FYP Artefact Development Report

By Joshua Crabb

Contents

Overview

This project focuses on the creation and implementation of a dominant diegetic interface that can be used within a puzzle game, while also working on creating a prototype that best showcases the feature.

This interface is themed around the retro gaming handhelds of the past, showing a unique core mechanic that pushes the topic of UI in games.

This document will cover:

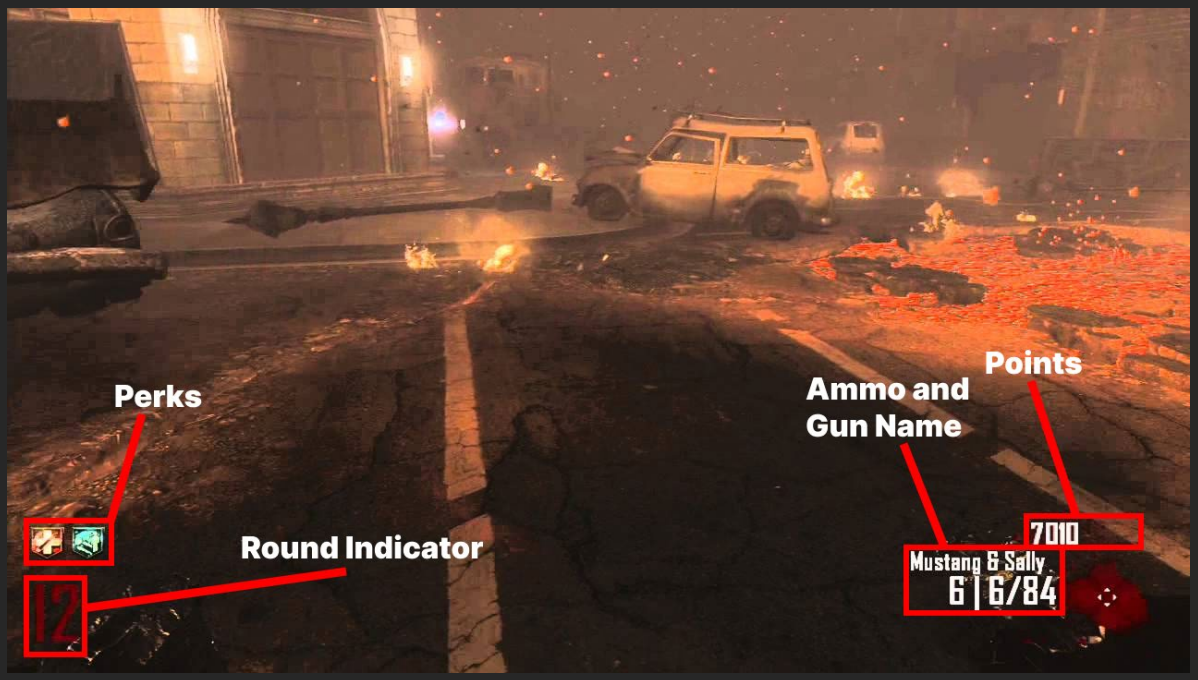
- The making of the Diegetic Interface, how it was designed and how it was implemented.
- How the secondary mechanics were made and why they were added.
- How the levels were designed, what they teach the player and what puzzle design methods are used.

The Diegetic Interface

“UI is the bridge between the player and the game world” (Angelikatosh, 2025), a way for players to interact with the game, to understand and to play it. Traditional non-diegetic UI has info communicated directly to the players screen, with it being used multiple times throughout games in the industry, for how easy it is to showcase information to the user. Games like Baldur’s Gate 3 *(Larian Studios, 2023)*, World of Warcraft *(Blizzard Entertainment, 2004)* and RuneScape *(Jagex, 2001)* are a great example of non-diegetic UI, being used in high quantities.

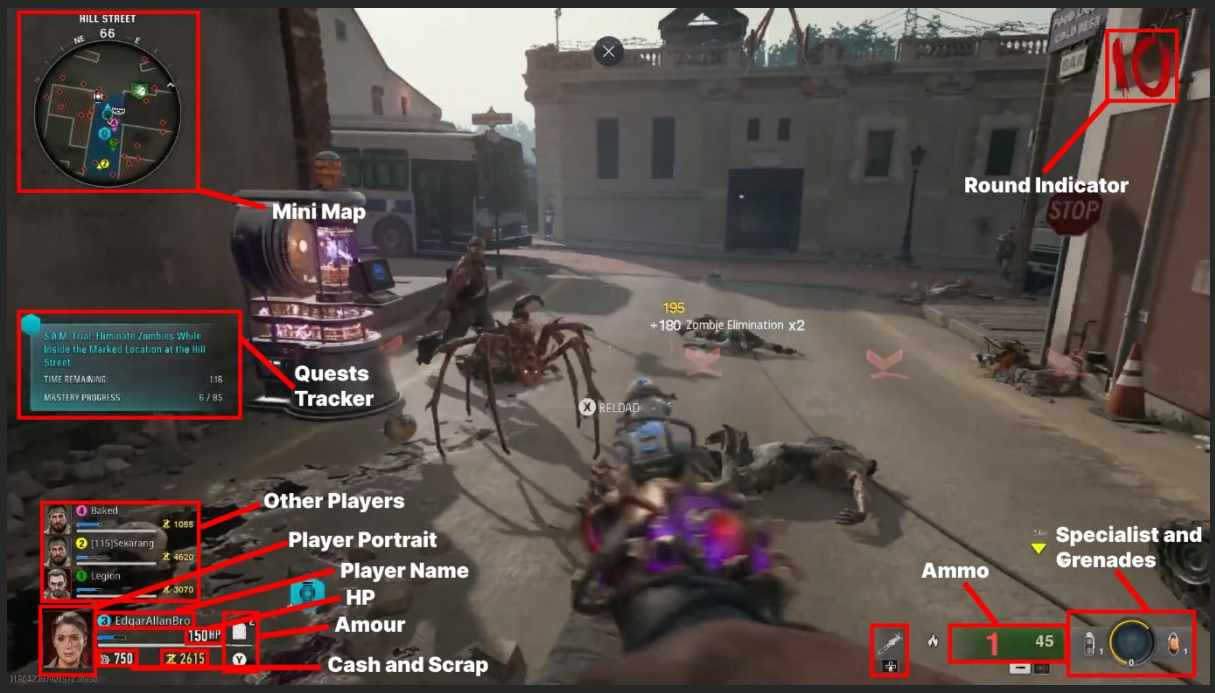
Non-Diegetic and Diegetic UI

Although non-diegetic UI can be used to communicate to the player, large amounts can cause issues. From a lack of immersion in the player, to compromising the design of gameplay.
In Call of Duty Black Ops 2 *(Treyarch, 2012),* and throughout the series there has been a game mode called ‘Zombies’. In this mode they had a great balance between both non-diegetic and diegetic UI.



The HUD was simple, keeping all the info, the player needed to know out of the centre of screen, while also not overloading the player.

The use of diegetic UI in this game also adds a gameplay feature with perk machines and pack-a-punch. When using these machines, it plays an animation that leaves the player in danger since they can't fire their gun. This communicates the info to the player very well and incorporates a risk and reward element to the feature.



However, in more modern versions of the mode there have been some changes to the UI. In Call of Duty Black Ops 6 *(Treyarch, 2024),* the HUD is cluttered with info that the player might need but is not necessary for gameplay. This HUD overloads the player's cognitive load, and sometimes *"less is more". (NovemberHotel, 2024)*

In Call of Duty Cold War *(Treyarch, 2020),* the diegetic UI was replaced with a non-diegetic UI. This breaks the pace for the player and makes pack-a-punch more confusing to use. Additionality, it removes a core game design principle of risk and reward.

Most games that have been released, creatively use all 4 UI types to make gameplay flow better and information easier to digest. However, *"for decades, game interfaces were often*

understood as an obstacle to the experience: health bars, mini-maps, and inventory screens floating above the action, breaking the "fourth wall"." (Felix Kalkuhl, 2025)

Diegetic interfaces on the other hand restore the fourth wall, immersing the player into the world. Diegetic UI occurs within the context of the story while also existing in the game environment. Horror games use diegetic interfaces to immerse their players, which intensify tension and increases the overall fear factor, with Phasmophobia *(Kinetic Games, 2020)* being a great example of this.

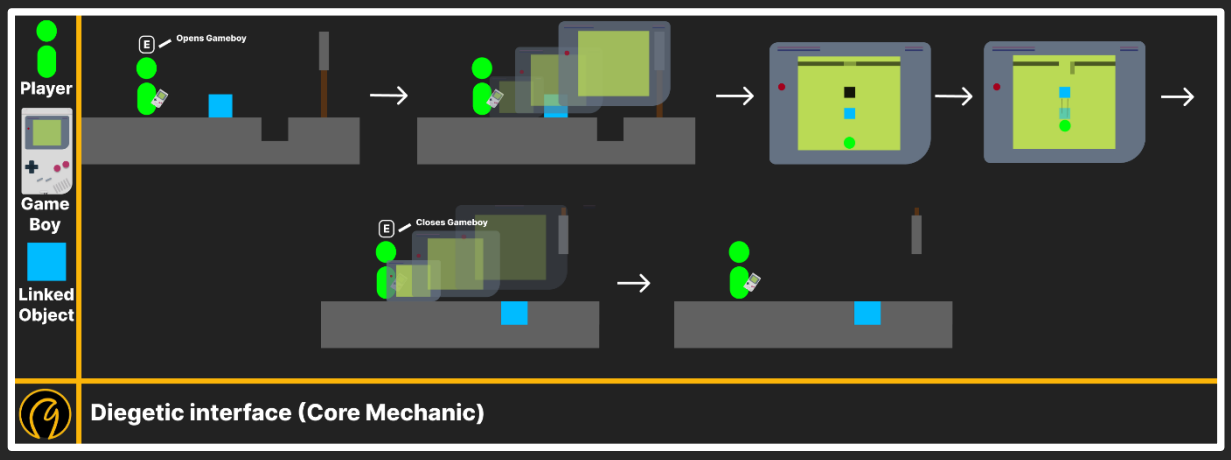
Diegetic UI can also be more enjoyable to interact with from a player's perspective as the interface could be added in the gameplay, games like circuit breakers *(Supersonic Software, 1998)* uses this for their character selection, allow for players to explore their environment.

Other examples of good diegetic interfaces are Dead Space *(EA Redwood Shores, 2008)* incorporating the health, inventory and ammo in the players amour. Fallout New Vega's *(Obsidian Entertainment, 2010)* Pip boy uses diegetic UI for its pause menu, however since the device is always on you, players will naturally get attached to it. This aspect of the Pipboy is an element of the artefact, making devices that people can get attach to.

The diegetic interface takes the form of a handheld gaming device, where the character can play a top-down 2D adventure game, with inspirations from Legends of Zelda the Links Awakening *(Nintendo EAD, 1993).*

The device mimics the environments and room layout. If the character interacts with an element within the gaming handheld, this changes that object in the real world.

Design Diagram



Figma

Figma

View Interactive Figma

Render Targets

The interface was implemented, using render targets. Render targets can capture a different area in the world and display it as a texture. This allows for many systems to be made, like a security camera system.

Why Render Targets?

This feature could have been made in a multitude of ways. The first option used a widget to display the interface. However, this was quickly shut down as that would mean storing the character data within a widget. This would make the feature incredibly complicated and create a very difficult hierarchy to maintain. Other issues that stem from this option are the different actors that would need to exist on this screen, which means even more functionality in the widget.

Another idea was to place the actors on the model itself however Unreal's physics system caused this option to be scrapped very quickly.
Unreal's render targets offer a good number of resources that help with the implementation of the feature and other aspects of the project, including the level design. Render Targets use the camera components, which allows for different camera settings to be applied. They

also allow for post processing effects which can help with the feel of the feature. The component also has an array that only renders what's in it, which can massively help on performance.

In terms of performance, the only risk to using render targets is that they could be very expensive on the system. If the render targets were created and destroyed, *"this could be exerting the memory beyond its capacity" (dhashuml, 2025).* To limit this risk, there is only one render target running in the game.

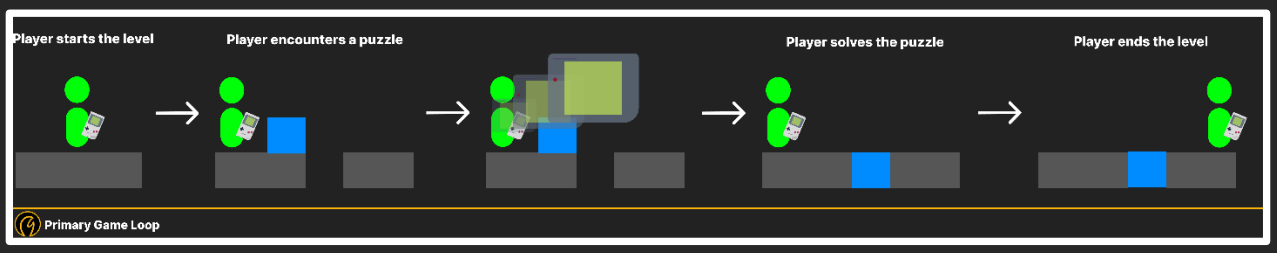
Game Loop

This project follows the style of a combinatorial style puzzle game, taking one puzzle system and explore that concept in depth *(Naomi Clark, 2019), (Herman Tulleken, 2011).* Games like Portal *(Valve, 2007)* and Viewfinder *(Sad Owl Studios, 2023)* take full advantage of this, taking the time to teach the player this new feature, making it feel like less of a mechanic but an extension of the player's skills.

The game loop for the project follows a similar structure of other combinatorial puzzles.

- Player enters a room.
- Player encounters a puzzle.
- Player solves the puzzle.
- Player leaves the room and enters a new room.

Even if there are variations or breaks to this loop, the game should keep coming back to this as its primary loop.



Mechanics and Systems

This section focuses on showcasing the technical and design decisions for:

- Character Features
- Level Features
- UI
- Gameplay Systems

Character Features

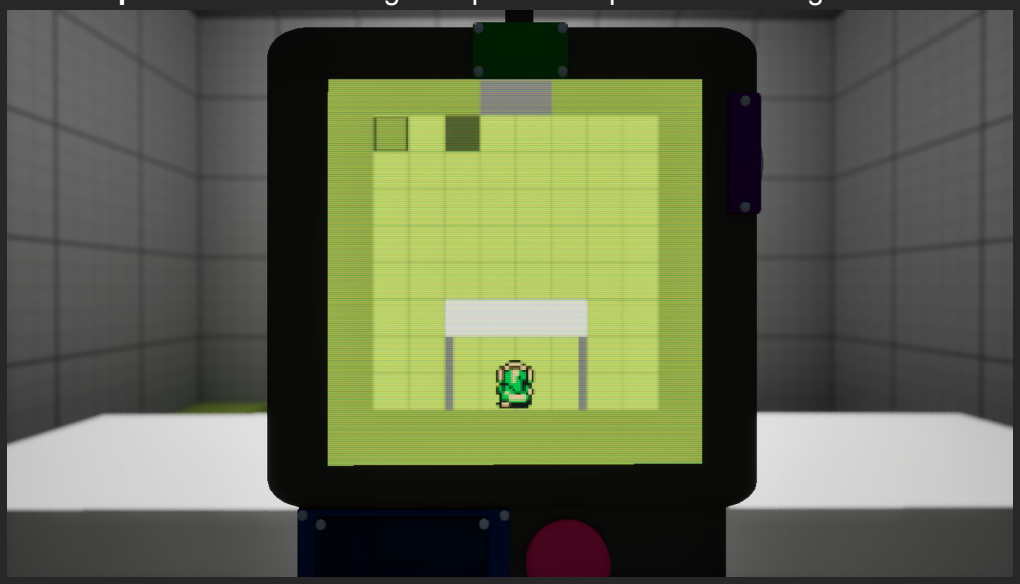
Gaming Device

The device has been designed in such a way to allow for the best experience possible for the players, allowing a connection to be made between the device and the user.

This feature had a lot of risk going into it, with players potential feeling disoriented from switching perspectives, the LED screen causing eye strain and the feature coming off as a gimmick.

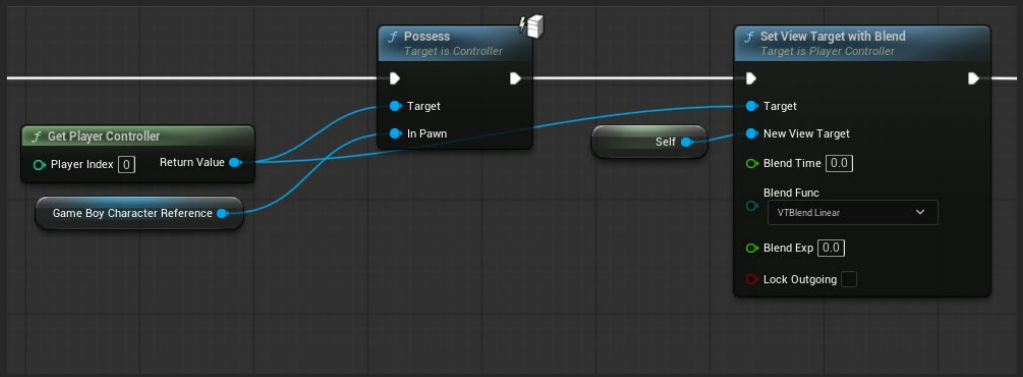
To resolve this, these were put into place:

- **Well, positioned** – While the screen is either active or idle, the player should always be able to see the screen.
- **Focus** – To help stop disorientation, there is a background blur added to give better focus on the system.
- **LED Filter** – Has been used to evoke nostalgic, however this has been tested and altered to get the perfect balance.
- **The Shape** – The vertical length helps with its position allowing for it to be visible.



How was the feature set up?

- While using a render target, the first-person character possesses another character within the render target.

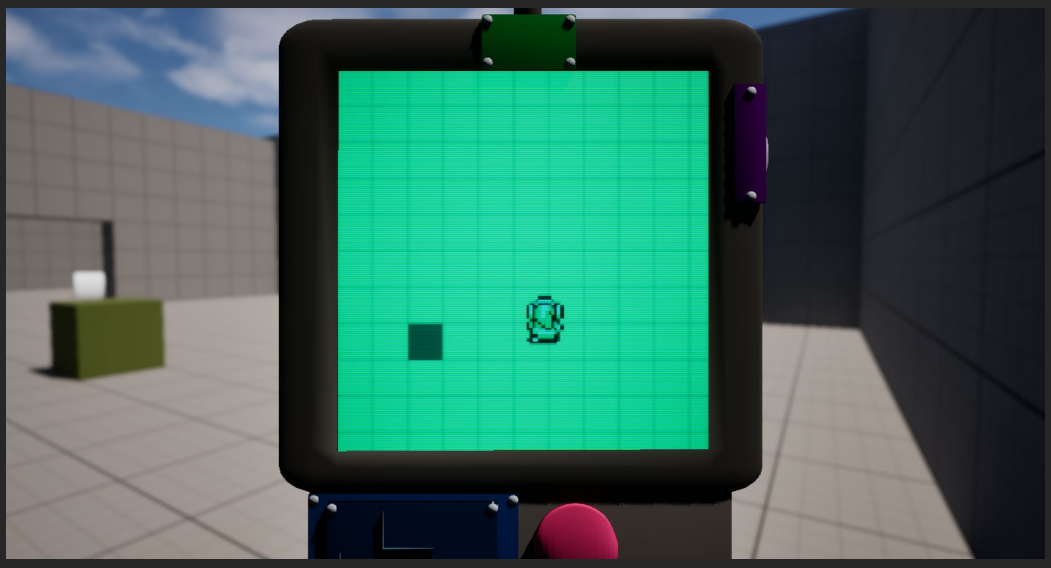


- The top-down character shares the same Inputs as the first-person player, allowing for a seamless transition between controls.

Colour Switching

The colour switching mechanic further adds to the diegetic interface, adding more interesting ways to solve the puzzles.
This feature is inspired by the colour dungeon within the Legend of Zelda the Links Awakening DX *(Nintendo R&D2,1998)*. Using the brand-new Gameboy Colour *(Nintendo,1998)*, developers created a dungeon using the colour to solve puzzles.

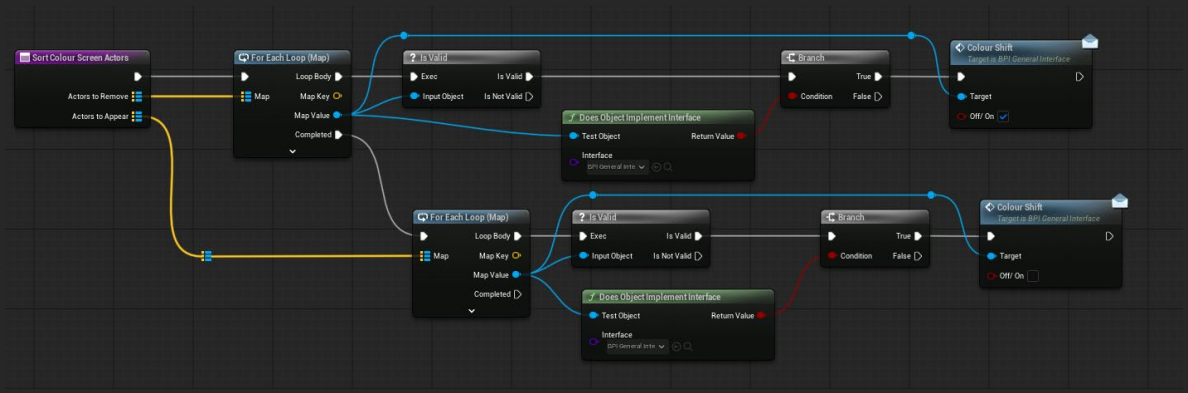
This feature will give the player the ability to change the screens colour. Once changed, it will change the actors in the world. There are 4 different states, green, yellow, cyan and red.



How was the feature set up?

- The render target has its own post processing volume, creating a custom function to set the colour on each switch.

- To store the actors that are required to be removed/added to the world, a map was used. This could be used to obtain both the location and object reference data.
- To communicate with the actors that needed to be removed, a interface was used, to run the maps through a for each loop.



Level Features

These features appear throughout the game's levels, to support the core feature and add to the level design. These mechanics are smaller in scope, however considering the core mechanic can be a little complex, the simple nature of these features will help the game to be more accessible.

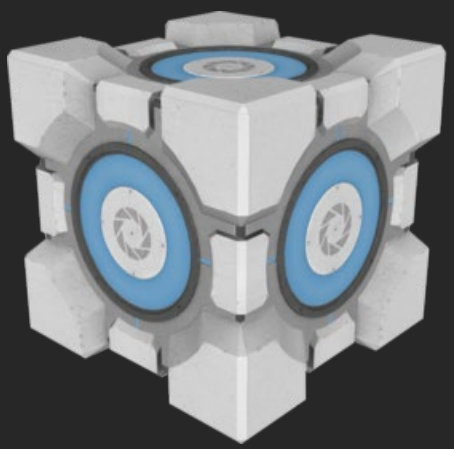
Portal's first entry does this as well. The games core mechanic is incredibly unique; however, it can get quite complicated. Throughout the game the other mechanics the player encounters are:

- Cubes
- Buttons
- Projectile Energy Ball
- Moving Platforms
- Enemy Turrets

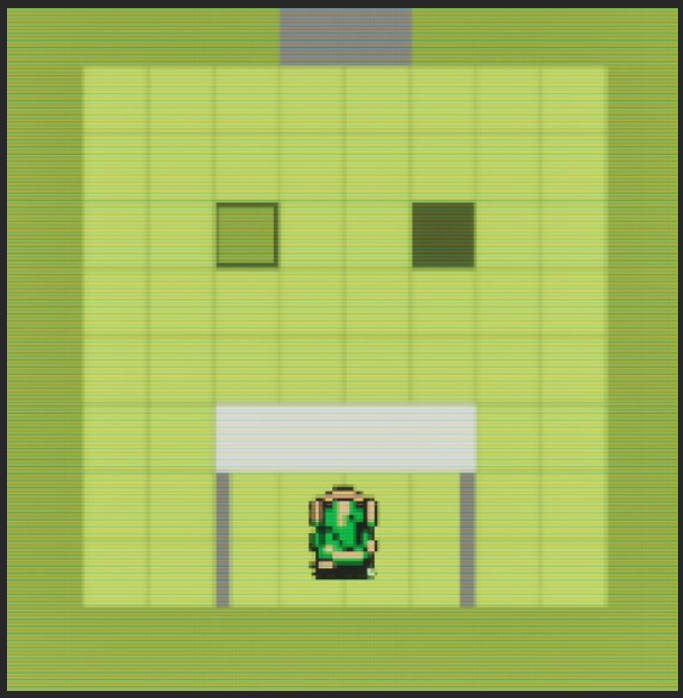
Moveable Objects

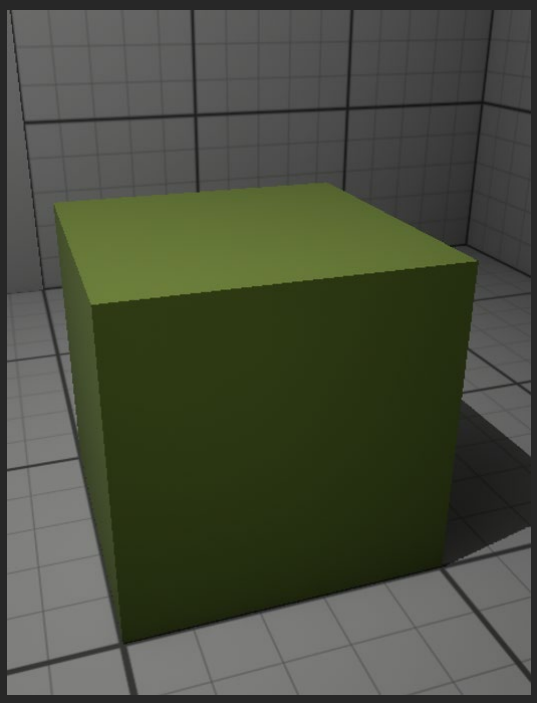
These objects are used by the player to be manoeuvred around the level, and can be very helpful for the following reasons:

- Can be used to on pressure plates.
- Can be used to fill in gaps to form bridges
- Can be used to help the first-person player up to higher places.



This feature was inspired by both Portal and The Legend of Zelda Links Awakening. The cubes in the Portal series can be used to activate buttons in simple puzzles. Zelda can push large objects around on a grid format. The university's own Mental Block *(University of Staffordshire, 2023)* which was made by the students of the 1UP scheme in 2023 was also very influential.



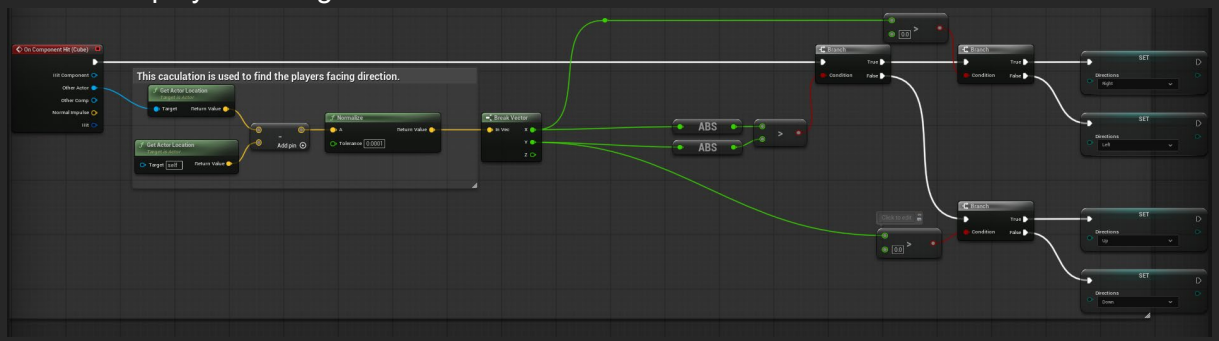


What design decisions were made?

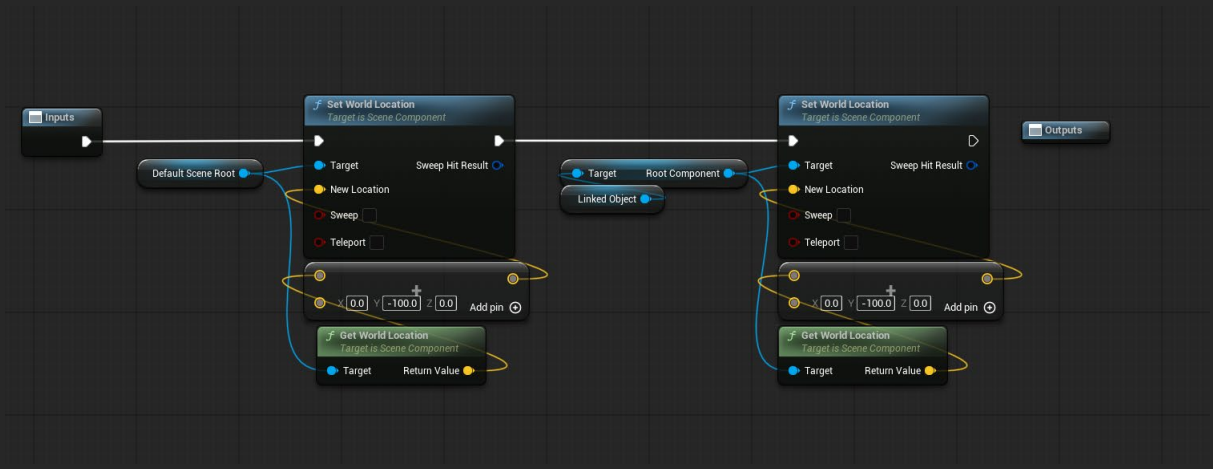
- Gaps – Allowing for interesting level design, while also adding a bit of depth to this feature.
- Usability in both environments – They help both players.

How were these made?

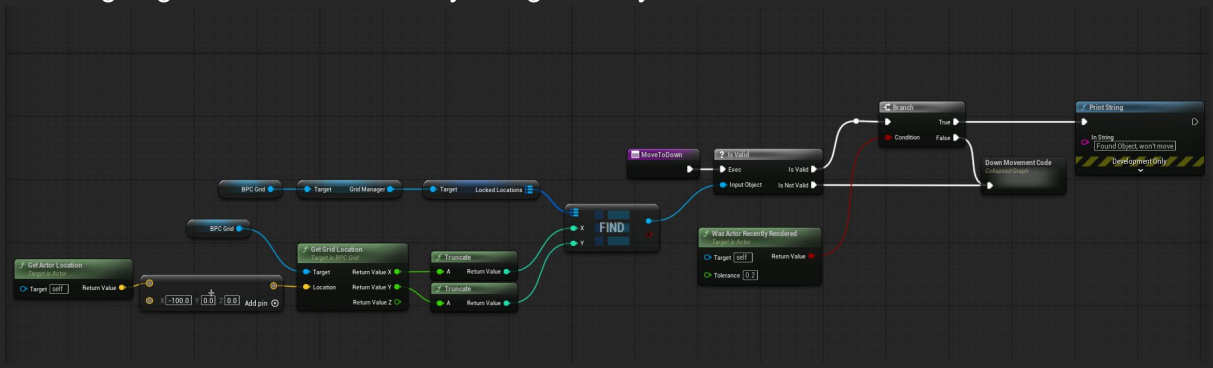
- When the player makes contact with the block, it will receive its location and calculates the players facing direction. It then uses this to dictate the future location of the block.



- Using 'Set World Location', the block moves to the next spot on the grid based on the players facing direction. It also moves the other block in the world with a public reference.

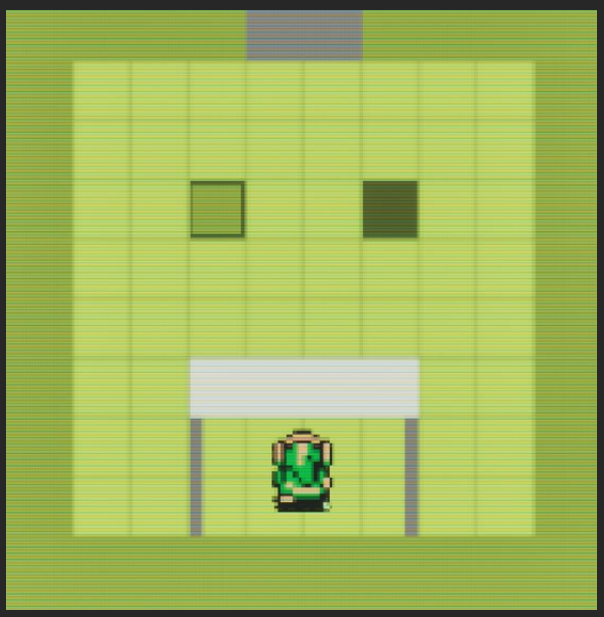


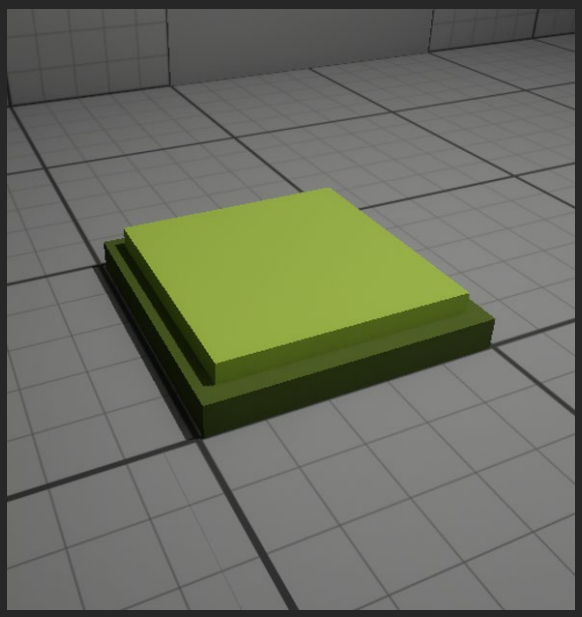
- Using the custom grid component, it will check to see if the position that the block is going to move to is currently being used by another actor. If it is, the block won't move.



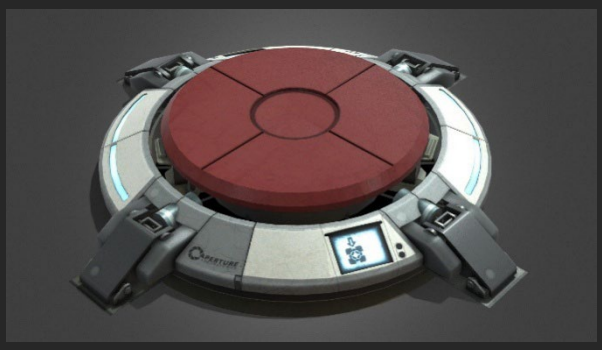
Pressure Plates

These plates are used to activate elements in the environment and are more dynamic, allowing for different functionality depending on the level.





This feature took inspiration from the Portal series and Viewfinder, to activate elements in the level.

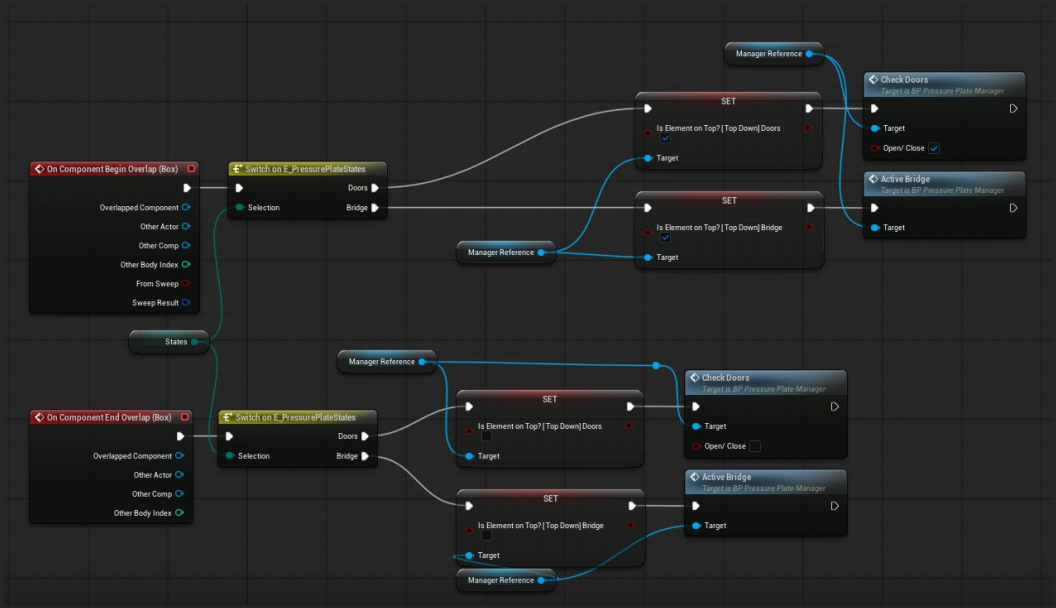


Design Decisions:

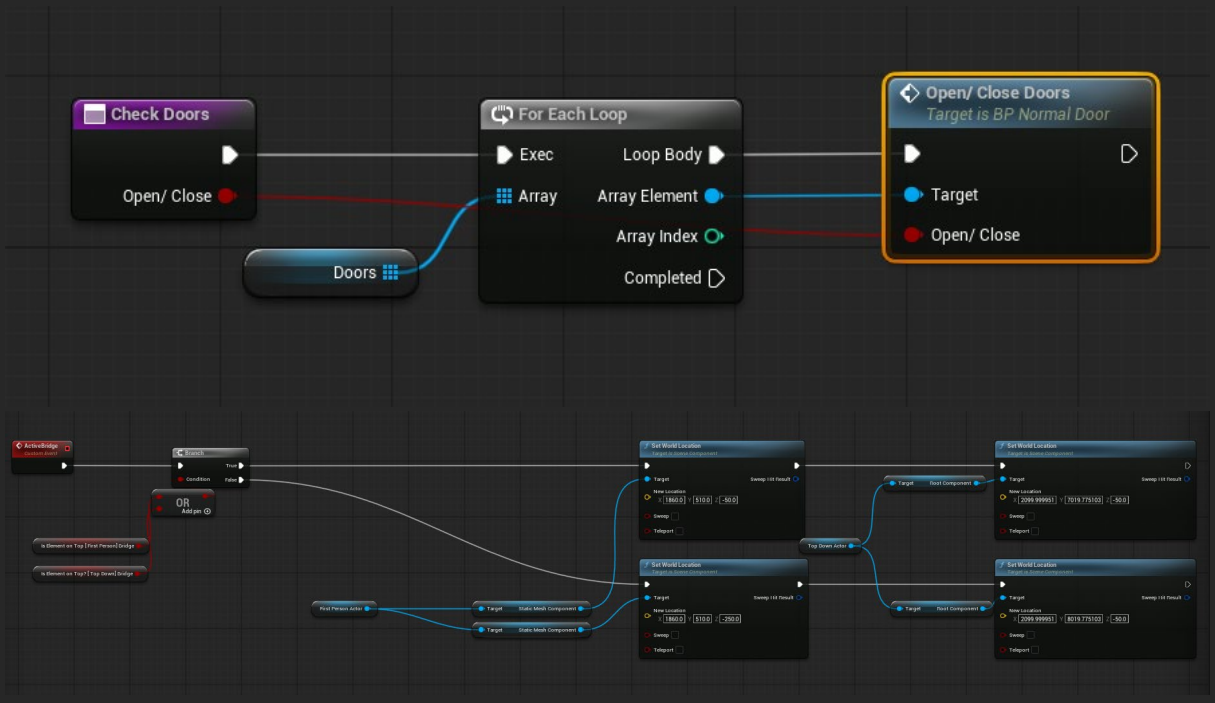
- These pressure plates can be activated by both players. It allows for player to see a preview of what the buttons functionality is.
- They also only need one player to activate. (i.e. If the first-person player is on the plate, then it will activate the function, and if the top-down player does the same it will not change anything).

How were these made?

- Using a public Enum, the functionality of the actor can change for what is desired for the level.



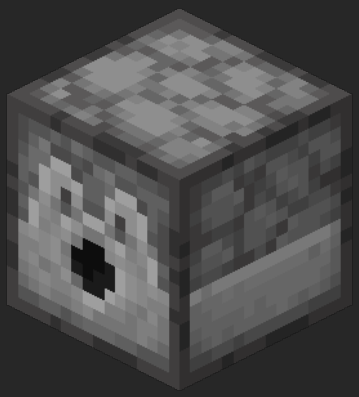
- Pressure Plate Manager was used to improve the functionality. This helps with handling multiple instances of the actor and holds multiple variables.



Projectile Launcher

The Projectile Launcher will fire small objects in a single direction, that can be dictated by the designer. Their purpose is to be pared with targets in the levels, to active elements in the world.

The inspiration stems from both Portal and Minecraft *(Mojang Studios, 2011)*. Portal's energy ball launcher is used in a similar way with the direction of the ball changing with the player's input. In Minecraft they have used the dispenser block which is a lot more dynamic but serves a similar purpose.

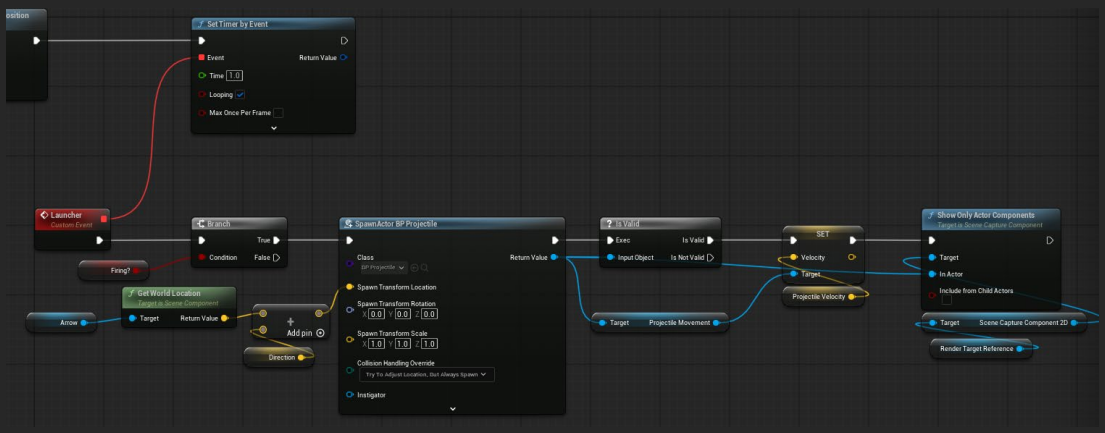


Design Decisions:

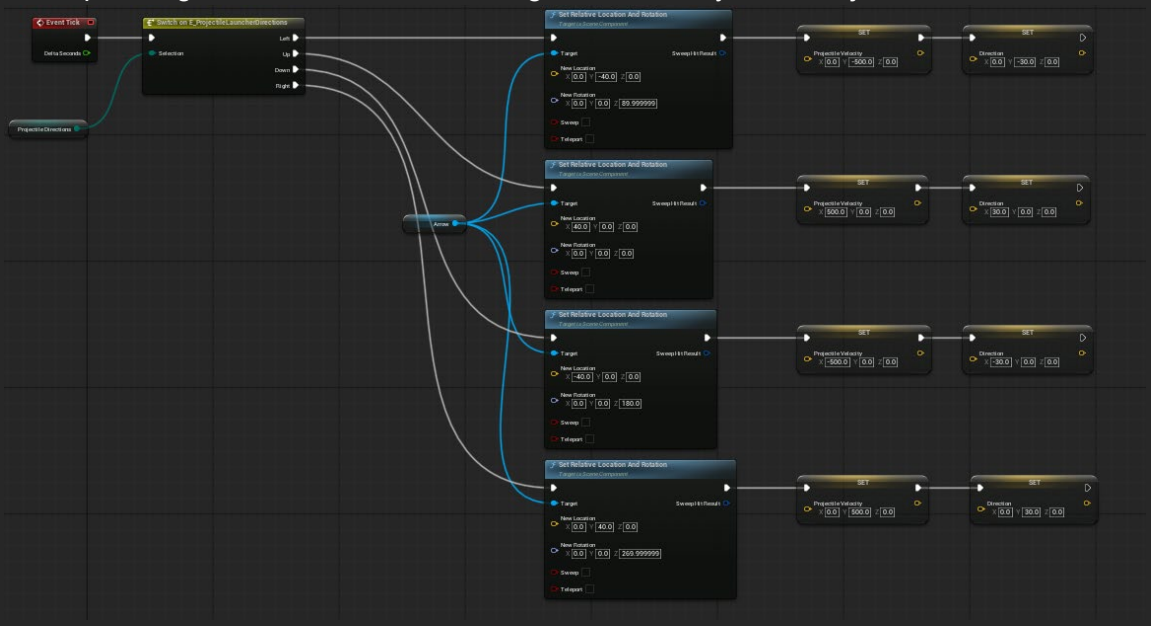
- This actor only exists within the top-down view, this improves the cognitive load.
- The player cannot be hurt by the actor and is mainly used to for the activation of elements in the world.

How were these made?

- A loop was used to spawn the projectiles from a certain point. Furthermore, to make sure that they are visible to the render target, they were added to the show actor array.



- Using a Enum, the designer can change the direction of the of the projectiles spawning location, which also changes the velocity of the object.



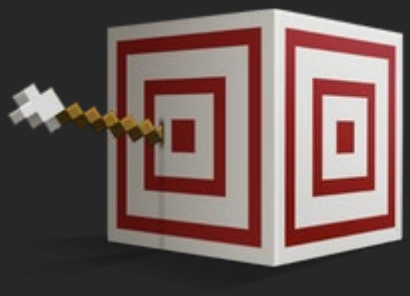
- The projectile destroys themselves upon contact, and if this is not the case it will destroy itself after 5 seconds, to prevent any leaks.

Targets

Targets are connected to the projectile launcher but share similarities with the pressure plates. They have different functionality based on what the designer needs. However, there are two different activation types.

- Once – Once contact has been made it activates the functionality and won't be activated again.
- Looping – Must keep hitting the Target for the functionality to be activated.

I took inspiration from the Target block from Minecraft, which can act in a very similar to this feature.

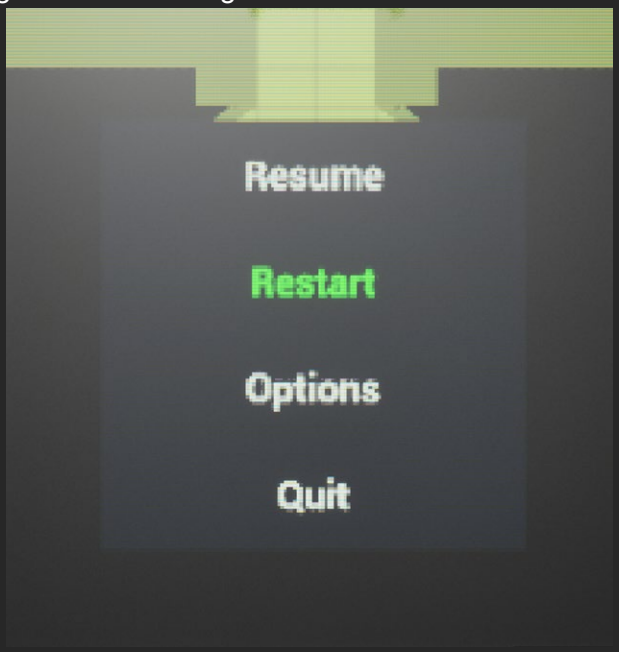


How was this made?

- Using an Enum, a designer can change the activation type as well as what it is activating in the world.

UI

The alterative UI widgets have been used to create both the pause and main menu. These have a mix of both diegetic and non-diegetic elements.



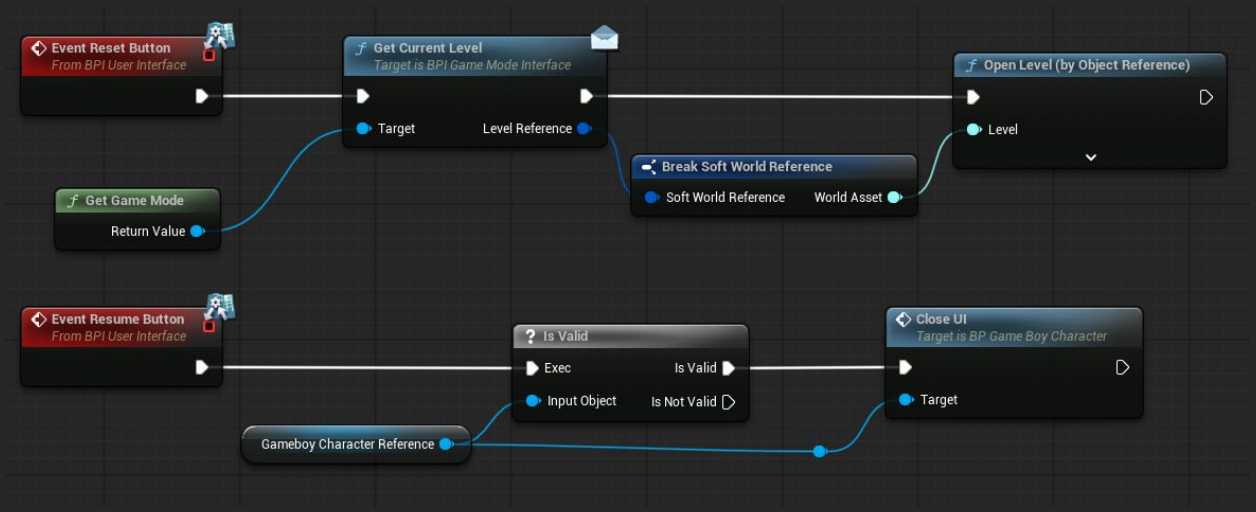
FYP | Pause Menu Showcase

Design Considerations:

- Since the UI is diegetic, it was positioned in front of the player's screen.

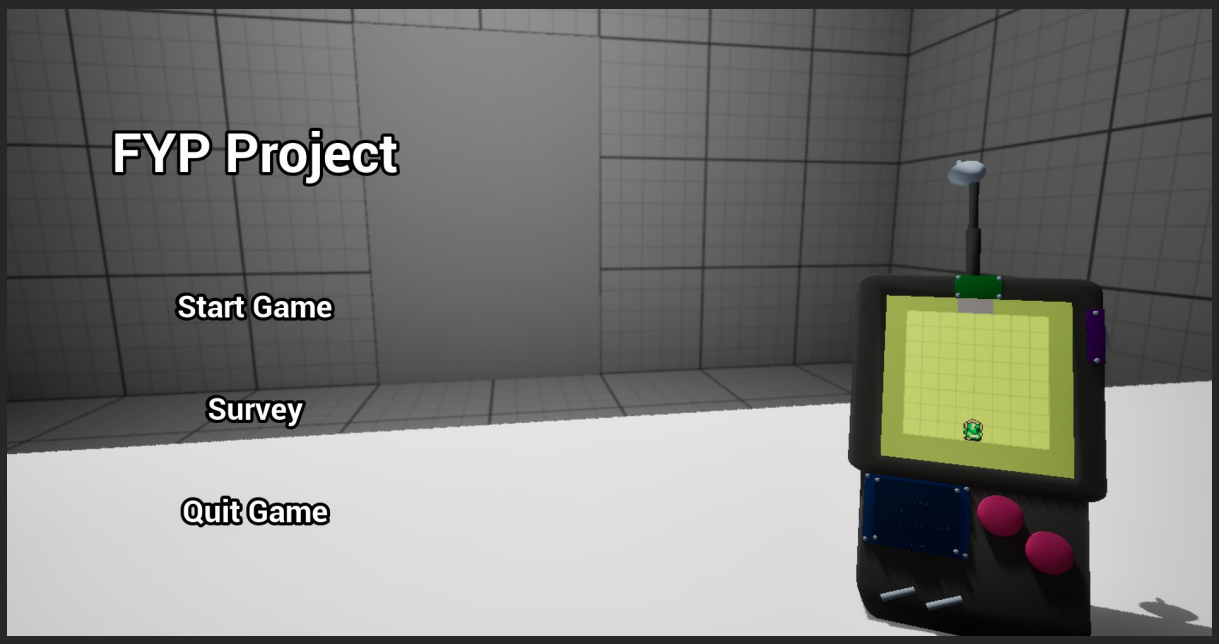
How was it made?

- Unreal's diegetic interfaces cannot be interacted with if it's far away from the player and since the UI is underneath the render target, it means that communication would be a little different.
  Interfaces were used as the main form of communication between player and widget.



Main Menu

The main menu was designed to be easily understood, not to overload the user and to launch the player straight into the action. Additionally, it clearly showcases the core mechanic of the game. In the future this could change but for the moment, at this stage of the project, it is sufficient.



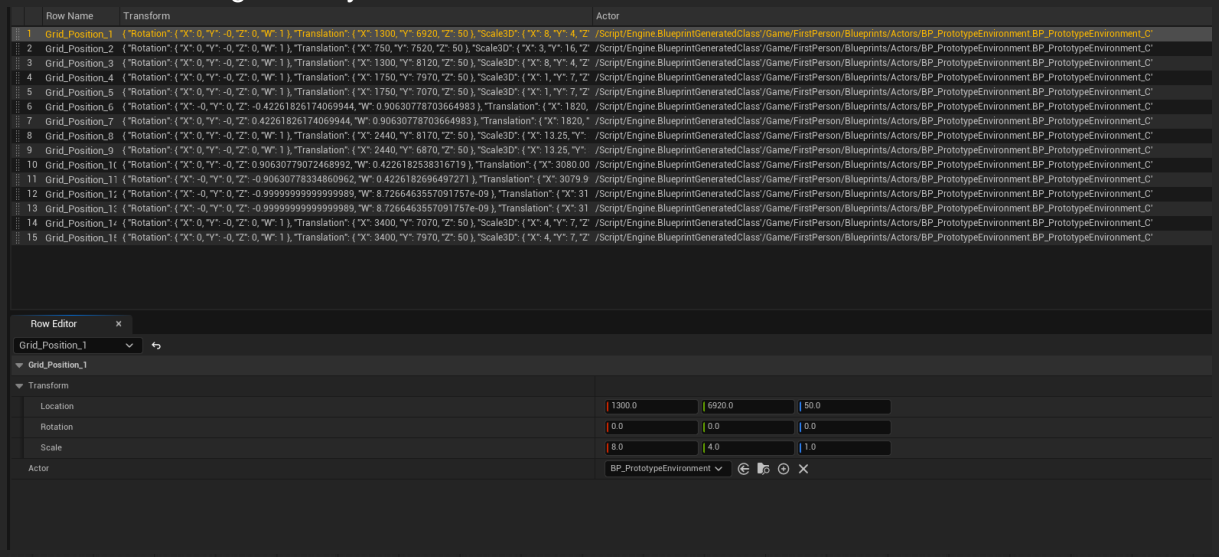
Gameplay Systems

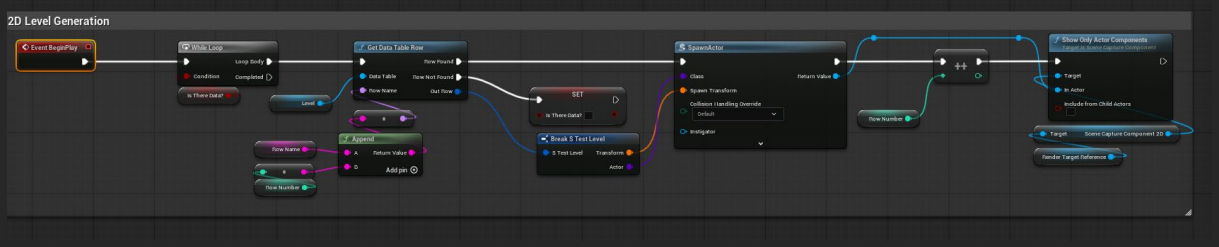
To make sure the development of the project went smoothly, it was necessary to create a couple of system around the project in order to help with development.

Level Generation System

To help with level's, a tool was created that generated the meshes for the top-down environment at runtime. This assisted with creating levels and reduced strain on the project.

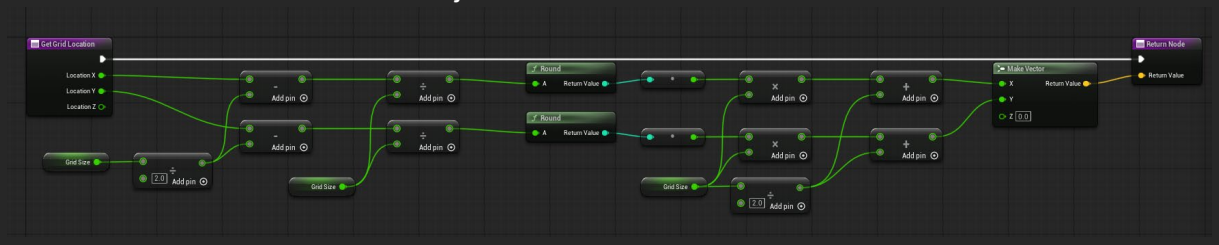
How was it made?
Using a Data table, the class for the environment assets was stored, along with a transform variable. Furthermore in the GameMode, a code runs that loops through the entries in the table until it cannot find another row. If it then finds an entry, it will spawn the actor and add it to the renders targets array list.





Grid System

The grid system was made specifically for the moveable objects, but it also helps with other features like the colour switching mechanic. The Grid component creates an invisible grid and uses custom function. Designers can convert an actor's location in the world and view the location on the grid.
By using a map to hold the grid location and actor data, grid locations are stored, locked and communicated to the moveable object.



Death System

The death system was made to create an easy method to respawn the player. Instead of the player being destroyed, the player will be teleported to the spawn location of the level. This should help with garbage collection.

Design Considerations

- The level does not reset, as this would frustrate players.
- If the top-down character dies, then it will not teleport the first-person player, and vica-versa.

Levels

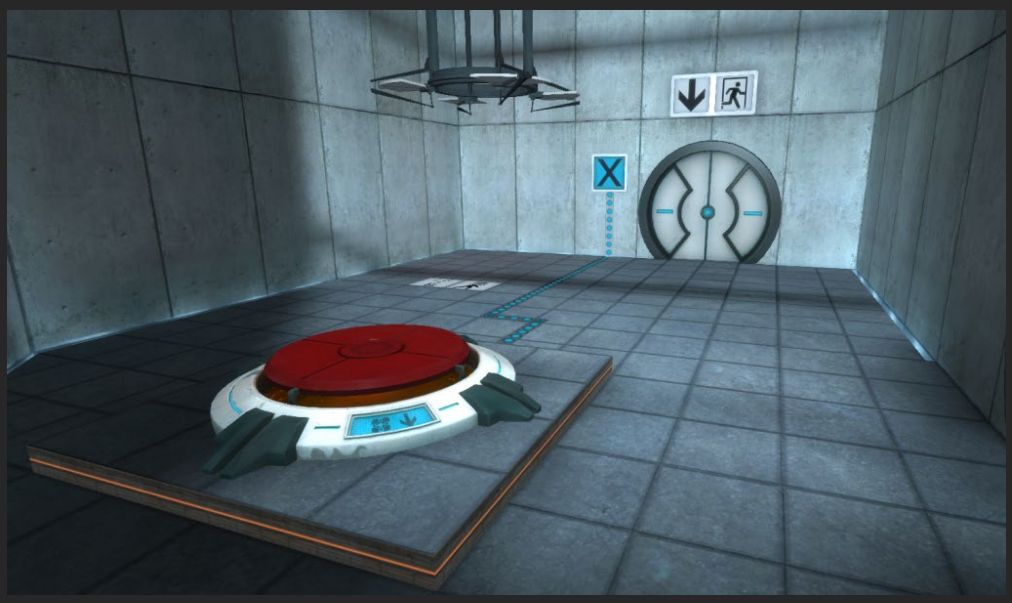
The levels within my project, use core puzzle design methods. Focusing on the progression and teaching of mechanics to the player, allowing for the last levels to ramp up the difficult accordingly.

Mark Brown of the ‘Game Makers Toolkit’ (Mark Brown, 2018) discussed 5 methods that help make puzzles fun and interesting, these were:

- The **Catch** - A logical contradiction
- The **Revelation** - A discovery
- The **Assumption** - A Trick
- The **Presentation** - Communication
- The **Curve** - Balancing

The Catch

The catch is "*where two things are seemingly in direct conflict with one another.*" *(Mark Brown, 2018) An* example of this method in action is in a simple pressure plate. A pressure plate opens the door when you stand on it, but if you step off the plate the door closes. In Mark Brown's video he notes that most puzzles use the catch to get players to think outside the box, to imagine another option.



The Revelation

Revelations signify that "eureka" moment when player solves a puzzle. By using the ‘catch’ to make the player think outside the box, it can be used to create a revelation for the user. Revelations allow the designer to use this problem somewhere else in a larger puzzle. Using this method, we can teach the player more about the mechanics they use to solve puzzles.

The Talos Principle *(Croteam, 2014)* have a great example of this, that was shown in Mark Browns video.

However, the main goal of all the levels is to teach the player, slowly improving their knowledge, so when the player reaches further levels in the future, it is possible for them to complete a complex idea, while also allowing designers to disrupt what they have learnt, in order to teach them something new.

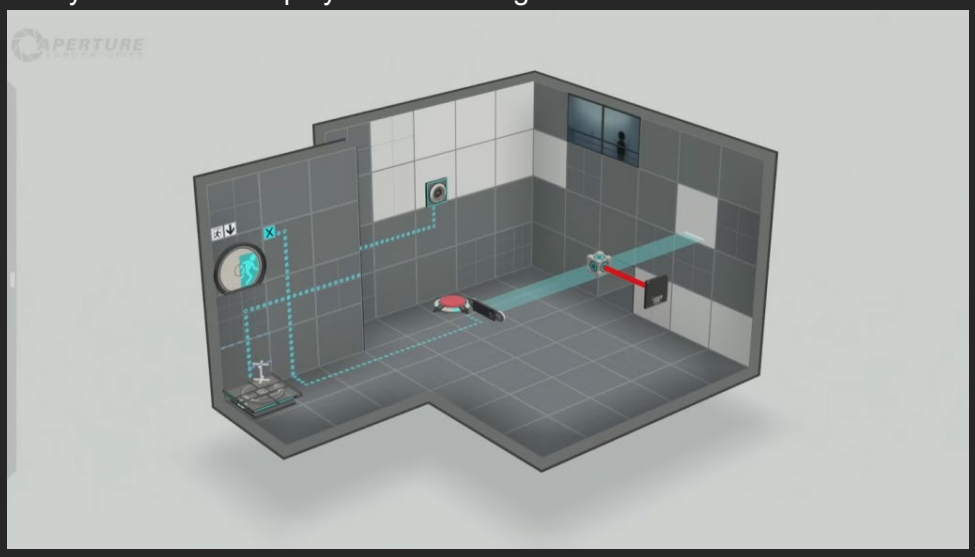
The Assumption

Now that the player has a learned a new element about how the game works, we as designers can now use this to our advantage. Using the players knowledge of the game we can design a later puzzle to go against what they already know. You could even design the puzzle to show the element they know first, and then the second problem will use the assumption.

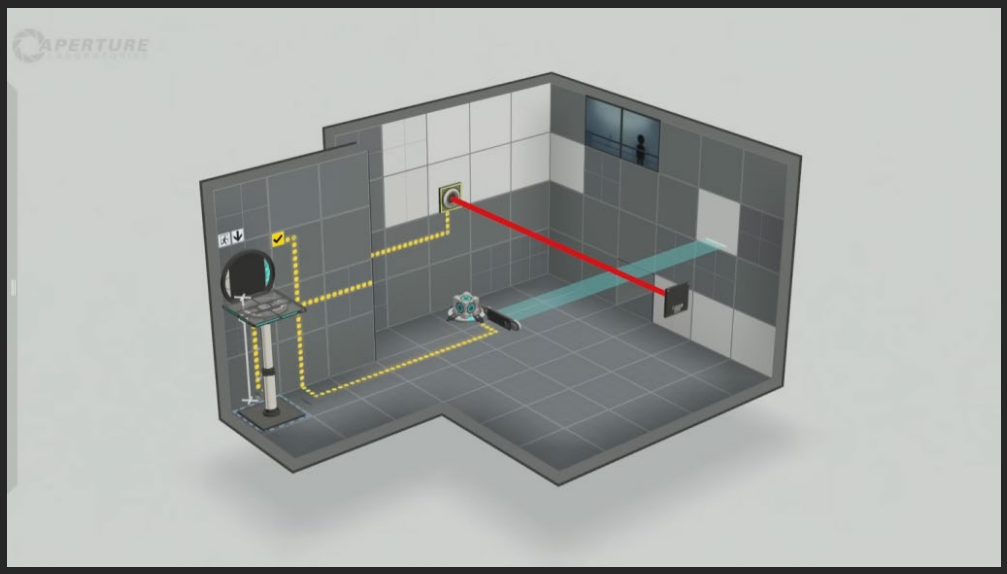
This does two things, one it allows the designer to give the player a new challenge, without introducing new features and two it allows the player to get a good starting point on the puzzle. This allows the player to feel confident that they know how to solve the problem, instead of becoming overwhelmed by a multitude of new features. Mark Brown uses the Lara Croft GO *(Square Enix Montreal, 2015)* to showcase this method in full.

The Presentation

Communicating is an important aspect of game design, as we must give info to the player without their awareness. This is what presentation is for puzzle design, the way you communicate your level to the player could change how difficult it is to find the solution.



Portal 2 *(Valve, 2011)* has a puzzle, where you must get a block from this energy bridge and then place in on a button. By removing the cube, the laser opens the door. Placing the cube on the button makes the platform by the door move up. However, this activates the ‘catch’, as you can't place the cube and get to the platform at the same time.



The energy bridge can be extended using portals. We can position the cube above the button, by extending the energy bridge using portals. You can go to the platform and remove your portal extending the bridge, this will make the cube fall on the button, making the platform you're on will elevate. The laser will now be able to hit its target and open the door for you to complete the level.

The way the portal developers presented this level, allowed players to find the 'catch' and then access the revelation. The way the level is positioned doesn't give the solution away, the energy bridge is not over the button but by the side, if it was over the button players could solve the problem too quickly. However, they also gave hits and clues to help, for example how the cube starts by blocking the laser, showcasing that it could be used later.

The Turing Test *(Bulkhead Interactive, 2016)* has a very similar puzzle to Portal 2, but the presentation makes this puzzle too simple. Since their energy platform is already over the button it's not hard to figure out that you need to turn off the energy bridge so the button can open the door.
Good presentation creates fun and challenging puzzles, while bad creates simple puzzles.

The Curve

Finally, the balancing on puzzle games are incredibly important. If Portal's levels showed up out of order, the game would become unbalanced and unenjoyable. Since the players learn something new in each level, the balance needs to be a smooth curve, making the plater feel they are improving at every level.

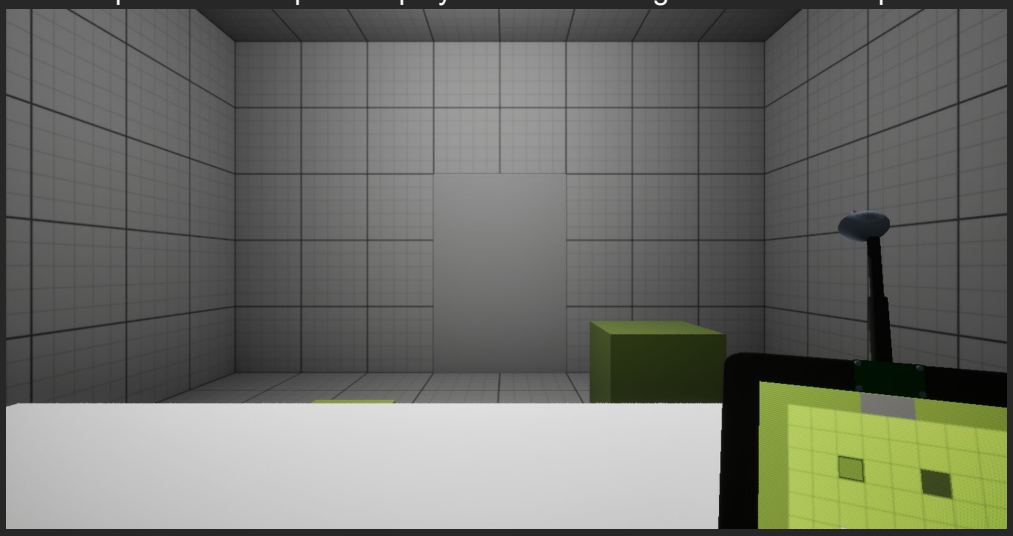
Level 1

This level served as an introduction to the games core mechanic, whilst also showcasing other mechanics that will be further explored in the upcoming levels.

In this level, the 'catch' technique is used in a similar fashion to Portal's first level. The player needs to get over a massive gap, the only way for the gap to be filled is to be on the pressure plate but only one player can be on it. This should create some conflict for the player.

What did players learn?

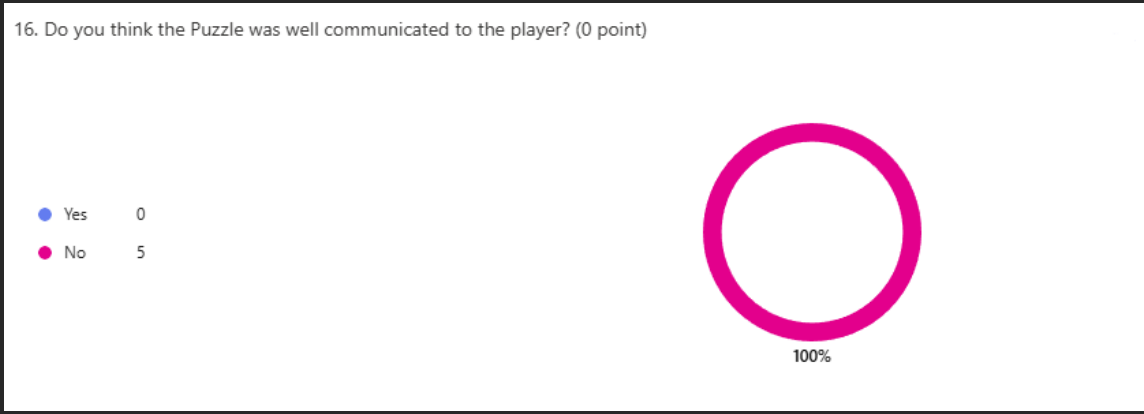
- The Gameboy can affect elements in the real world.
- The Gameboy Character can go underneath objects that have a gap in the real world.
- The Gameboy Character can move blocks and can push them onto pressure plates.
- Pressure plates can be activated by both players.
- Pressure plates can have multiple functions.
- The first person and top-down player must work together to solve a puzzle.

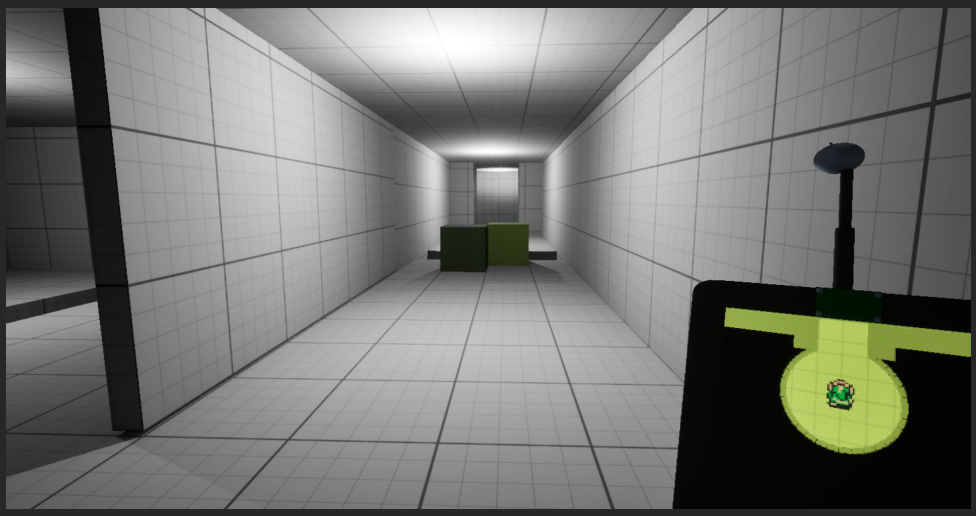


FYP | Level 1 Showcase

Level 2

The second level is used to teach the player a new feature. From research conducted, this showed how important it was to introduce this level. Previously players struggled with a later puzzle due to a lack of communication.





FYP | Level 2 Showcase

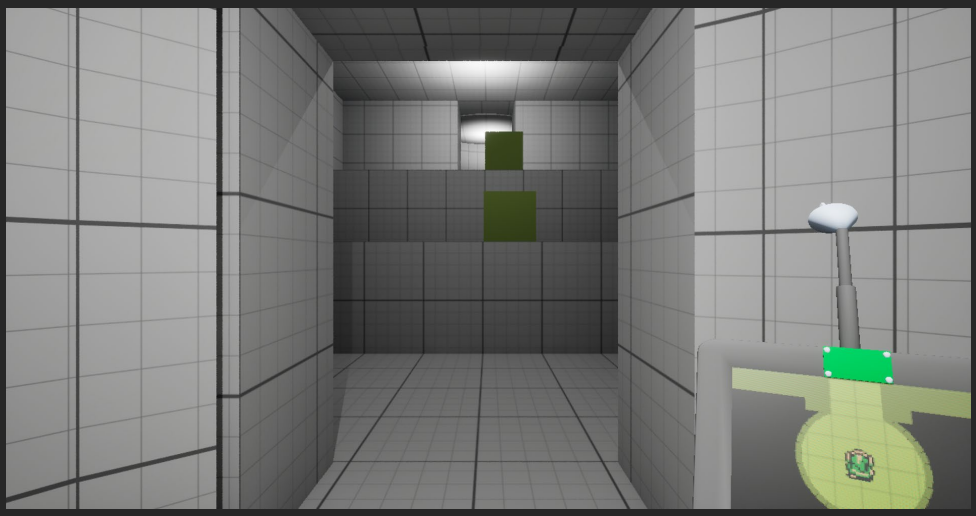
This level serves to solve this issue, by teaching that moveable objects can fill in gaps or holes, allowing for a bridge to be made. This allows the player to find a revelation, giving the user a moment to think outside the box.

Level 3

Level 3 teaches the player about perspective, and that the top-down player doesn't have the elevation that the first player must deal with. This level also uses the revelation method to build on what was learnt before.

What does the player learn?

- That the top-down player doesn't have any elevation changes.
- Blocks within the first-person environment have gravity.



FYP | Level 3 Showcase

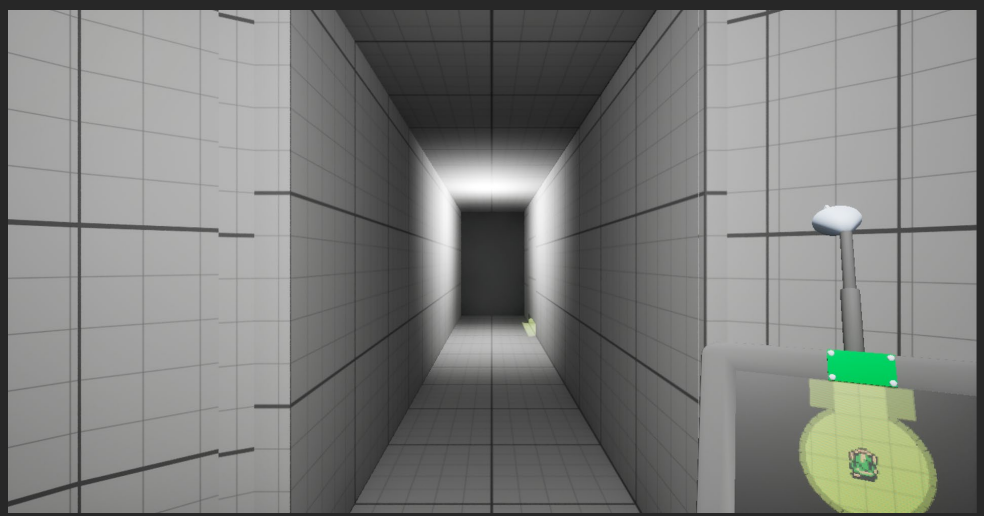
These 3 levels also help form a pyramid structure, which uses a lot of smaller and easier puzzles. By solving them, it will help the player complete a difficult puzzle later. *(Jesse Schell, 2019)*

Level 4

Level 4 is a mix between the assumption and the revelation methods. It takes what the player understands and shifts it on its head. The player runs into a corridor with no way to exit, just a light shining through a wall. This is supposed to hint that something is wrong.

What does the player learn?

- Hidden pathways exist.



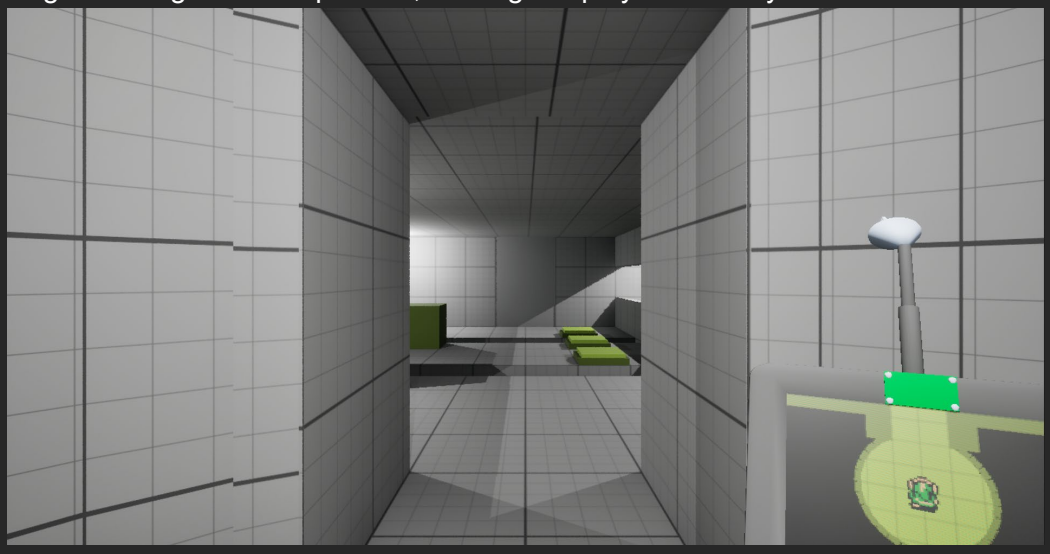
FYP | Level 4 Showcase

Level 5

Level 5 is the final test for the current set of levels. It pushes what players have learnt to the maximum, using almost everything from the other levels whilst introducing other level elements like the projectile launcher.

During testing, this was the level players had the most difficulty with, finding alternative ways to solve the puzzle. The introduction of level 2 really changed the results of the second playtest, with most players finding the right way to solve the puzzle. However, with some still using the alterative, which is ideally needed.

Games like Portal will have multiple ways to solve the puzzle however there is always a desired way to complete it. This makes the player feel like they broke the game or did something the designer didn't plan for, making the player feel very smart.

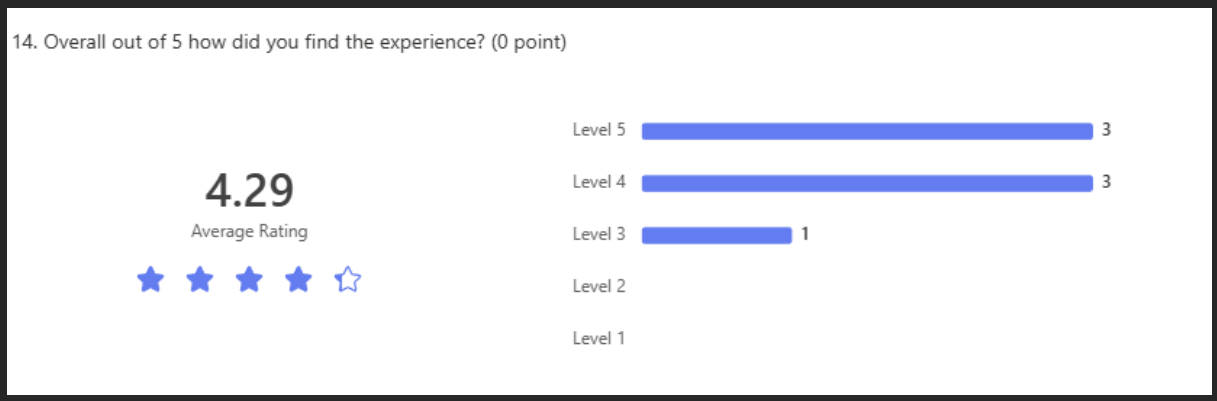


FYP | Level 5 Showcase

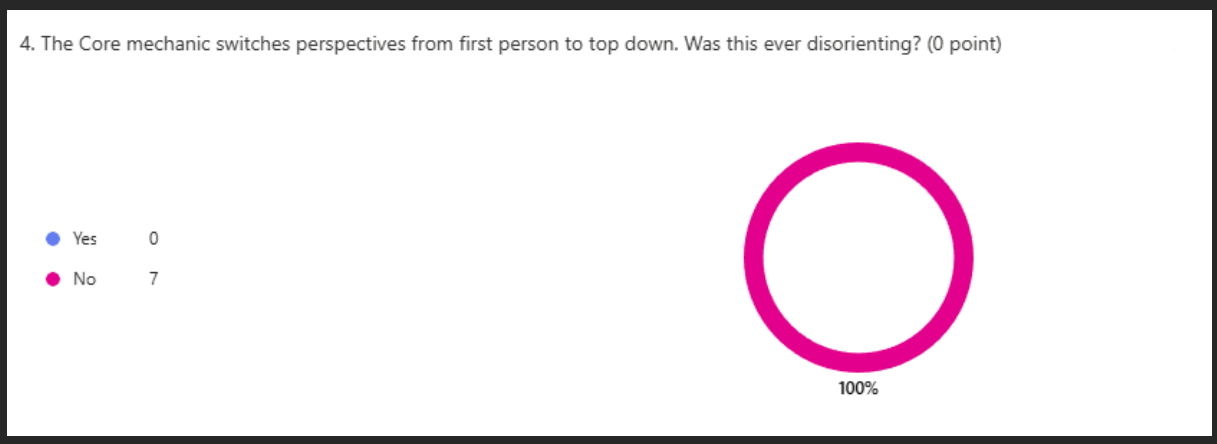
With all these levels, the ultimate goal was to achieve a balanced curve. This ensured that the difficulty naturally ramped up with the players own skills. My playtest showcased the progression of player learning, and this evidenced that the goal was completed.

Evaluation

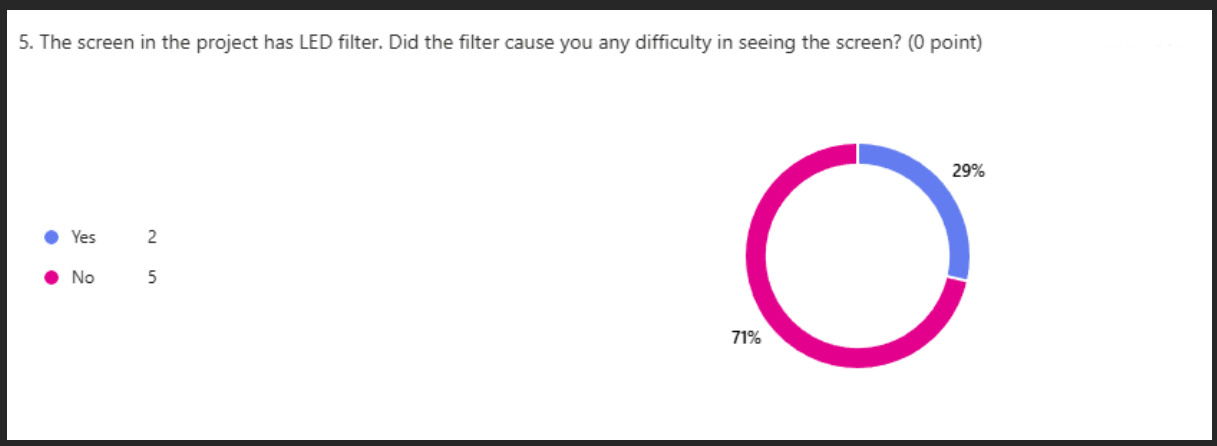
The research conducted over the last month has proven that the application of the interface and the puzzle design was well received by the testers. Comments such as "*This is incredibly impressive and is a really enjoyable puzzle game. I could see myself playing this non-stop if expanded with the same trajectory as it is headed. Well done!"* and *"I think you've done a great job on this, I've really enjoyed the game. It's very easy to understand and it's innovative at the same time. Well done!"* were used along with other comments.



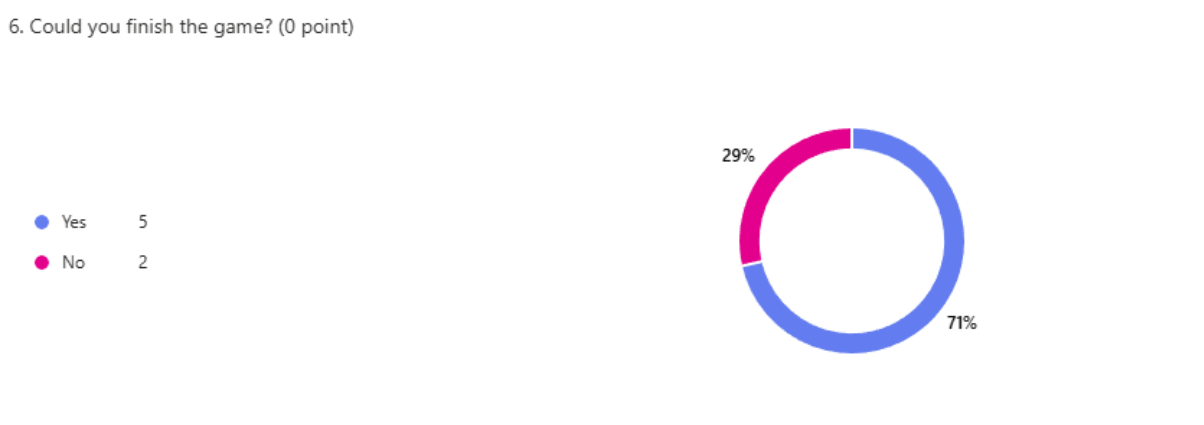
It is great to see the enjoyment the play testers had for the project, especially at this stage in development and shows the potential if the progress was continued.
Play testers responded well to the core feature and the interface didn't disorientate users. This was a concern since diegetic UI can cause players to become confused if not implemented well.



However, some results showcase the changes the project needs to make in the future, with the LED filter being one of them. Right now, most players do not mind the filter however 29% of players said they did. This indicates that a setting needs be included to help those users. In the future, an accessibility option to turn this feature off will be added.

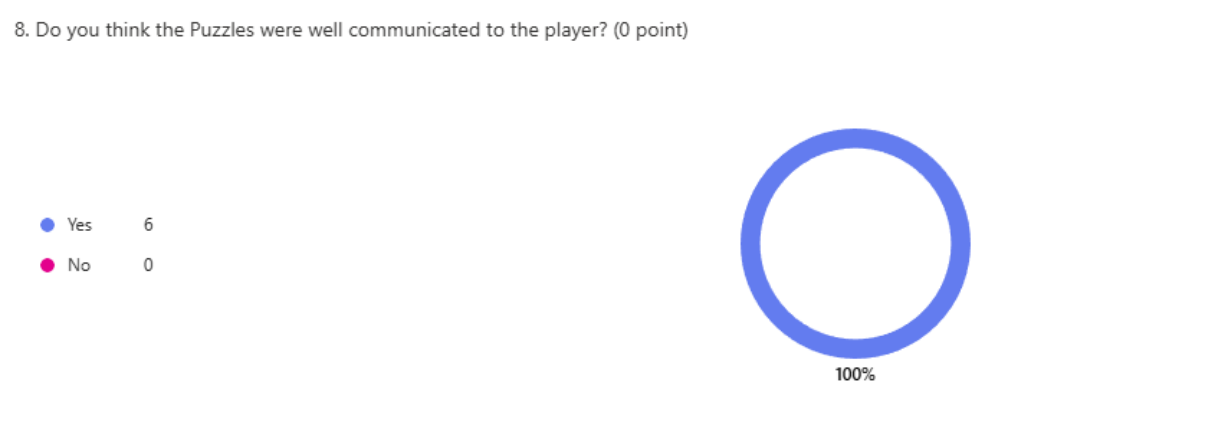


While watching play testers I noticed that the first level confused players and caused them to stop playing. This is because the information was poorly communicated and lead to misinformation. Before the submission this will be changed.



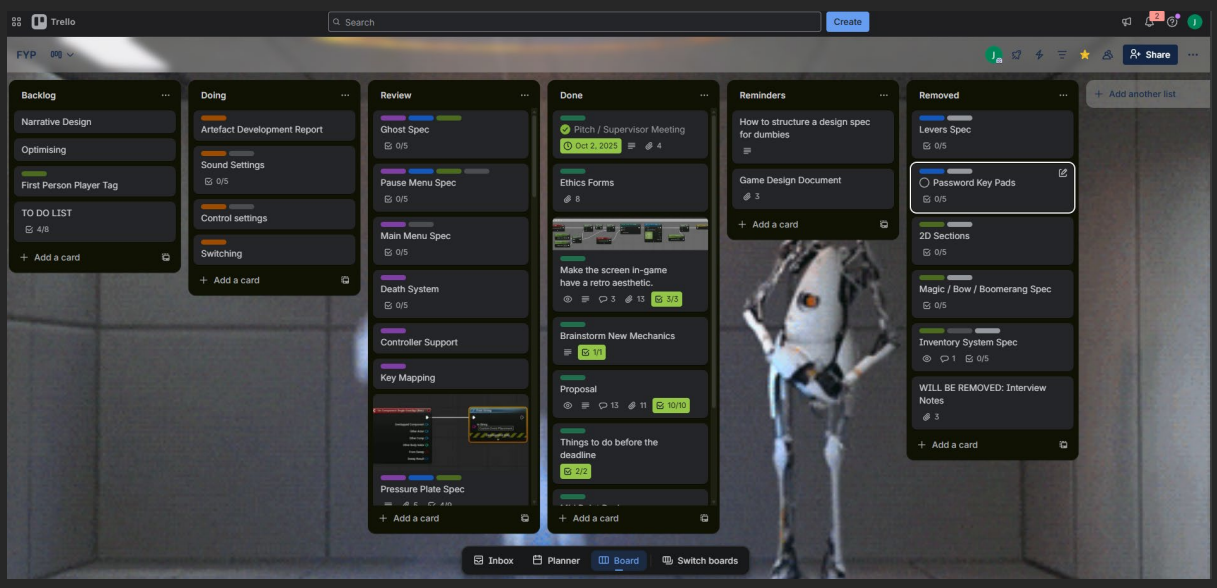
The final point I wanted to make is how the players adapted and learnt during their experience. A puzzle game must be able to teach the user information that will be needed further on in the game, this will make later levels more satisfying in return. Play tests have shown that my users were able to not only figure out the puzzle, but carried the information they learnt into a more difficult level.

When you go through the levels, something new is introduced each time, then one big test that tests your knowledge as a whole. [1] Learning how to switch characters and interaction [2] Interacting with boxes how they affect the player's environment [3] Using boxes to elevate the player while the pixelated player is one even ground. [4] Making a complex puzzle where you have to find a way to press 3 buttons with 2 players. Using one of the boxes!
you can push blocks into holes, fps can see perspective and depth where tps can't but can see a much wider view. fps can jump and tps can push
Puzzle 1, sprite can trigger pressure plates as well as the player, also sprite has weird unexplained rules with barriers and can push blocks Puzzle 2, blocks can cover gaps, player can jump over 2 block gaps Puzzle 3, sprite disregards elevation, player can jump one block high Puzzle 4, Sprite can only push 1x1 blocks Puzzle 5, nothing
I learned what the mechanics do. Intuitively, I noticed that there were platforms that needed to be activated with blocks.
The mechanics were self-taught very well, with each puzzle expanding on previous mechanics



I am incredibly proud of this project. It has encouraged my learning and taught me new skills while also progressing previous skills I had already learnt in the last couple of years. I was able to develop a better understanding of solo/indie development whilst also gaining a real passion for this area. It has enabled me to learn more about diegetic interfaces, a subject that I had been keen to investigate. I know how to use and implement diegetic interfaces as a result. This has driven an ambition to continue to learn more about this subject and continue to work on this project after my submission and GradX. It's an idea that has so much potential and I want to progress the game further.

Production Evaluation



This project utilised an agile methodology to my workload and production using Trello to accomplish this. My Trello Board used 4 lists to hold my backlog, the tasks I am currently working on, tasks that have been completed but need to be reviewed and completed tasks.

While this method did help me during the being of the project, I noticed that the review section rarely grew smaller. My intention had been to wait until the design spec was written before moving to the completed section. Although this could have been better utilised, this method works better in a team setting as this didn't lend itself well to the agile production style. Going forward I would change this.

References

- Angelika (2025) *The Art of Game UI - Good VS Boring*, *YouTube*. Available at: https://www.youtube.com/watch?v=dL152bMdK0k (Accessed: 19 February 2026).

- Larian Studios (2023) *Baldur's Gate 3* [Video Game], Larian Studios. Available at: https://baldursgate3.game/ (Accessed: 19 February 2026).

- Blizzard Entertainment (2004) *World of Warcraft* [Video Game], Blizzard Entertainment. Available at: https://worldofwarcraft.blizzard.com/en-gb/ (Accessed: 19 February 2026).

- Jagex (2001) *Runescape* [Video Game], Jagex. Available at: https://play.runescape.com/ (Accessed: 19 February 2026).

- Treyarch (2012) *Call of Duty Black Ops 2* [Video Game], Activision. Available at: https://store.steampowered.com/app/202970/Call_of_Duty_Black_Ops_II/ (Accessed: 19 February 2026).

- Treyarch (2024) *Call of Duty Black Ops 6* [Video Game], Activision. Available at: https://www.activision.com/games/call-of-duty/call-of-duty-black-ops-6 (Accessed: 19 February 2026).

- Brown, M. (2021) *The Power of Video Game HUDs*, *YouTube*. Available at: https://www.youtube.com/watch?v=4Bv45aPMGyI (Accessed: 19 February 2026).

- NovemberHotel (2024) *The Importance of Good UI Design*, *YouTube*. Available at: https://www.youtube.com/watch?v=u-Ml53Y29cs (Accessed: 19 February 2026).

- Treyarch (2020) *Call of Duty Black Ops Cold War* [Video Game], Activision. Available at: https://store.steampowered.com/app/1985810/Call_of_Duty_Black_Ops_Cold_War/ (Accessed: 19 February 2026).

- Kalkuhl, F. (2025) *Diegetic interfaces*, *Diegetic Interfaces - by Felix Kalkuhl*. Available at: https://nativeui.substack.com/p/diegetic-interfaces (Accessed: 19 February 2026).

- Kinetic Games (2020) *Phasmophobia* [Video Game], Kinetic Games. Available at: https://store.steampowered.com/app/739630/Phasmophobia/ (Accessed: 19 February 2026).

- Supersonic Software (1998) *Circuit Breakers* [Video Game], Mindscape. Available at: N/A (Accessed: 19 February 2026).

- EA Redwood Shores (2008) *Dead Space* [Video Game], Electronic Arts. Available at: https://store.steampowered.com/app/17470/Dead_Space_2008/ (Accessed: 19 February 2026).

- Obsidian Entertainment (2010) *Fallout: New Vegas* [Video Game], Bethesda. Available at: https://store.steampowered.com/app/22380/Fallout_New_Vegas/ (Accessed: 19 February 2026).

- Nintendo EAD (1993) *The Legend of Zelda: Link's Awakening* [Video Game] Nintendo. Available at: N/A (Accessed: 19 February 2026).

- Dhashuml (2025) *How to properly capture image using render targets and free them and performance monitoring for rendertargets.*, *Epic Developer Community Forums*. Available at: https://forums.unrealengine.com/t/how-to-properly-capture-image-using-render-targets-and-free-them-and-performance-monitoring-for-rendertargets/2579921/2 (Accessed: 19 February 2026).

- Schell, J., Clark, N. and Fay, I. (2019) *Teaching puzzle design*, *GDC Vault*. Available at: https://gdcvault.com/play/1025845/Teaching-Puzzle (Accessed: 19 February 2026).

- Valve (2007) *Portal* [Video Game], Valve. Available at: https://store.steampowered.com/app/400/Portal/ (Accessed: 19 February 2026).

- Sad Owl Studios (2023) *Viewfinder* [Video Game], Thunderful Publishing. Available at: https://store.steampowered.com/app/1382070/Viewfinder/ (Accessed: 19 February 2026).

- Nintendo R&D2 (1998) *The Legend of Zelda: Links Awakening DX* [Video Game], Nintendo. Available at: N/A (Accessed: 19 February 2026).

- Nintendo (1998) *Gameboy Colour* [Console], Nintendo. Available at: N/A (Accessed: 19 February 2026).

- University of Staffordshire (2023) *MentalBlock - 2023 1UP Scheme Gameplay*, *YouTube*. Available at: https://youtu.be/RVK1WAmstb4?si=GFOzapjrbFd2egcE (Accessed: 19 February 2026).

- Mojang Studios (2011) *Minecraft* [Video Game], Mojang Studios. Available at: https://www.minecraft.net/en-us (Accessed: 19 February 2026).

- Brown, M. (2018) What Makes a Good Puzzle?, YouTube. Available at: https://www.youtube.com/watch?v=zsjC6fa_YBg (Accessed: 19 February 2026).

- Croteam (2014) *The Talos Principle* [Video Game], Devolver Digital. Available at: https://store.steampowered.com/app/257510/The_Talos_Principle/ (Accessed: 19 February 2026).

- Square Enix Montreal (2015) *Lara Croft Go* [Video Game], Square Enix. Available at: https://play.google.com/store/apps/details?id=com.squareenixmontreal.lcgo&hl=en-US&pli=1 (Accessed: 19 February 2026).

- Valve (2011) *Portal 2* [Video Game], Valve. Available at: https://store.steampowered.com/app/620/Portal_2/ (Accessed: 19 February 2026).

- Bulkhead Interactive (2016) *The Turing Test* [Video Game], Square Enix Collective. Available at: https://store.steampowered.com/app/499520/The_Turing_Test/ (Accessed: 19 February 2026).