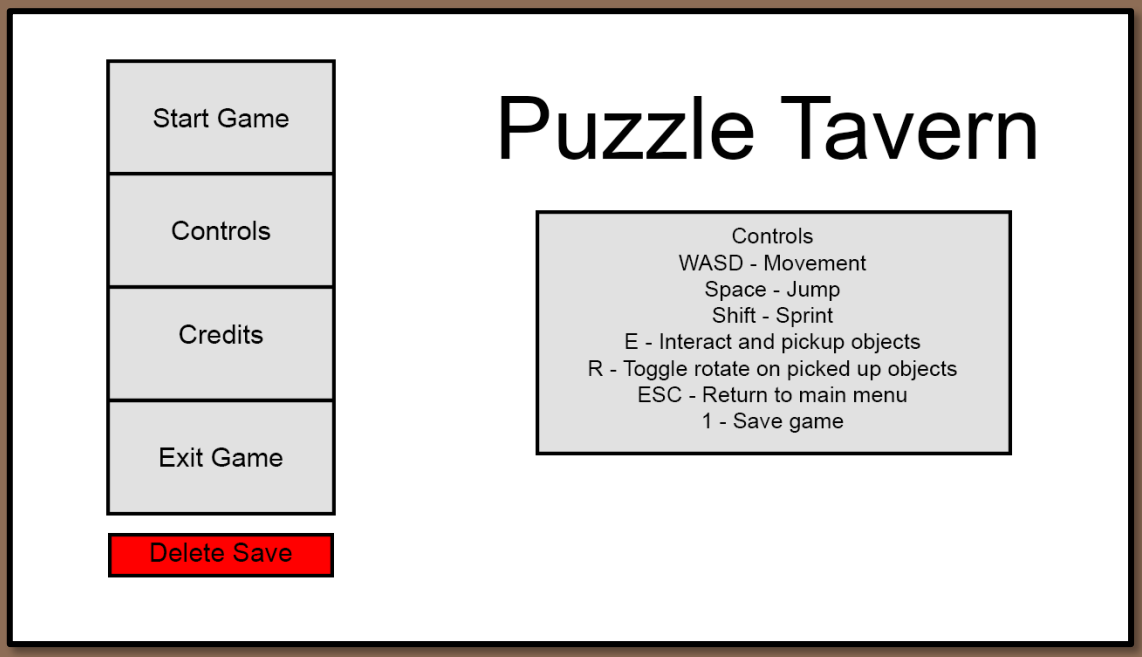
UI Design & UX considerations

Menu Interface

So, while puzzle tavern is not going to be very interface heavy, a main menu is still required to communicate certain information, like controls as well as other functionality like save management.

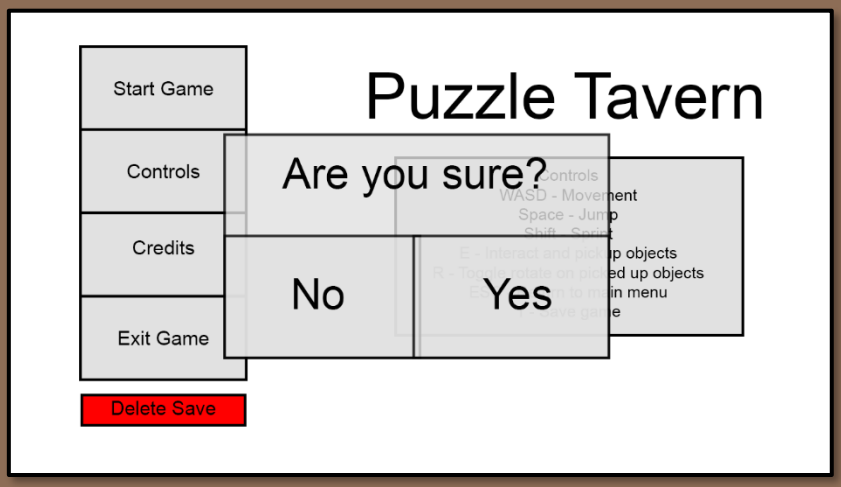
The only specific information/functionality I needed to display on the main menu was
- Start Game
- Controls
- Exit Game
- Delete Save data
- Credits

While I lack, the art skills to make nice looking UI I can at least make a nice functional interface that considers UX throughout. Here is a quick mock up of the main menu.



The buttons have been placed on the left-hand side of the screen, as generally speaking people read from left to right, so with this button placement I am conveying important information as quickly as possible when using the menu. An important thing to note however, is the controls menu on the right-hand side of the screen, will only show up once the 'controls' button has been pressed and vice versa with the credits menu.

As well as this, another important thing to mention is the use of red and the separation on the 'Delete Save' Button. I have done this as to not cause any confusion about the importance of this button and also to stop the player from accidently clicking on it and deleting their save. However, as a precautionary method I have also decided it would be a good idea to ask the player an additional time if they really do want to delete their save, as shown here.



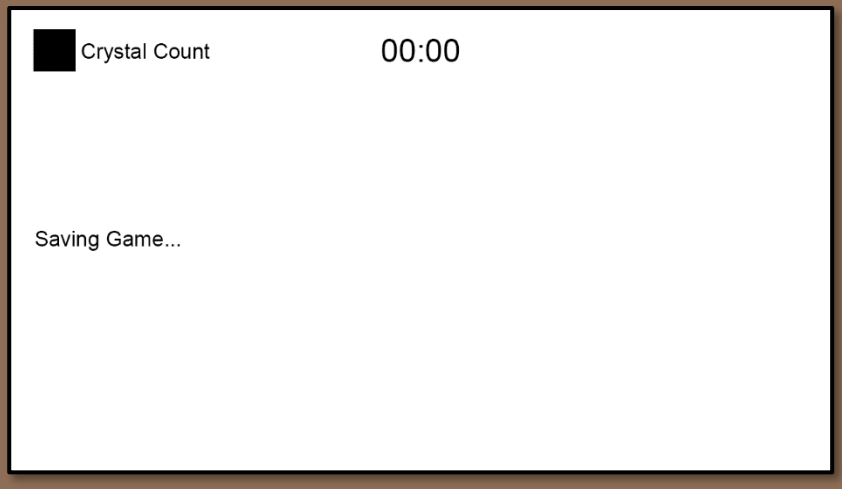
As you can see here, an overlay will appear over the rest of the menu to double check with the player if they really do intend on deleting their save. The 'No' and 'Yes button are displayed left to right so if the player does automatically click a button, they are more likely to hit no, meaning they will not accidently delete their save.

Visual Style

While I said before I am not really capable of making very artistically skilled looking visuals for my UI I can atleast put some thought into how I want my menu to look in-game. My main idea here is instead of having a blank background I will have the background of the UI be the actual environment the game takes place in, most likely the starting room of the tavern.

Hud

While I do not have much information that I need to display to the player there is a still a few little bits that would be quite important to know during gameplay.
The main features I need to communicate to the player are
- Countdown timer
- Crystal Amount
- Saving game



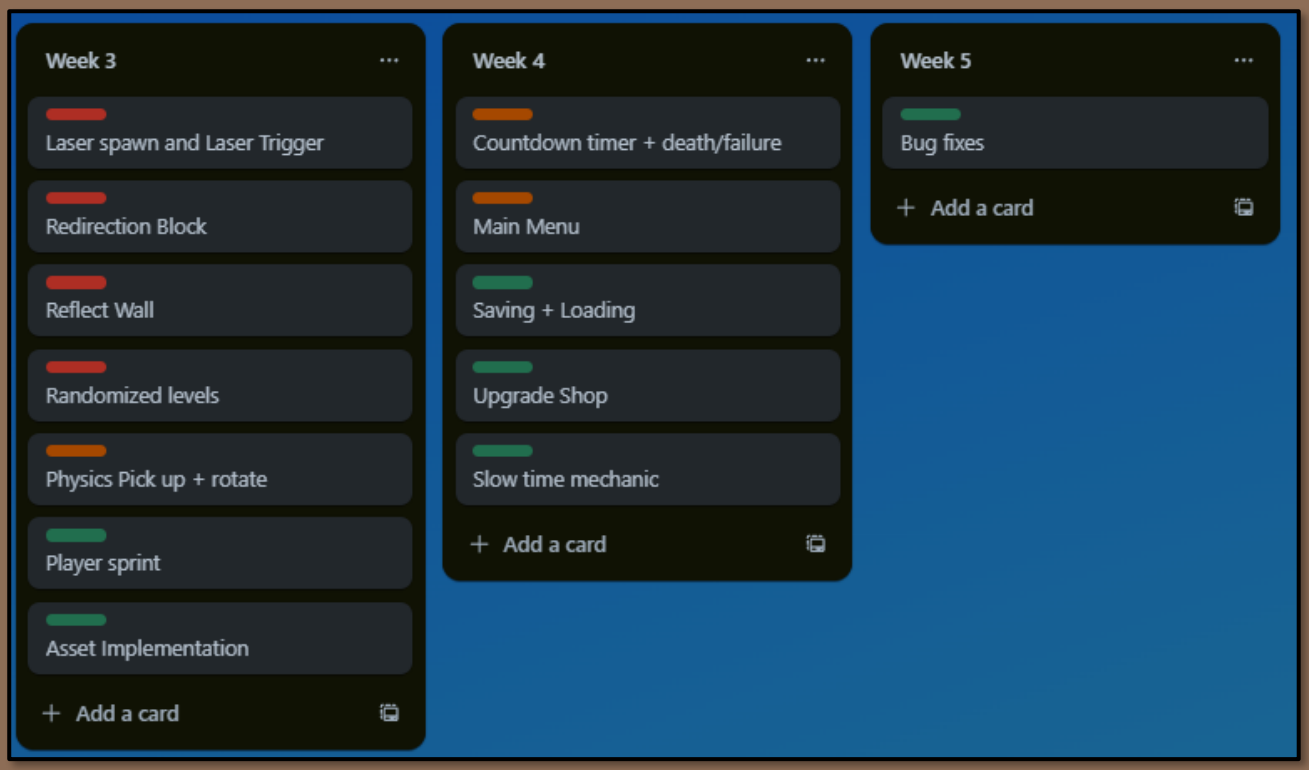
Here, while the HUD is simple, it effectively communicates all the information the player needs in a non-intrusive way.

The crystal count is probably the most important information for general gameplay as this will affect if the player can purchase upgrades later so it has been put in the top left. However, the timer has been placed in the middle as it is the most

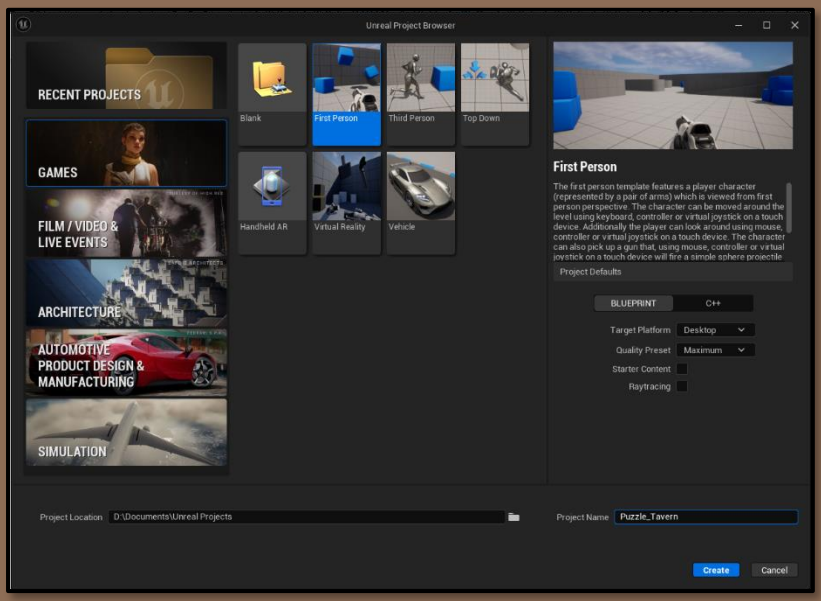
immediately important during the actual gameplay. It is also important to note that the save game text in the left middle, will fade in and out only when the game is saving and not remain there at all times. The black square next to the crystal count also will end up being an image of the crystal, when I end up implementing it into the engine.

MVP development

Before going into engine, I started off with developing a checklist on Trello so I could keep track of the tasks I need to help towards producing my MVP and keep my focus, so I do not lose track of what I am doing.

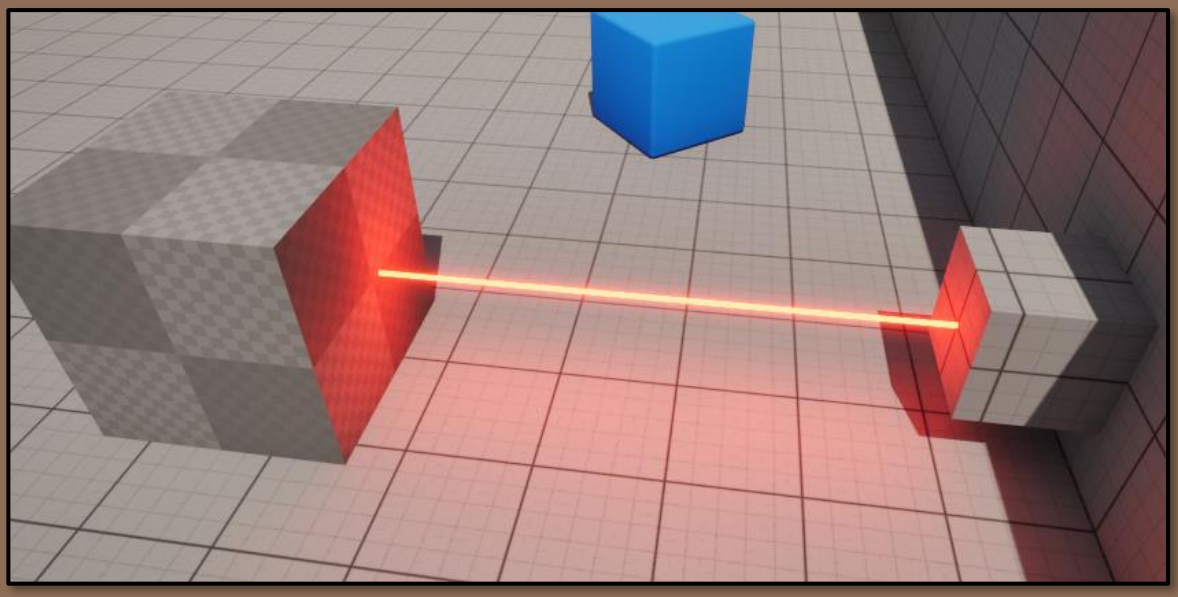


I have separated my MVP goals into the remaining 3 weeks of this project as well as labelled in them in colours to determine importance of tasks, since the laser is so pivotal to the gameplay that was the first thing I had decided needed to get done, as then I can also start building the puzzles.

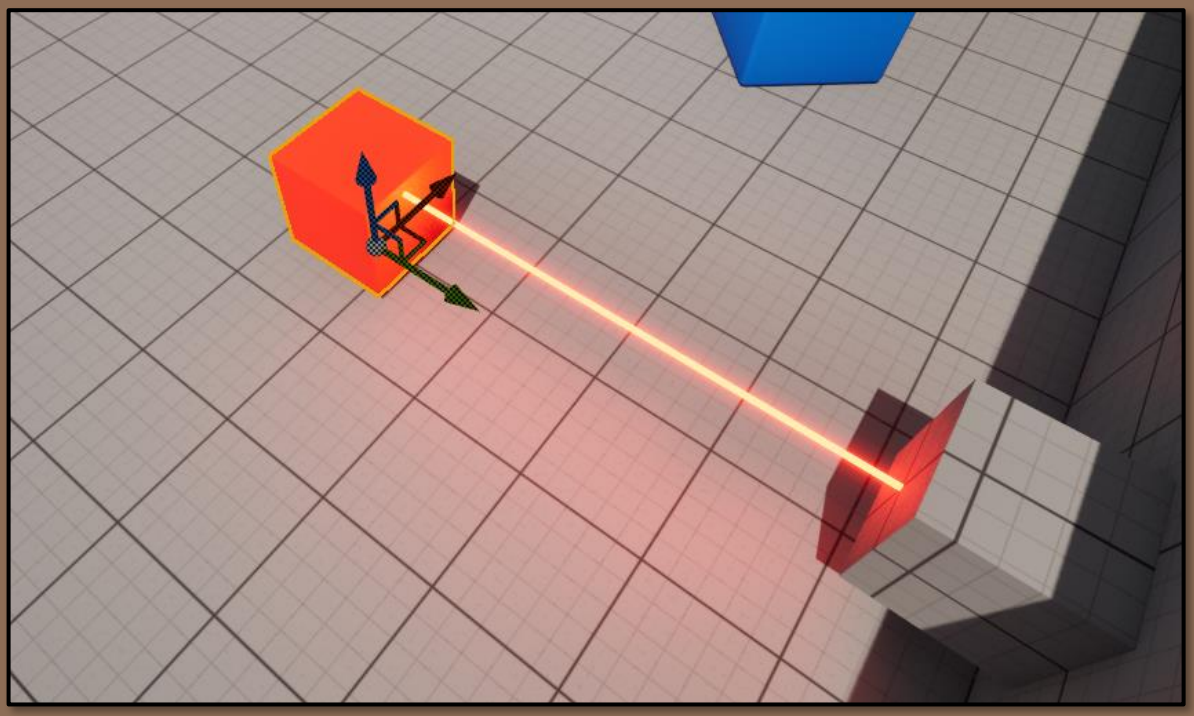


So, while setting up the project I deemed the best template to use was the first-person template as it allowed me to get movement and jumping done by default as well as the camera which was in the same position and had the same behaviour as previously discussed.

So, the first thing to get going was developing my laser system. The first thing to get working was the actual laser otherwise I would have no puzzles. Functionally, the laser will be a red laser that glows and shoots straight forward until it hits the closest object to block its path.

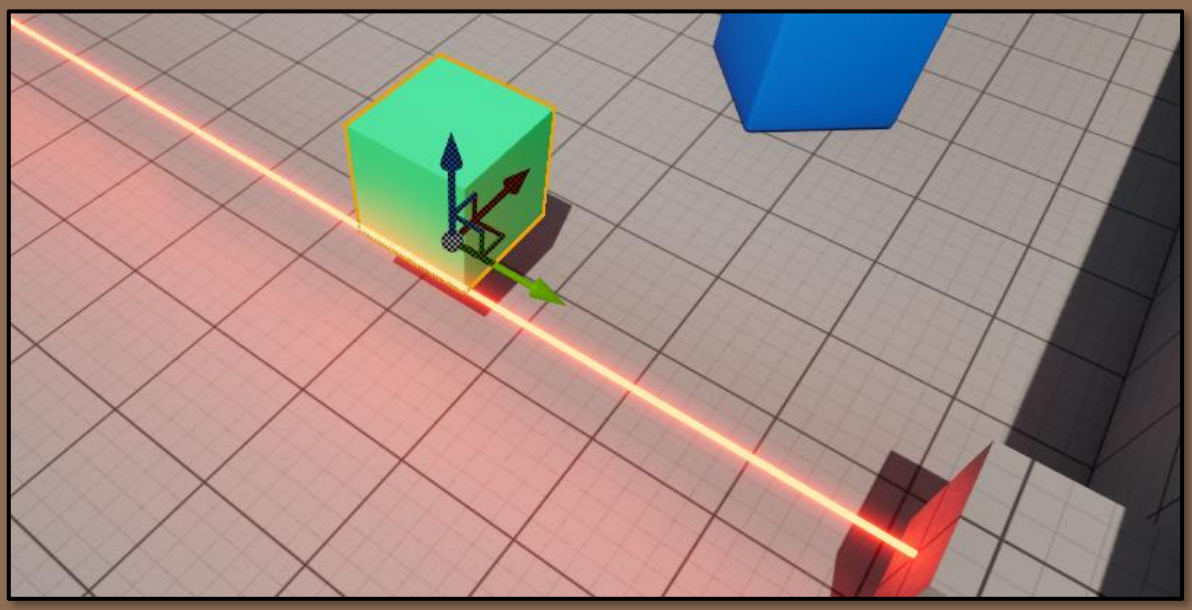


The laser I have created here, has no collision with the player and will otherwise not affect the player in anyway. While it is being blocked also, the laser will not interact with whatever it hits unless what it hits has a specific function like the redirection block, reflect wall or trigger that I am about to develop.
The next thing I needed to get of the way was the trigger block, as otherwise I would not have a way of

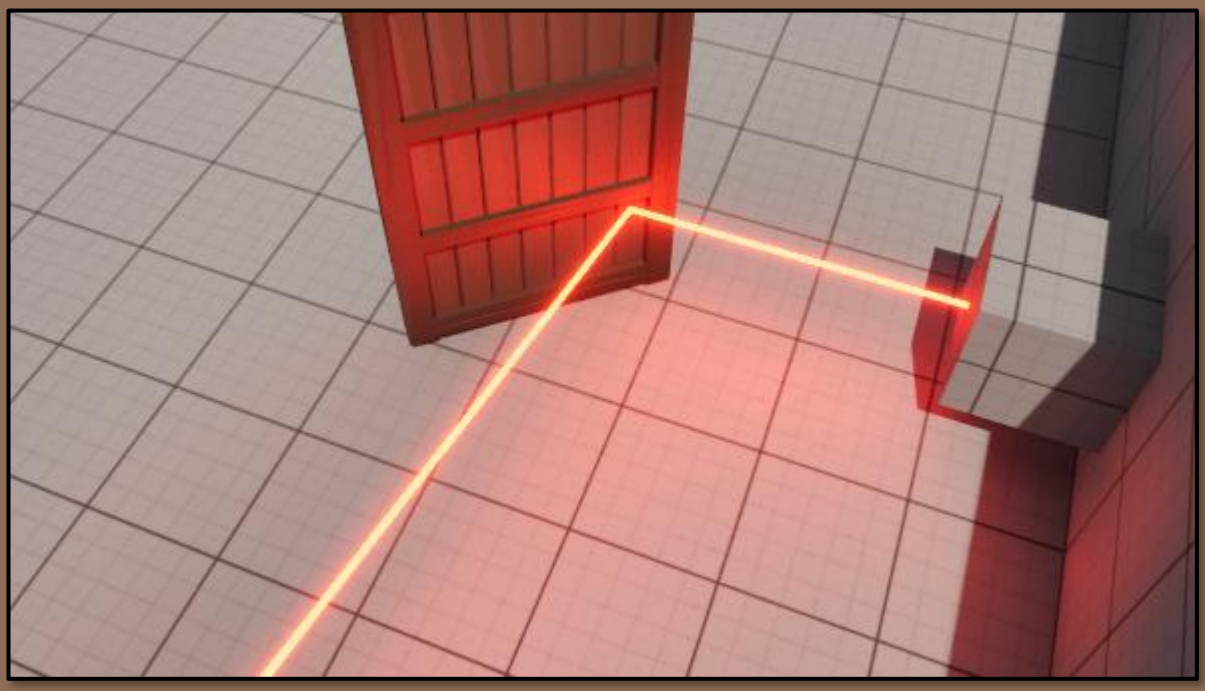


finishing each puzzle.

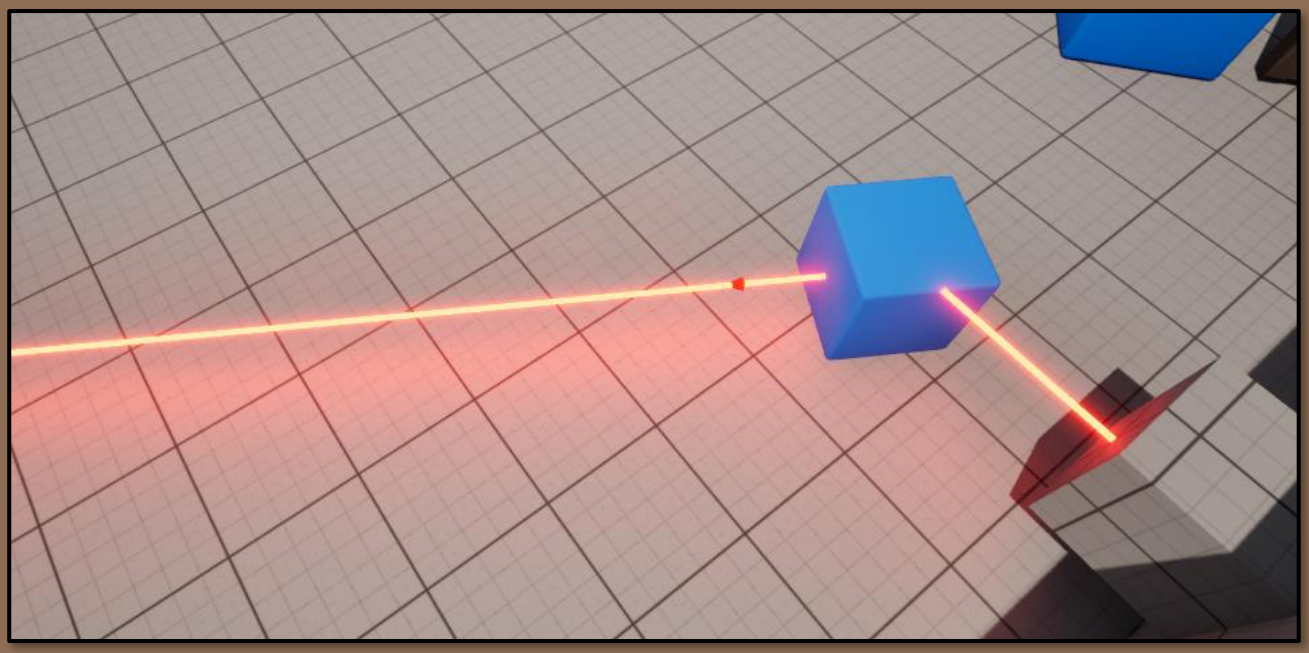
Here, the laser now hits a new cube, that lights up red while it is being hit by the laser. This is when the cube is powered and will trigger the ending of a puzzle. However, I will also have another trigger cube that will be toggleable and cause other events to happen throughout puzzles like moving objects.



This is the second trigger that will be available throughout the game. Once the cube has been hit with the laser once, it will stay a green colour unlike the previous trigger that reverts to its default state. This cube will be used to change the state of the puzzle, like move or rotate objects to further push the puzzle to is completion state.



Next, I wanted to work on the reflect wall as yet another vital component to the puzzles I have planned out. This reflect wall upon being hit by the laster will reflect the laser at an appopriate angle. This wall is unmovable however, unless the button for it is interacted with, where it will rotate 45 degrees.



The third main mechanic I needed to work was the redirection block, otherwise my puzzles would not really be that entertaining, plus it was in my plan either way. This block redirects the laser that hits it in a specific direction, in the middle of the block to make it easier to aim in the futuer.

This block will also be generating physics so it can be moved around, picked up and rotated by the player.

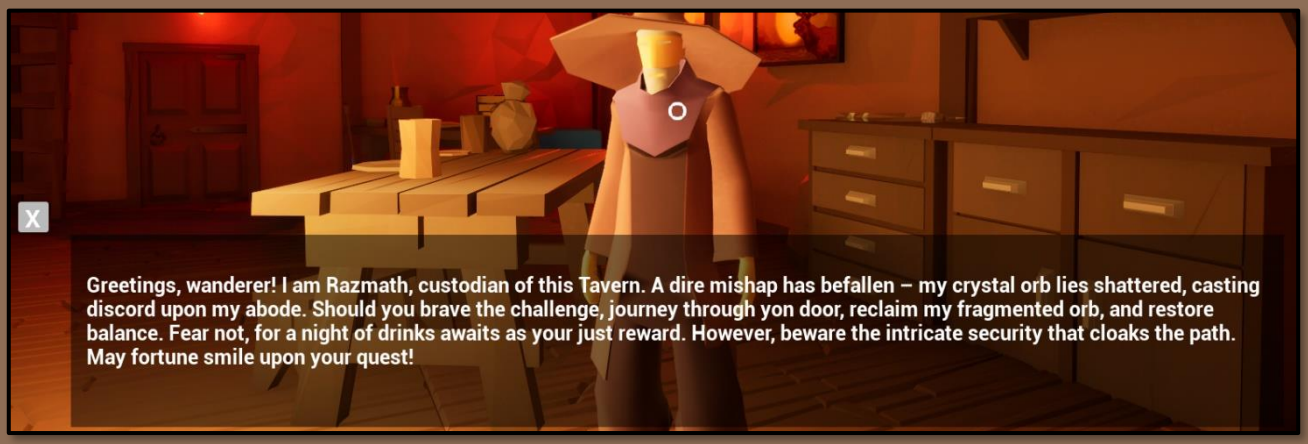
The next step was the randomization of puzzles that I wanted, so the game would really feel like a rogue-lite, as well as deciding how many puzzles needed to be completed to finish a run. At the moment, as I only had 9 puzzles to play through, I decided it would be best if the player only had 4 levels to complete before finishing a run, however in later renditions of the game I will likely expand this. I achieved this by creating an *array* of all of the puzzles the player could down and it would incrementally remove one from the array until it had removed 4 after each level. Once 4 were removed the player would move onto the final room.

Next, while a bit unconventional I decided I would go and find an asset pack so I could have a solid foundation for how my puzzles were going to look and feel to be in and implement my assets into the starting room so I could get a basis for how I wanted to populate my puzzles and make them feel lived in. I ended up going with a tavern low poly pack I had available to me and started to look at references for how to populate my level.

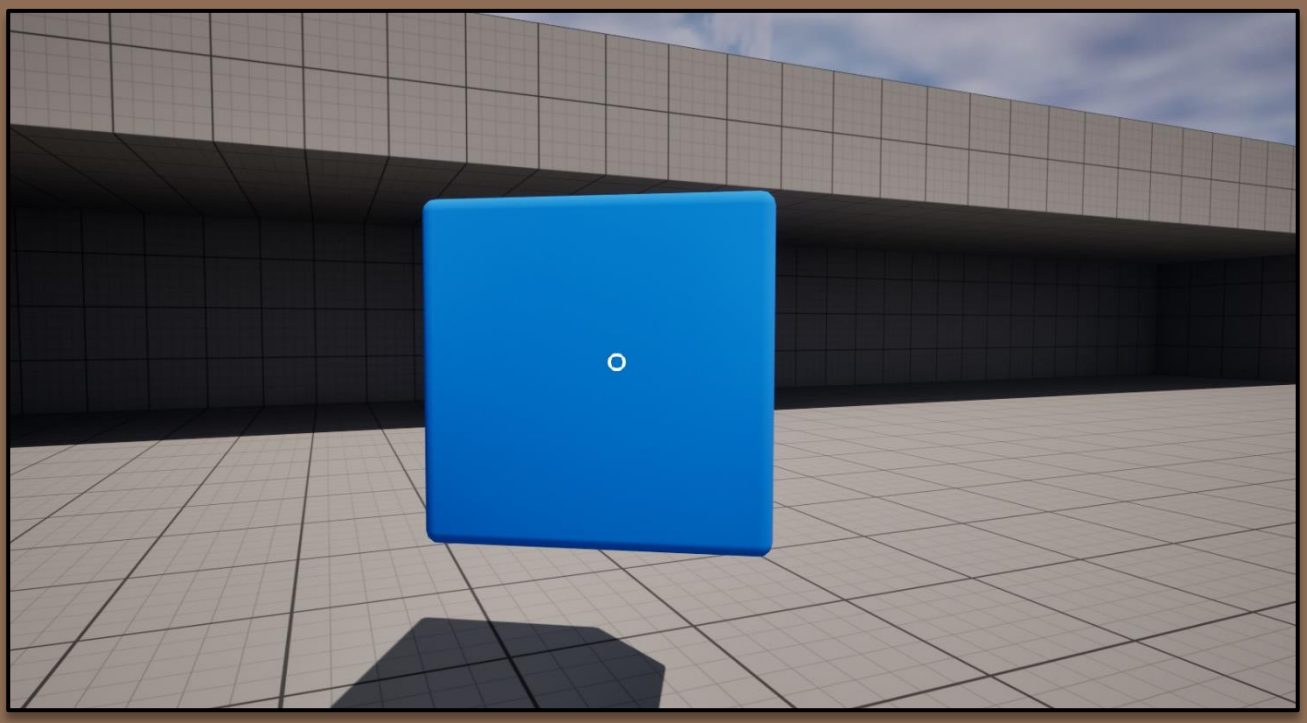
This was my final outcome, for the starting room and now I had this I have a good basis for how to populate the rest of my puzzles when I build them. While building this I got to experiment with lighting as well which is something I wasn't very comfortable with doing until now. As well as this, I setup a simple NPC that the player can speak to which in dialogue will give the player an idea of what they are doing in this tavern and why.

This wizard NPC, will tell the player they need to retrieve his crystal as he shattered it on mistake that is setting off his security system, hence the maze of random puzzles that are taking place throughout the tavern.

This will give the player a basis for why they are playing the game as well as give them a reason for getting to the end and solving all of the puzzles.

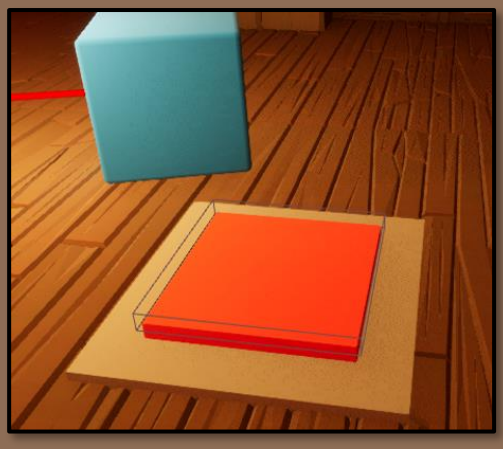


Now to finish off the week, I had to get my pickupable objects working with and implement the function to rotate them. Luckily unreal engine has an actor component called *Physics Handle* that helped handle a lot of the functionality I needed so I got this working fairly quickly.



I did initially get rotation to work however the cube specifically rotates using its own relative axis, which makes it a big weird to control so I decided I will mark this as in-progress and come back to it a little bit later.

Lastly I also need to develop the pressure plate that is required to finish puzzles as per my designs. This was simple enough. When the pressure plate has the player or some kind of object on top of it, it will move an object by parameters I can set in the inspector.



Design Issues

So, throughout development of my MVP this week, I was consistently testing my mechanics and figuring out if they felt nice. One big issue I noticed was due to my lack of puzzles it would be quite easy to repeat the same room over and over through the randomization. My solution to this however is make it so the player cannot repeat a puzzle during a run, so once they have played it, it is then removed from the last that the randomizer picks from when choosing a puzzle.

I decided the appropriate fix to this would be to stop the player from repeating levels in a single run, as I thought it would lead the player to getting annoyed if they had to keep repeating puzzles, especially since after they solve it once the puzzle would become drastically easier in repeated attempts.

As well as previously stated, my rotation was a little strange to control, so I will be fixing this in this in the following week by making the cube rotate on an axis relative to the player and not the cube itself.

Week 3 reflection

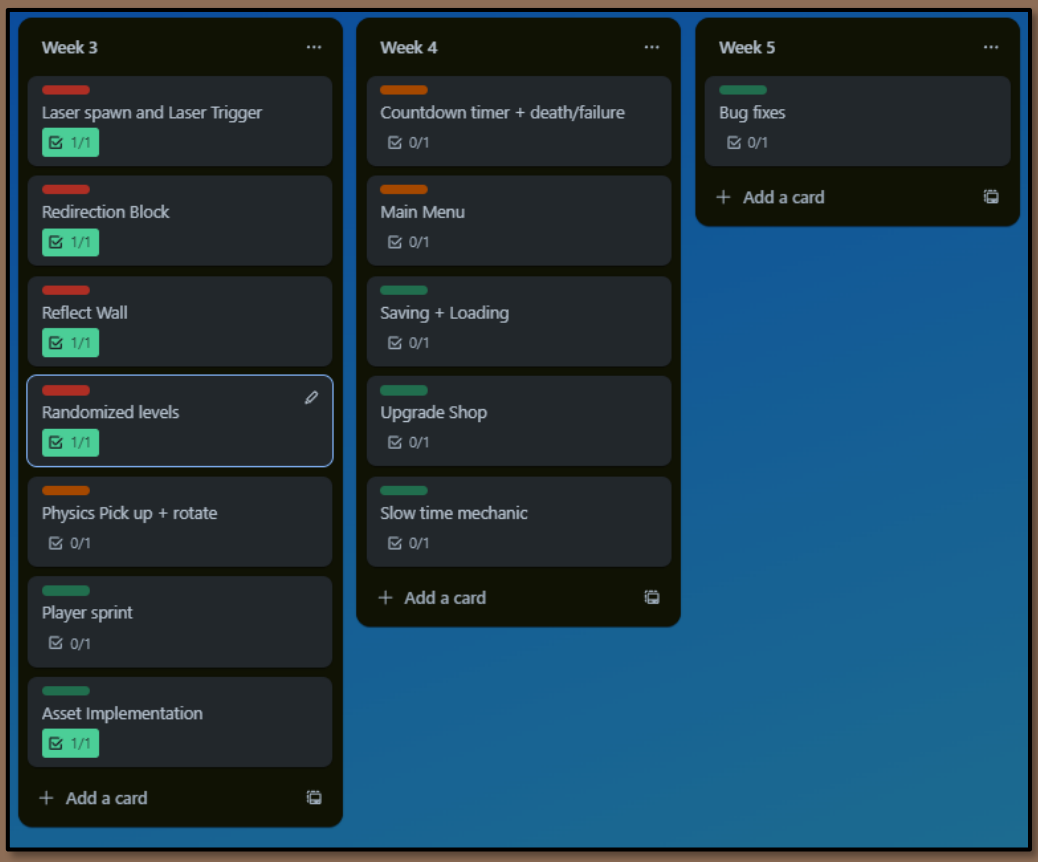
Most of this week was taken up with development of my MVP and I feel like I made good progress, especially in line with my plan I had established earlier in my Trello, while I did not quite finish my pickup & rotation system, I got most of the way there and it works for the most part it is just a little awkward to use so I will go back to it next week.

I am however, very happy with the asset pack I chose as it gives the game a nice visual style and makes the game feel more complete, ideally I would like to acquire assets for my laser mechanics but depending on what I can find this may not be possible.

Week 4

Weekly Goals:
- Further mechanic Implementation
- Build puzzles
- Menu setup
- Bug Fixes

Further MVP development



So, to make sure I am still following my plan I continued to follow my Trello board I created. I noticed immediately that I forgot to implement the sprint system last week, due to the other mechanics taking more of my time.

Just before I get started on my Trello plan, I wanted to build the puzzles as that way I would have gameplay I could properly test with my mechanics.

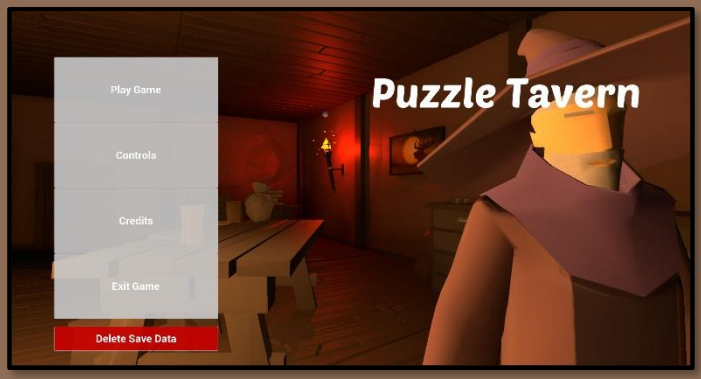
While I could have done this later, it provides me a more suitable testing environment for the mechanics I am going to implement this week.

So, the first mechanic I wanted to implement was the timer and death mechanic. This was important to implement as it allowed the player to fail during gameplay, which at the moment is not possible. This mechanic, however, was quite simple to implement as I simply just did a countdown as soon as the first puzzle started.

The time I set to complete the puzzles by default were 5:00 minutes however, late on in the game throughout the upgrade system, this will be changeable.

As well as this, when the timer hits 00:00 the player will not necessarily 'die', but they are being reset. What happens is the player gets a white flash on their screen and they reappear back in the starting room.

Now, with the next set of systems I wish to implement have to be setup in a specific order, meaning I need to setup the main menu next, so I can delete and load save data.



Most of the main menu was relatively simple to setup, as it mostly just consists of *buttons* and *widget switches* however, the background required a little extra setup. In order to capture, the starting scene with all of its animations I had to implement a render target camera that would capture the scene and take that captured image into a texture I could apply to an image for my video. As the render target would update when anything in the camera changed, it allowed me to have an animated background for my main menu. Personally I feel this makes for a more interesting menu design to look at.

As well as this, accurate to my previous design mockups, when the player clicks Delete save data, this

overlay menu, slides onto the screen from the left, via an animation to double check they wish to

delete the save data.

As well as this, when the controls and credits button is pressed, the menus on the right show up, displaying the relevant information.

Now upon finishing the menu which did not take a lot of time at all I can start to implement the save and load system, so the player can exit the game and retain the progress they have kept. While this is simple in concept, I had not implemented such a feature in previous projects, so I had to use a few new nodes, *Create Save Data, Load Save data, Does save data exist*.

Functionally the save data was quite simple as all I had to do was save a variety of variables which allowed the player to retain their progress on subsequent runs of the game.

Now as the save data mechanic is now fixed I can setup the upgrade shop, which allows the player to have some sort of meta progression while playing the game, another rogue-lite aspect.

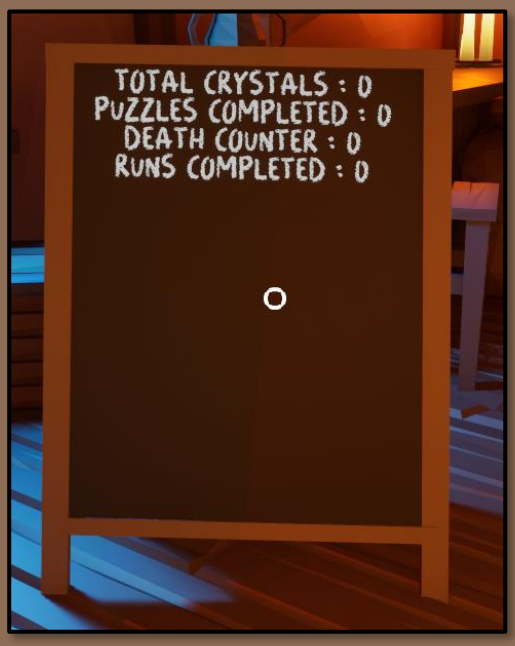
The basis of this does, however, require a resource which I decided would be appropriate to be little shards of his crystals that the player can then use to purchase these upgrades.

After the first puzzle is completed, regardless of if they fail the run a new room will appear in the starting area. This room will lead to a shop where the player can buy these upgrades.

As detailed before, the two upgrades will be used to give the player the ability to slow time and the

second upgrade will increase the maximum time they have to complete the entire run. This gives the player incentive to do repeat runs and earn the crystals required to get each upgrade. After each upgrade is purchased the next one shows up in its place with a new price and better effect.

Now the upgrade system is designed I needed the slow time upgrade so it would actually mean something. However, this mechanic was quite simple to make as it just required me to use the set *global time dilation* node in order to slow down time for a short period of time.



After developing, all of these features I noticed I had a little extra time left, while I was waiting for feedback from my peers, so I developed an additional feature. This feature is a chalkboard housed within the starter level that tracks various stats between each run, so the player can see the progress they have made while playing.

While this is not a neccesary feature, I think it adds to the player engagement because they can track how good or bad they are throughout the game.

Player-testing and feedback

To make some clarifications on the timeline of my playtesting period, all of the playtesting was done on a build prior to the upgrade system being implemented in the game

Now with player testing, I went about these multiple ways, I had given a build of my game to a wide group of my friends who also share a similar experience in Game Development to play the game and provide feedback on design and bugs, as well as I put this version of my game into the Grads In Games QA week thread so anyone on that server could provide feedback.

From both of these sources I got very good feedback regarding playability, design, and bugs. One major piece of feedback was that the relationship between Puzzle and Roguelike was not that great. However, as the build that was shown to people was an earlier build, I did later implement meta-progression throughout the game to solve this problem.

The next set of feedback mainly related to bugs and awkward mechanic feel. The first one relating to the pressure plate. The pressure plate, if it had multiple objects placed on top of it would cease any of its behaviour if only one of the objects were removed, even if another object is still present.

As well as this, there was also a bug where if the reflect wall was hit at a certain angle it would no longer reflect the laser. Another thing that was mentioned was the rotation feeling a little clunky and awkward to use, something I did already know about but had yet to fix at the time, luckily I have no fixed it, before it was rotating based on its own individual axis meaning it would rotate differently depending on where you were compared to the object, but it now does it relative to the player meaning all the rotations make sense and are much easier to use.

I did also receive some feedback based around the upgrade shop being a little unclear and also boring as it wasn't decorated at the time of my playtest. To solve these issues, I have included a little UI box that appears when the player approaches each upgrade column describing what they are buying, as well as adding the crystal image next to the number cost of the upgrade, so the player can now tell

more clearly what they are spending and buying.

The last bit of feedback that I was given was mainly relating to, there not being enough in-game feedback for when a puzzle is completed, my redesign for this was to inside have the exit door, automatically open instead of having the player do it, when the puzzle is solved, additionally I also play a little sound effect when the door opens to provide a little more feedback for the player.

Week 4 Reflection

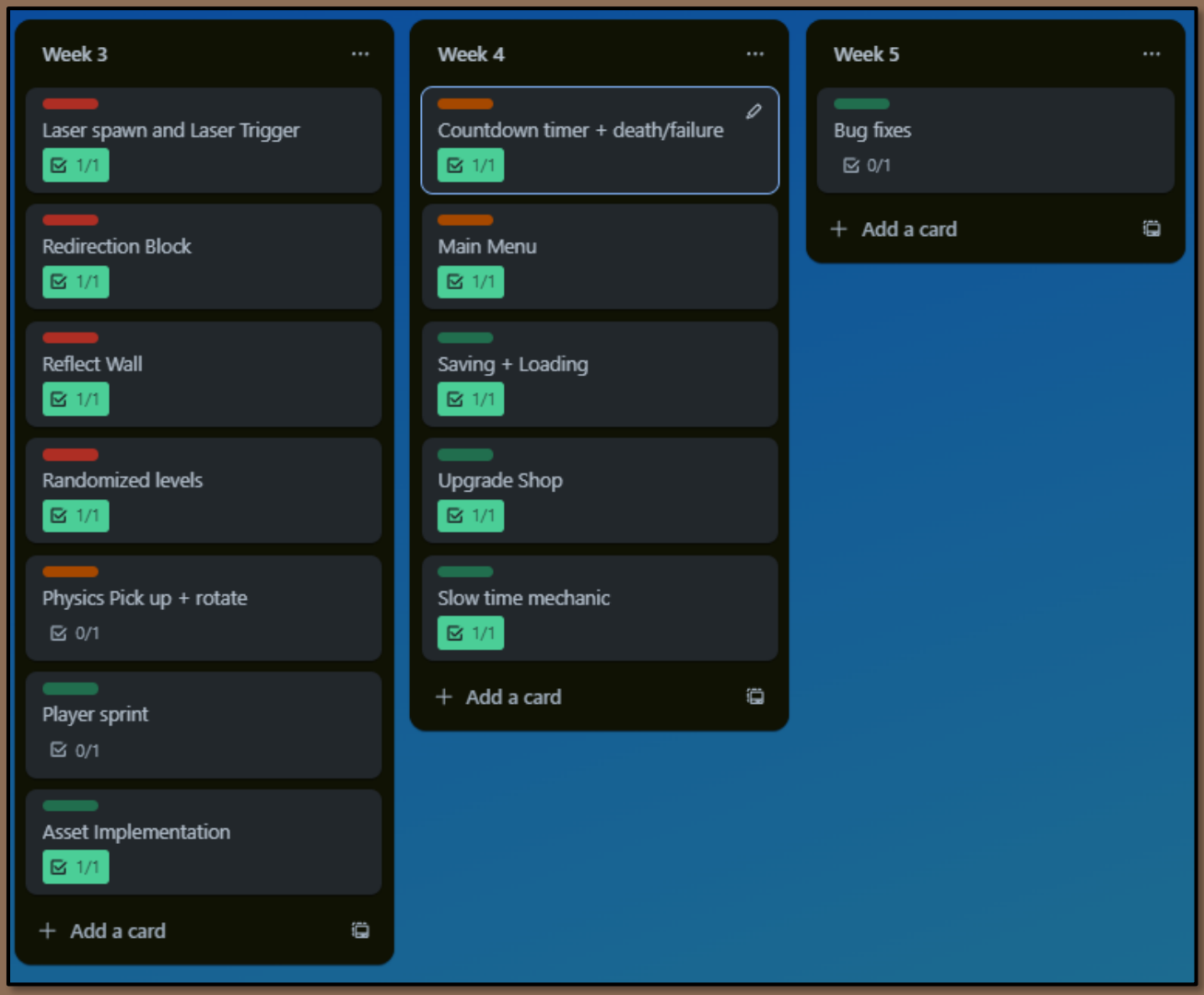
Now, week 4 has ended I have a lot of things to think about. I am very glad about my progress on this project and now everything is finally starting to come together, and my game really feels like it is starting to reach its final product. Some of the feedback I got really helped me see things in a different way and allowed me to expand on my game in a really positive way, while I did have some issues here

with my design and bugs, I got these all sorted with ease, and this allowed me to make more progress than I really thought I was going to.

Next week, I am going to finish up on polish with my prototype and then plan out and make the showcase video for the submission.

Week 5

Weekly goals:
- Acquire and implement music & SFX
- Plan and create showcase video.



Sound Design

With the last week in mind, I have checked everything off my list. However, I realised I was missing any kind of sound effects from my game, so this is an additional task I need to complete. While my games main focus isn't how immersive it is, I still wanted sound effects to make the game livelier and nicer to playthrough. So, the audio assets I acquired were

- Tavern like background music
- A Ca-ching sound effect when purchasing upgrades.
- An old man grumble.
- Walking sounds.

The old man sound effect and the walking sound effect, when played randomize from a selection of 4 or 5 different sound effects as to not get too repetitive while playing.

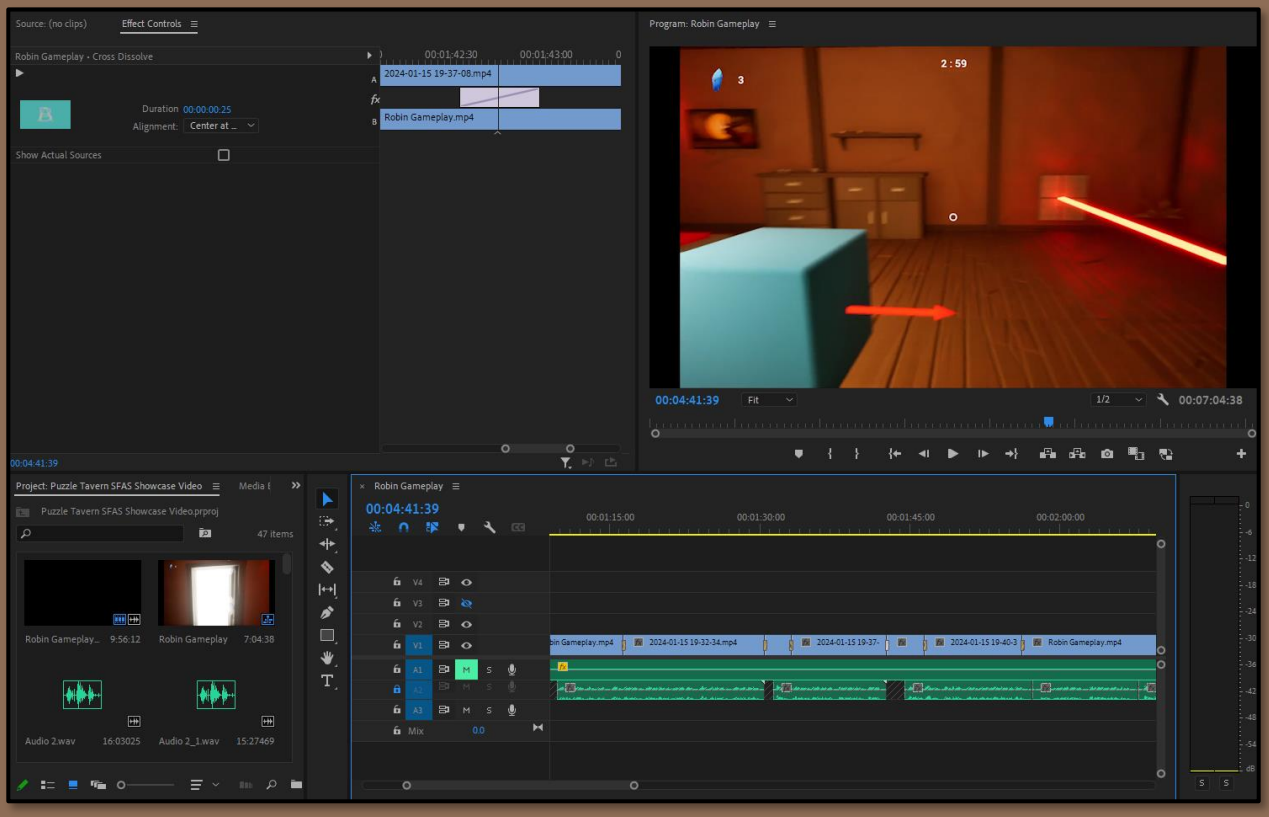
While I only decided to implement a few different sound effects I believe they really add to the games polish and feel.

Showcase Video Development

Showcase Video Plan

- Introduction
  - Introduce Puzzle tavern and quickly summarise what my submission is for and briefly explain my goals
- Part 1 of video
  - Explain the project brief and how my project adheres to it.
  - Talk about my two concepts/genres and why I am combing them.
  - Briefly describe the mechanics and concepts I took from each genre.
- Part 2 of video
  - Go over in more detail what systems and tropes I took from each genre and how I combined them.
  - Talk about the challenges I faced during ideation and how I overcame them.
- Conclusion
  - Summarise what I learned and what I have managed to deliver.

I quickly wrote up a script and got to filming footage for the video. My main goal of the video was to display gameplay as well as talking through my design choices.



As well as my previous plan, I have also included a gameplay video going through one run of the game and showing features that only appear after at least one puzzle is complete.

Evaluation

What would I have done differently if I had more time?

While I do think I scoped quite appropriately for the time I was given to develop the project, I would have really liked to have designed more puzzles for the game, as of the end of the project I only have nine puzzles total in the game while that isn't a bad amount of puzzles to have the game would only play and feel better the more puzzles there are to provide more variety to each run of the game.

I also would have liked to identify and implement more upgrades to be available at the upgrade shop, as while I have two upgrades now, they are quite similar and only have vague use cases, especially due to the limited number of puzzles.

I would also like to ideally implement a stage system similar to 'The binding of Isaac' where each stage would implement a new mechanic or different implementation of a current mechanic to spice up the gameplay, while also more linearly increasing difficulty.

Design issues

One issue I do sort of have with the final outcome is, even while I have broadened my understanding of puzzle design there are still some issues with difficulty, I found while playtesting and those more experienced in the puzzle genre found all of the puzzles quite easy to complete which lowers the tension while playing, so this is definitely something I would need to focus on more and do even more research than I already had done prior to the project.

I also think there is not enough distinction between the two upgrades the player can buy, as increasing the timer and slowing down time, while different do end up leading to the same outcome just in different ways. Being able to buy both upgrades mostly just leads the player to massively increasing the time they have to beat each run, and as stated before since my total difficulty is quite low, it makes the game a lot easier than I originally intended it to be.

Final thoughts

Despite my grievances stated above, I do still believe I hit my objective and successfully combines the two genres I intended to combine with my project. While there is room for improvement, in the 5 weeks I had I think I have made a lot of progress and if I choose to have the foundations for a project that could easily become bigger and even more interesting if expanded upon.

Itch.IO Link

This is the link to my playable prototype.

https://quertx.itch.io/tavern-puzzle