The Sporadic Puzzles

a puzzle minigame system for AAA

Development Report

Break into the safe

Guide both lasers to the center

Optional Challenge

Complete the puzzle by using one block or less

MODULES

B Back

Reset

Move Position

A Select Position

Contents

Project Proposal

Pre-Production
- Planning
- Research
- Concept

Production
- Prototype
- Controls
- Visual Identity
- Tutorial
- User Testing

Final Artefact

Evaluation

Bibliography

Project Proposal

Project Proposal

Introduction

AAA Action-Adventure games often offer puzzle minigames.

- They can help to **break the pace** and bring some **variety** to the player experience.
- However, these are too often considered tedious by both action fans and puzzle fans alike.
- Too often, the necessity of being quick comes at the price of being trivial.

Can a puzzle minigame for AAA Action-adventure games be fun for everyone?

Project Proposal

Aims and Objectives

This project aims to create a **puzzle minigame system** for AAA action-adventure titles, that caters to puzzle fans and actions fans alike.

To achieve this, we are planning the following steps:

- Produce a thorough **research analysis** on puzzle games and AAA action-adventure minigames;
- Concept one (or more) puzzle minigame system;
- **Develop** the chosen concept on UE5;
- **Playtest** & **iterate** to determine if aims were met.

Deliverables

- A UE5 **artifact** for the puzzle minigame system, industry-matching in size and scope;
- **Research** paper;
- Design **Documentation.**

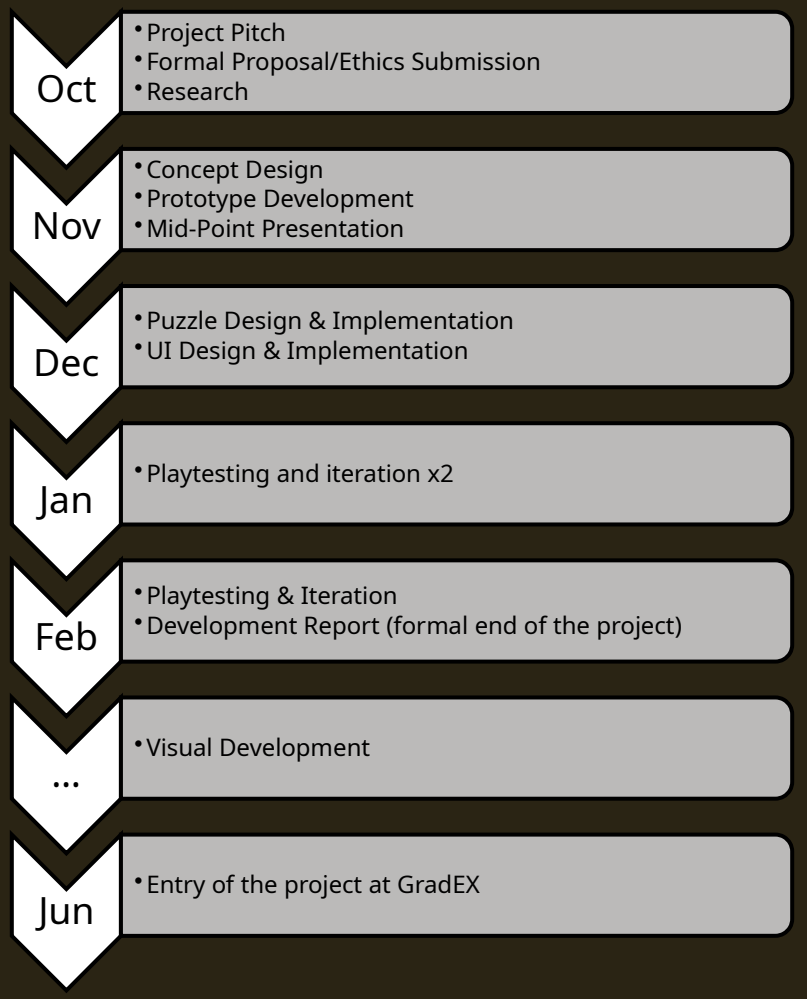
Pre-Production

Pre-Production

Planning

This project taking place as part of a final year project, formally running from October 2024 to February 2025.

More work is planned on the project past that date, to prepare the project for GradEX (June 2024).



Pre-Production

Research

This is a **research-first project**. Instead of using research to justify and existing idea, we wanted the idea to come from the research.

The result is a **9000+ word research paper**, which can be divided in 4 parts:

- **Defining Minigames & Puzzles:**
  - Understanding the boundaries of both minigames and puzzles;
  - Getting the right vocabulary to define classify and critic puzzles.

Raphael Colucci The Sporadic Puzzles Research

Defining Minigames & Puzzles

What is a Minigame?

A minigame can be broadly defined as a **short activity within a larger game**. Gameplay will substantially **differ** from the rest of the experience. In general, **controls** will change, and the player might gain a new **perspective**, or temporarily lose control of their player-character while they interact with the minigame.

A minigame is generally considered both **smaller** and **simpler** than the main gameplay experience, although these metrics can vary from game to game and are inherently subjective. In some case, the game itself can be a collection of minigames and interacting with them constitutes the primary activity of the player[1].

Minigames are sometimes part of the critical path and **mandatory** in order to progress through the story. They can also be completely **optional**, constituting a side activity separate from the main experience. Often, an **in-between** approach is used: the critical path will include a small number of mandatory interaction with said minigame, and making others optional.

When added as a side activity, minigames can take many different **forms**. They can require the player to practice either or a combination of:

- **Reflexes**
- **Memory**
- **Spatialisation**
- **Precision**
- **Rhythm**

Because they usually have their own controls and objectives, they will normally come with their own tutorial/onboarding systems.

Inside of a bigger game, they are usually **sprinkled throughout the course of the game** instead being available as a block, which constitutes its own set of design challenges.

What is a Puzzle?

A puzzle is defined as "something that is **difficult to solve, resolve or understand**: something that puzzles" (Merriam-Webster, 2024) or "something (such as a game or test) that is **designed to be challenging** to answer or solve" (Merriam-Webster, 2024).

In the context of games, it is correct that things like combat interactions or a platforming challenge can sometimes be referred to as "puzzles" by the designer. It is however not what most players would colloquially refer to when they use the term: people will generally think of "cognitive puzzles" specifically. When faced with a puzzle, a player will expect to be presented with a problem, along with a clear goal, but that will require reflection and experimentation to achieve.

[1] See the Wario-Ware series, and to some extent the Mario Party series.

4

Pre-Production

Research

- Analysing the **purpose** of minigames inside of a bigger project
  - Break the pace?
  - Offer rewards?
  - Teaching mechanics?

Purpose of a Minigame Inside of a Bigger Project

Different Gameplay to Break the Pace

Looking at modern open-world offerings, we can find a lot of side-activities to do outside of the main quest. It is important for this optional content to offer **more of the main gameplay features**. For instance, Marvel's Spiderman, the player should be able to do more combat or stealth challenges outside of the main story (and if possible, infinitely repeatable). We can however get to a point where players might want to do something else entirely.

Inside of a bigger project, such as AAA action-adventure, minigames allow for the player **to take a break from the action without taking a break from the game**. Going back-to-back with action set-pieces or combat challenges can lessen their impact, and the player will need some punctual breathing room in-between, or they might be pushed to close the game and do something else. In essence, it lets the player **interact with the game**, but **outside of its main gameplay draw**, or without advancing the story.

An Avenue for Rewards?

Rewards are normally involved in these side-activities but are not exclusive to minigames: **any side-activity will offer a level of reward**. These rewards might be XP, consumables, equipment, crafting materials, collectables… and can also include some story bits (i.e. a tape with a recorded dialogue) deepening the lore.

Side-activities must be **fun and engaging** for the player to take pleasure in partaking, but there needs to be another motivation in the place to want to interact, and more importantly keep interacting, with a type of activity. A designer could then modulate engagement with a feature by improving the rewards associated to it. It would however create a situation where players do what they feel like they HAVE to do instead of WANT to do.

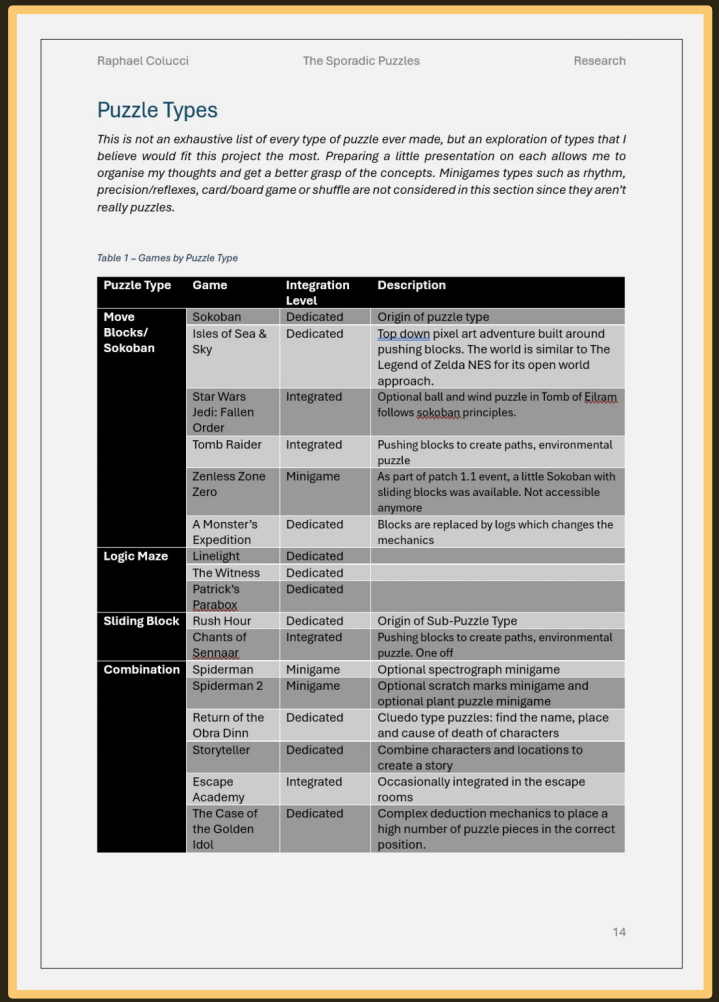
Figure 5 – Bonus Objectives in Marvel' Spiderman 2

Pre-Production

Research

- Exploring **puzzle types**
  - Sokoban;
  - Sliding Block;
  - Logic Maze/Line Drawing;
    - Nikoli logic puzzles;
  - Combination

For each of the puzzle types, we looked at the **main features, history, variation examples, controls and presentation.**



Puzzle Types

This is not an exhaustive list of every type of puzzle ever made, but an exploration of types that I believe would fit this project the most. Preparing a little presentation on each allows me to organise my thoughts and get a better grasp of the concepts. Minigames types such as rhythm, precision/reflexes, card/board game or shuffle are not considered in this section since they aren't really puzzles.

Table 1 – Games by Puzzle Type

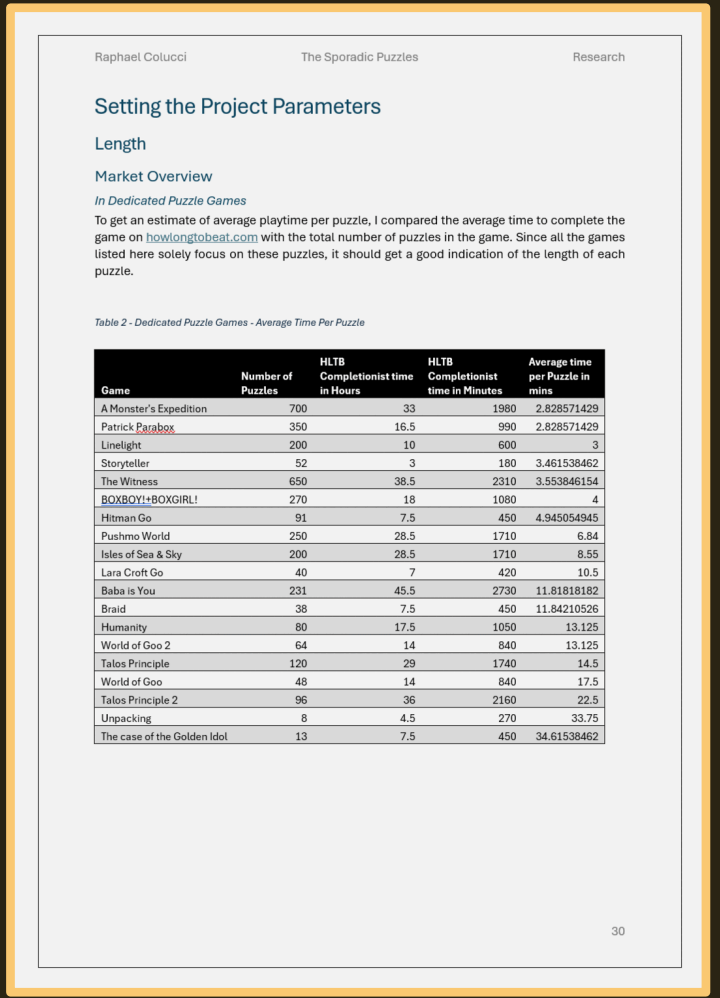
Puzzle Type	Game	Integration Level	Description
Move Blocks/ Sokoban	Sokoban	Dedicated	Origin of puzzle type
	Isles of Sea & Sky	Dedicated	Top down pixel art adventure built around pushing blocks. The world is similar to The Legend of Zelda NES for its open world approach.
	Star Wars Jedi: Fallen Order	Integrated	Optional ball and wind puzzle in Tomb of Eilram follows sokoban principles.
	Tomb Raider	Integrated	Pushing blocks to create paths, environmental puzzle
	Zenless Zone Zero	Minigame	As part of patch 1.1 event, a little Sokoban with sliding blocks was available. Not accessible anymore
	A Monster's Expedition	Dedicated	Blocks are replaced by logs which changes the mechanics
Logic Maze	Linelight	Dedicated	
	The Witness	Dedicated	
	Patrick's Parabox	Dedicated	
Sliding Block	Rush Hour	Dedicated	Origin of Sub-Puzzle Type
	Chants of Sennaar	Integrated	Pushing blocks to create paths, environmental puzzle. One off
Combination	Spiderman	Minigame	Optional spectrograph minigame
	Spiderman 2	Minigame	Optional scratch marks minigame and optional plant puzzle minigame
	Return of the Obra Dinn	Dedicated	Cluedo type puzzles: find the name, place and cause of death of characters
	Storyteller	Dedicated	Combine characters and locations to create a story
	Escape Academy	Integrated	Occasionally integrated in the escape rooms
	The Case of the Golden Idol	Dedicated	Complex deduction mechanics to place a high number of puzzle pieces in the correct position.

Pre-Production

Research

- Determining the **project parameters** on:
  - **Length**
  - **Amount of noise**
  - **Difficulty**
  - **Instructions/Onboarding**

Research Paper download link



Setting the Project Parameters

Length

Market Overview

In Dedicated Puzzle Games

To get an estimate of average playtime per puzzle, I compared the average time to complete the game on howlongtobeat.com with the total number of puzzles in the game. Since all the games listed here solely focus on these puzzles, it should get a good indication of the length of each puzzle.

Table 2 - Dedicated Puzzle Games - Average Time Per Puzzle

Game	Number of Puzzles	HLTB Completionist time in Hours	HLTB Completionist time in Minutes	Average time per Puzzle in mins
A Monster's Expedition	700	33	1980	2.828571429
Patrick Parabox	350	16.5	990	2.828571429
Linelight	200	10	600	3
Storyteller	52	3	180	3.461538462
The Witness	650	38.5	2310	3.553846154
BOXBOY!+BOXGIRL!	270	18	1080	4
Hitman Go	91	7.5	450	4.945054945
Pushmo World	250	28.5	1710	6.84
Isles of Sea & Sky	200	28.5	1710	8.55
Lara Croft Go	40	7	420	10.5
Baba is You	231	45.5	2730	11.81818182
Braid	38	7.5	450	11.84210526
Humanity	80	17.5	1050	13.125
World of Goo 2	64	14	840	13.125
Talos Principle	120	29	1740	14.5
World of Goo	48	14	840	17.5
Talos Principle 2	96	36	2160	22.5
Unpacking	8	4.5	270	33.75
The case of the Golden Idol	13	7.5	450	34.61538462

Pre-Production

Research

The initial research concluded with the following set of project parameters. Research determined that these conditions should ensure achieving the project's aims:

- **Puzzle Structure:**
  - Puzzles will have **1+ optional challenge** on the side of the screen.
  - Completing these challenges will get the player **more rewards**.
- **Length:**
  - The **main puzzle** aims to be solved in approx. **1 min**.
  - The same puzzle with **optional challenge** aims to be solved in approx. **3 min**.
- **Difficulty:**
  - **Main puzzle** should require **some limited effort** from the player. It is quickly rewarded.
  - **Optional challenge** forces the player to be **more intentional** in their puzzle solving and forces the player to **use the mechanic in a novel way**
  - The player can **choose** themselves **the level of challenge** for each level **by partaking** on optional challenges or not.
- **Noise:**
  - The puzzle should be **grounded in the game world** and not feel fully separated from the main gameplay. It will be instanced but it should be elegantly transitioned from main gameplay.
    - Option to use the world as background for puzzle.
  - If visual noise is added, it should **not detract from the puzzle** or make it harder to read in any way
  - **Distance between different elements** should be kept to a **minimum** to limit the realm of possibilities.
  - **The number of puzzle pieces should be of (or very close to) the necessary number** to solve the main puzzle.
- **Direction**:
  - **Instructions** will be **given in a sentence**: a few words must be enough to explain the puzzle goal. **Icons** and Pictograms can be used to reinforce communication.
  - **Failure** and **success** must be **clearly communicated** to the player with **puzzle animation** and **eye-catching text**.
  - **Controls** must be **simple** and always clearly **present on screen.**

Pre-Production

Concept

After the initial research, **two different concepts** were designed:

- LightPath
- **NEXUS**

It was not initially planned to concept two different systems, but since we were not fully satisfied with the first (LightPath), a second concept was developed while still staying on schedule.

For both concepts, the process consisted of creating **documentation decks** containing:

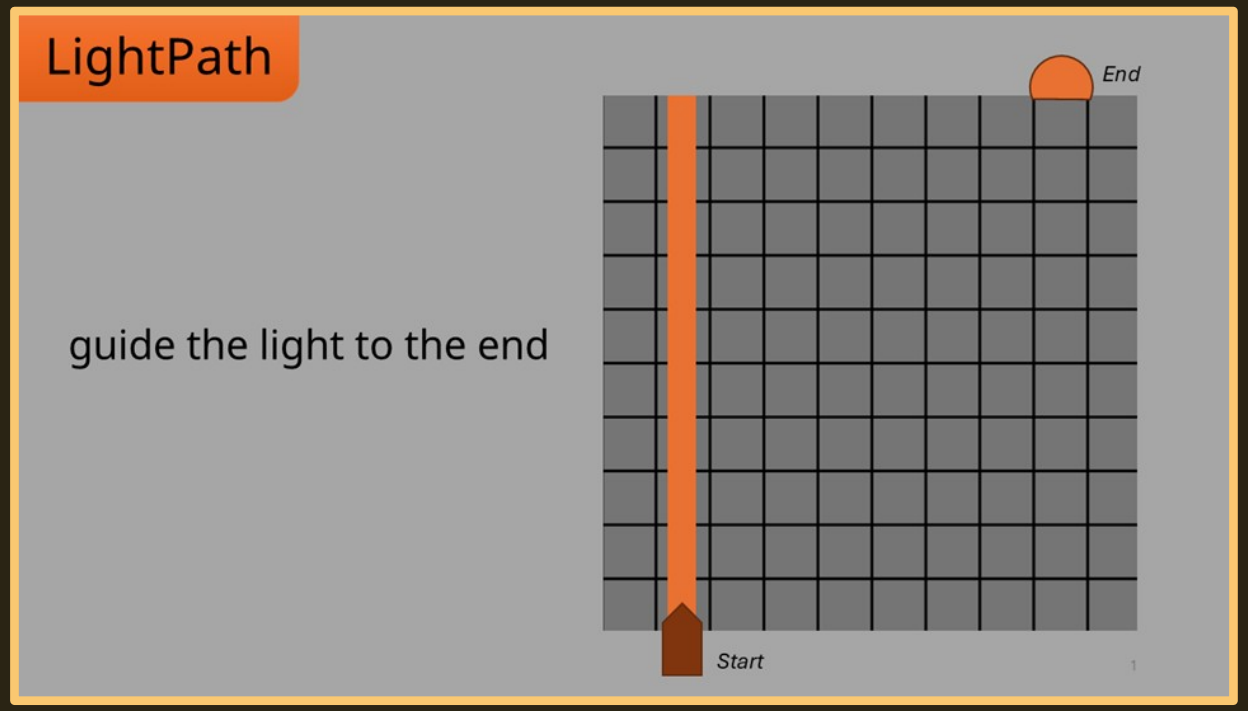
- Mechanic Diagrams
- A **puzzle design** to match the concept and solved with **expected player behaviour**
- Explored **expansion possibilities**
- **Self-critiqued** the result
- **Iterated** on the concept to solve all the issues I could find
- Designed another puzzle to **test the changes** and solved it with expected player behaviour.

Pre-Production

Concept

LightPath

- Genre: **Logic Maze**
- Place light bulbs on the grid, to create a continuous light to the end point(s).
- Partly inspired by a Japanese puzzle system called Akari.



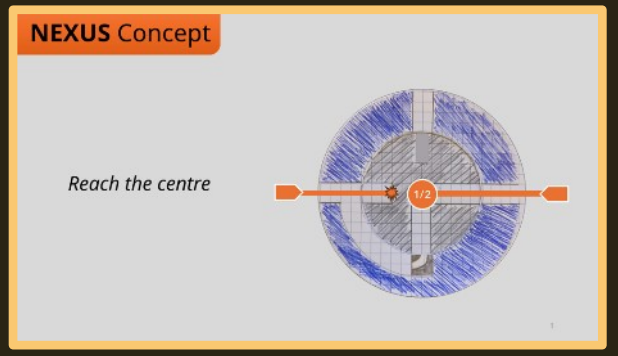
LightPath Documentation download link

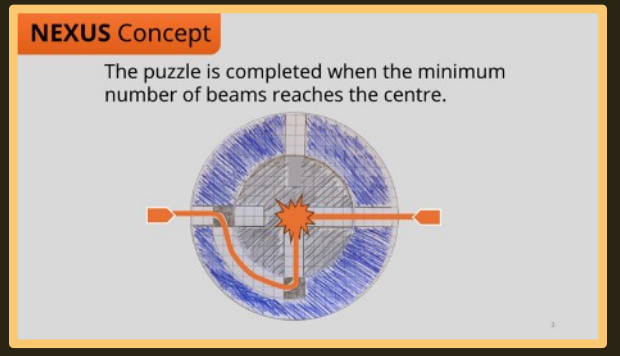
Pre-Production

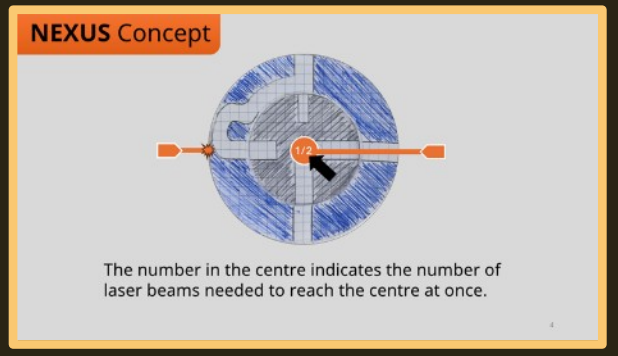
Concept

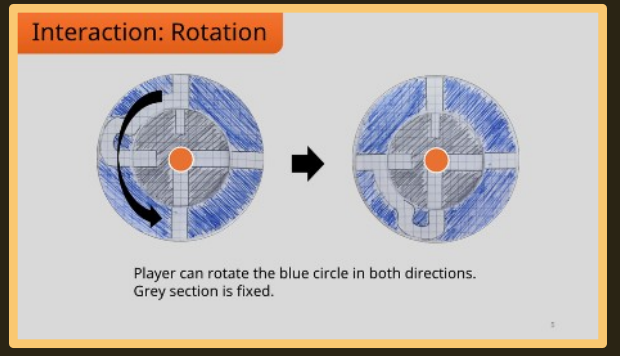
NEXUS

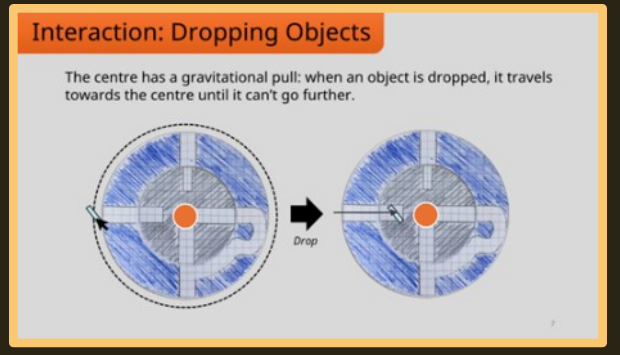
- Genre: **Logic Maze**
- Circular design where you are looking to **guide laser beams towards the centre**
- Features two core interactions:
  - **Rotating the rings** to change the puzzle configuration
  - **Dropping objects** from outside the puzzle to affect the laser beams' paths.











NEXUS Documentation download link

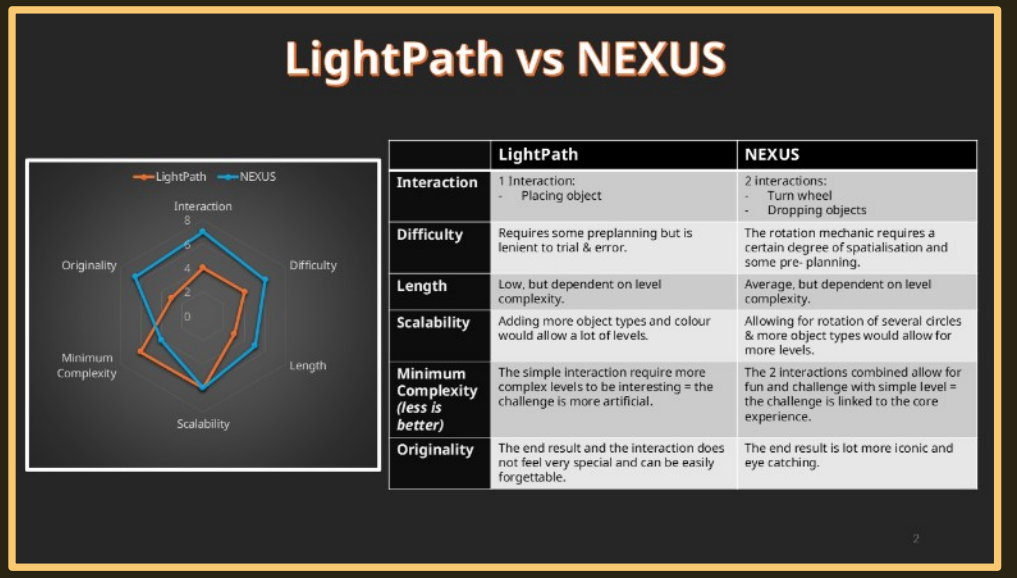
Pre-Production

Concept

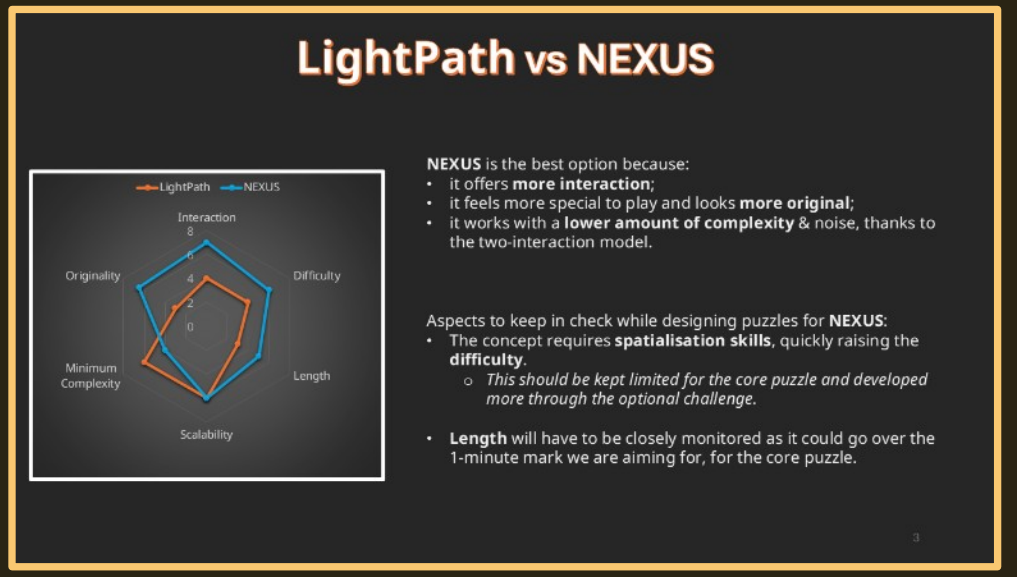
LightPath vs NEXUS Evaluation

- Both concepts were **evaluated** and **compared** on several metrics:
  - Interactions
  - Difficulty
  - Length
  - Scalability
  - Minimum Complexity
  - Originality
- Both concepts were also put against the **project parameters** set out at the end of the initial research.

- **NEXUS** came victorious!



	LightPath	NEXUS
Interaction	1 Interaction: - Placing object	2 interactions: - Turn wheel - Dropping objects
Difficulty	Requires some preplanning but is lenient to trial & error.	The rotation mechanic requires a certain degree of spatialisation and some pre- planning.
Length	Low, but dependent on level complexity.	Average, but dependent on level complexity.
Scalability	Adding more object types and colour would allow a lot of levels.	Allowing for rotation of several circles & more object types would allow for more levels.
Minimum Complexity *(less is better)*	The simple interaction require more complex levels to be interesting = the challenge is more artificial.	The 2 interactions combined allow for fun and challenge with simple level = the challenge is linked to the core experience.
Originality	The end result and the interaction does not feel very special and can be easily forgettable.	The end result is lot more iconic and eye catching.



Evaluation Documentation download link

Pre-Production

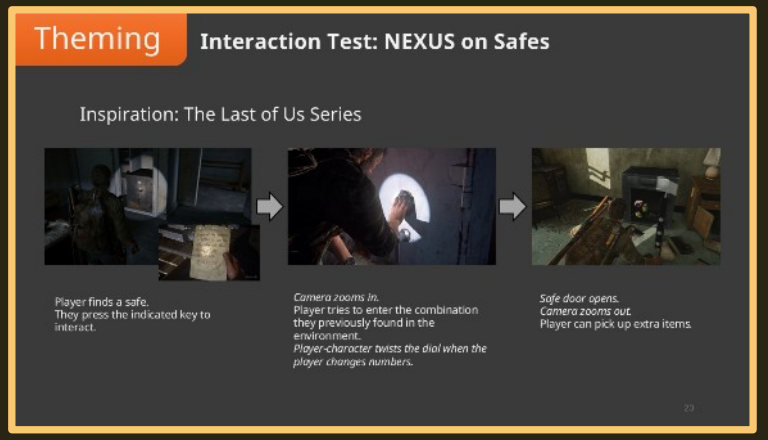
Concept

NEXUS Theming and In-World Integration

One of project parameters was to keep the puzzle minigame **grounded** in the world and not fully separated from the rest of the experience.

With **NEXUS** featuring a **circular** design, with the player **rotating** part of the puzzle and gaining **extra materials** when they succeed, it was clear that **NEXUS** needed to be themed around a **safe box**.

From there, we researched an interaction loop and designed in-world integration inspired by The Last of Us:



Production

Production

Prototype

Project Needs

NEXUS presented a number of specific needs:

- The player can select different **rings** and **rotate** them.
- Rings can only **stop at specific angles** (8-directional).
- They can **drop objects** from the outside in, and these need to get as close to the centre as possible when dropped.
- Objects and lasers are **not directly controlled** by the player but need instead to **find their own way** inside the puzzle.
- Object and lasers can only stop at **specific points** inside of the puzzle.
- Objects, once set in position, will need to **attach** to the ring they are on.
- Objects will **affect** other objects (i.e. blocking the path) and lasers (i.e. making it turn).
- The **optional challenges** (a key project parameter for this project) will require to **keep track** of every object used.

Production

Prototype

Implementation: Splines

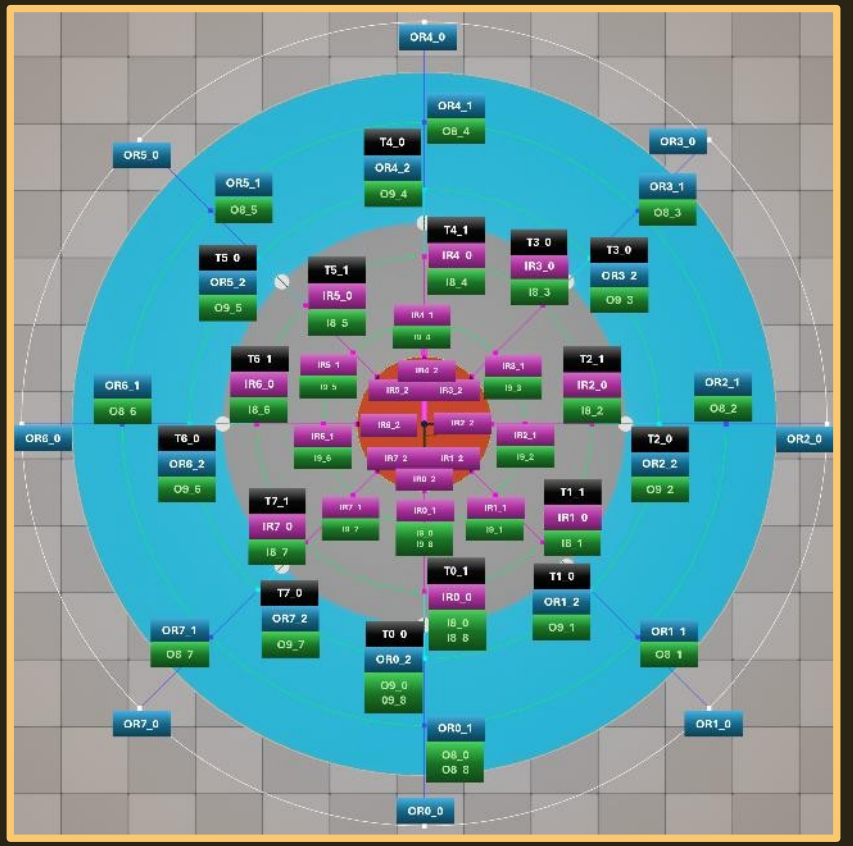
It was apparent that the solution to a lot of these needs was using **splines.**

- Every **path** a block or laser can take would be a **spline**.
- Every **position** where an event could happen (e.g. stopping or turning) would be a **spline point.**

Since rings can turn, and by extension the configuration of **a radius line change**, each radius could not be composed of just one spline, but three: one for the outer ring, one for the inner ring, one for transitioning between the two.

In total, each puzzle is composed of **30 splines:**

- 10 on the outer ring (8 radius + 2 circumference);
- 8 for the transitions between outer and inner rings;
- 10 on the inner ring (8 radius + 2 circumference);
- 1 for the “drop zone” (outside of the puzzle);
- 1 for the centre.



Production

Prototype

Implementation: Data Tables

Each **spline point** is referenced with a **code**, which can be found in each puzzle's **data table.**

The **first four columns** do not require any modification, as they are used by the system to **retrieve information** about a specific spline or spline point. They correspond to (left to right):

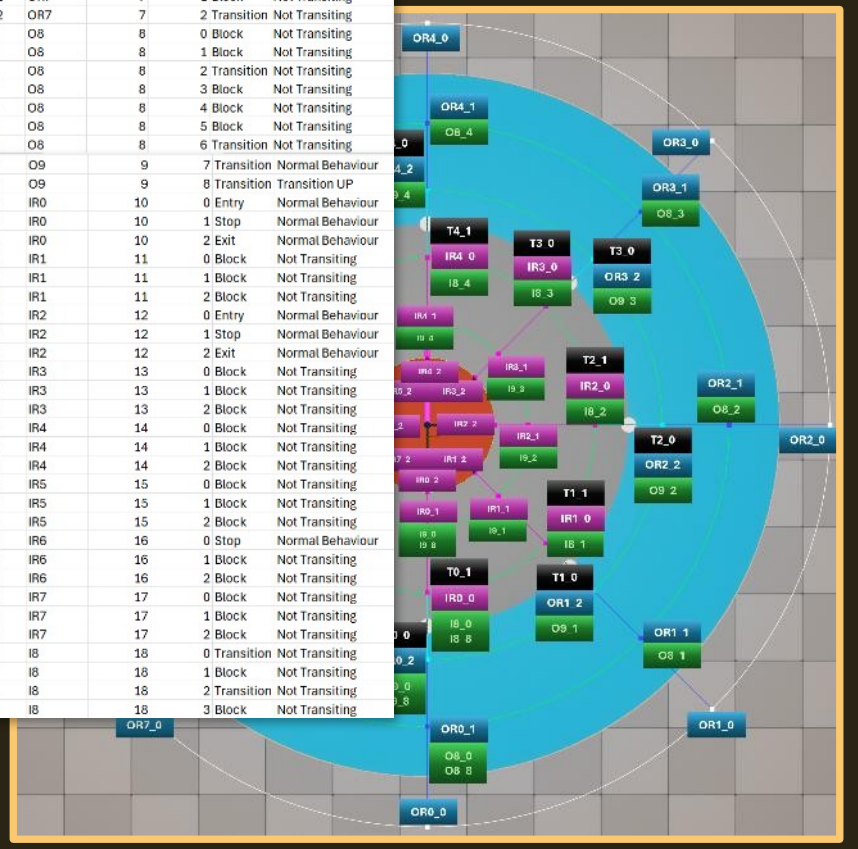
- The spline point code (SplineName_PointNumber)
- The spline name
- The spline number
- The point number

The **last two columns** need to be **manually set** for each puzzle, and they set the **starting state** of the puzzle:

- The **4th column** manages the **block behaviour.**
- The **5th column** manages **laser behaviour.**

Since data tables are read-only, the data is fed into a **struct array**, which will allow for **modifications at runtime.**

OR0_0	OR0	0	0	Block	Not Transiting
OR0_1	OR0	0	1	Block	Not Transiting
OR0_2	OR0	0	2	Exit	Normal Behaviour
OR1_0	OR1	1	0	Block	Not Transiting
OR1_1	OR1	1	1	Block	Not Transiting
OR1_2	OR1	1	2	Block	Not Transiting
OR2_0	OR2	2	0	Entry	Normal Behaviour
OR2_1	OR2	2	1	Transition	Normal Behaviour
OR2_2	OR2	2	2	Exit	Normal Behaviour
OR3_0	OR3	3	0	Block	Not Transiting
OR3_1	OR3	3	1	Block	Not Transiting
OR3_2	OR3	3	2	Block	Not Transiting
OR4_0	OR4	4	0	Block	Not Transiting
OR4_1	OR4	4	1	Block	Not Transiting
OR4_2	OR4	4	2	Block	Not Transiting
OR5_0	OR5	5	0	Block	Not Transiting
OR5_1	OR5	5	1	Block	Not Transiting
OR5_2	OR5	5	2	Block	Not Transiting
OR6_0	OR6	6	0	Entry	Normal Behaviour
OR6_1	OR6	6	1	Transition	Normal Behaviour
OR6_2	OR6	6	2	Exit	Normal Behaviour
OR7_0	OR7	7	0	Block	Not Transiting
OR7_1	OR7	7	1	Block	Not Transiting
OR7_2	OR7	7	2	Transition	Not Transiting
O8_0	O8	8	0	Block	Not Transiting
O8_1	O8	8	1	Block	Not Transiting
O8_2	O8	8	2	Transition	Not Transiting
O8_3	O8	8	3	Block	Not Transiting
O8_4	O8	8	4	Block	Not Transiting
O8_5	O8	8	5	Block	Not Transiting
O8_6	O8	8	6	Transition	Not Transiting
O9_7	O9	9	7	Transition	Normal Behaviour
O9_8	O9	9	8	Transition	Transition UP
IR0_0	IR0	10	0	Entry	Normal Behaviour
IR0_1	IR0	10	1	Stop	Normal Behaviour
IR0_2	IR0	10	2	Exit	Normal Behaviour
IR1_0	IR1	11	0	Block	Not Transiting
IR1_1	IR1	11	1	Block	Not Transiting
IR1_2	IR1	11	2	Block	Not Transiting
IR2_0	IR2	12	0	Entry	Normal Behaviour
IR2_1	IR2	12	1	Stop	Normal Behaviour
IR2_2	IR2	12	2	Exit	Normal Behaviour
IR3_0	IR3	13	0	Block	Not Transiting
IR3_1	IR3	13	1	Block	Not Transiting
IR3_2	IR3	13	2	Block	Not Transiting
IR4_0	IR4	14	0	Block	Not Transiting
IR4_1	IR4	14	1	Block	Not Transiting
IR4_2	IR4	14	2	Block	Not Transiting
IR5_0	IR5	15	0	Block	Not Transiting
IR5_1	IR5	15	1	Block	Not Transiting
IR5_2	IR5	15	2	Block	Not Transiting
IR6_0	IR6	16	0	Stop	Normal Behaviour
IR6_1	IR6	16	1	Block	Not Transiting
IR6_2	IR6	16	2	Block	Not Transiting
IR7_0	IR7	17	0	Block	Not Transiting
IR7_1	IR7	17	1	Block	Not Transiting
IR7_2	IR7	17	2	Block	Not Transiting
I8_0	I8	18	0	Transition	Not Transiting
I8_1	I8	18	1	Block	Not Transiting
I8_2	I8	18	2	Transition	Not Transiting
I8_3	I8	18	3	Block	Not Transiting



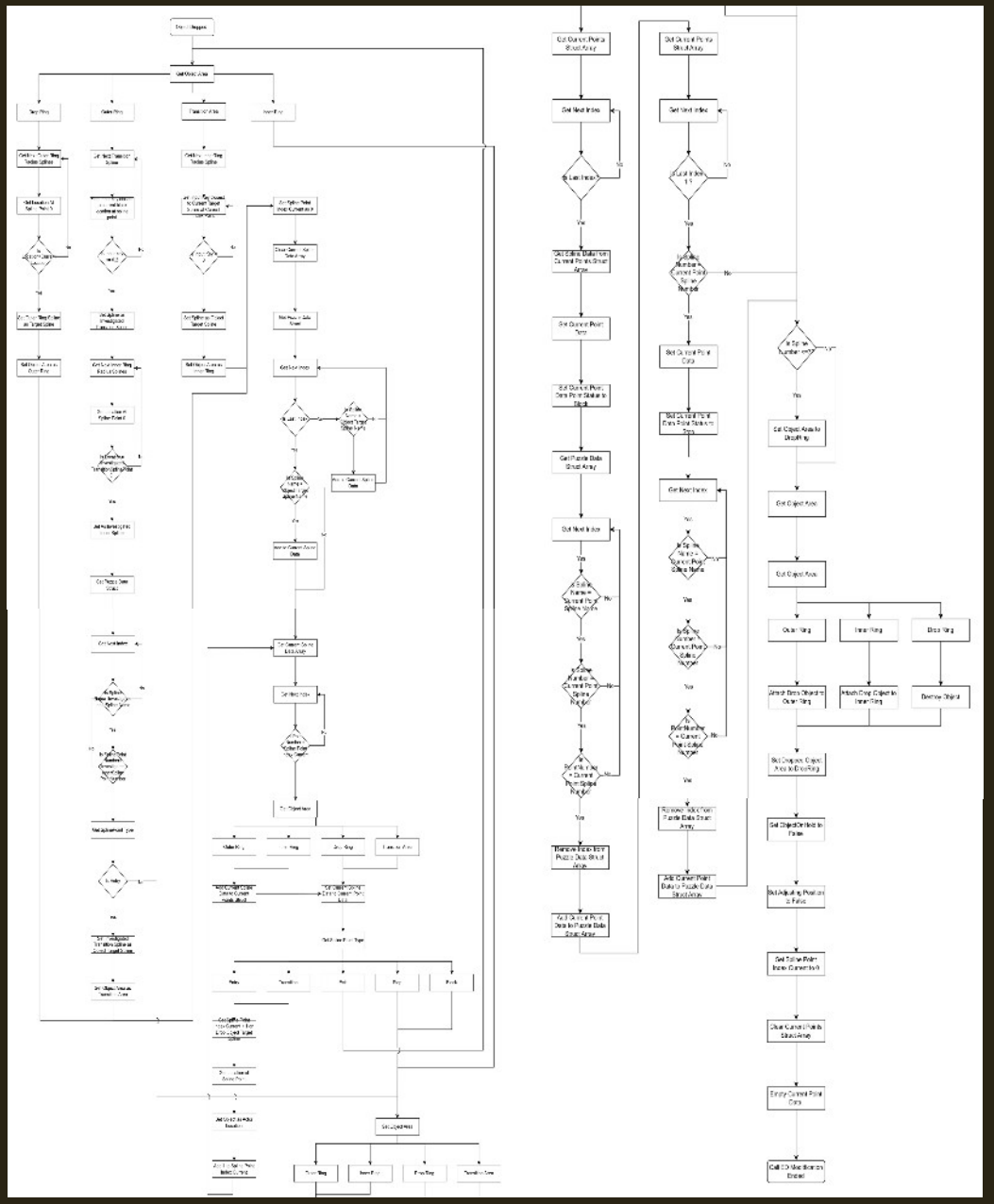
Production

Prototype

Implementation: Drop Object

In practice, the aforementioned struct arrays are for example used by drop objects enter the puzzle.

The diagram here shows the logic followed for drop object movement, from entry to lock in position.



Production

Prototype

System Architecture

Each puzzle is composed of:

- **the safe**
  - Interaction from player-character
- **the circular puzzle**
  - Rings rotation and selection
  - Module Mode movement
  - Modules movement in the puzzle
  - Configuration changes
- **a puzzle manager**
  - Puzzle specific objectives/win conditions
  - Puzzle specific modules
  - Communication with UI
  - Back/reset functions
- **one or more laser**
  - Determining path for laser inside of the puzzle
- **a laser manager**
  - Controls all the lasers
  - Communicates changes to puzzle manager

Production

Prototype

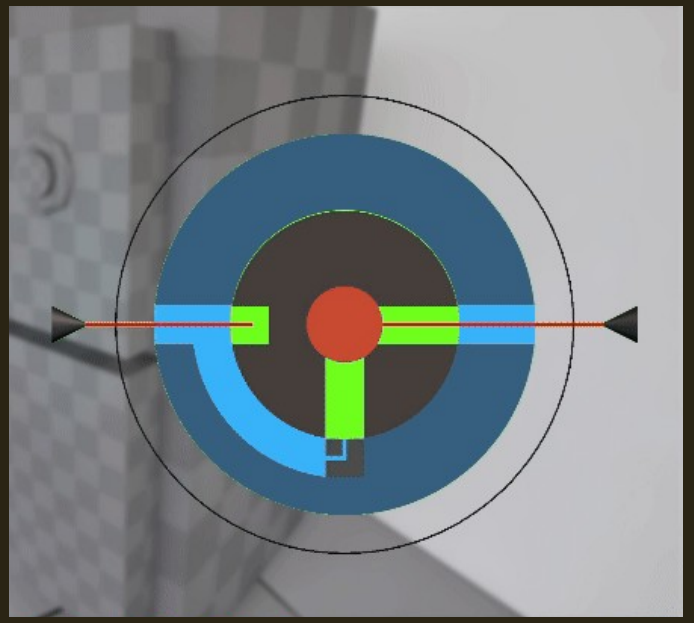
Implemented Gameplay Loop

In time for the mid-point, and on schedule, we were able to implement a **complete gameplay loop,** with the following features:

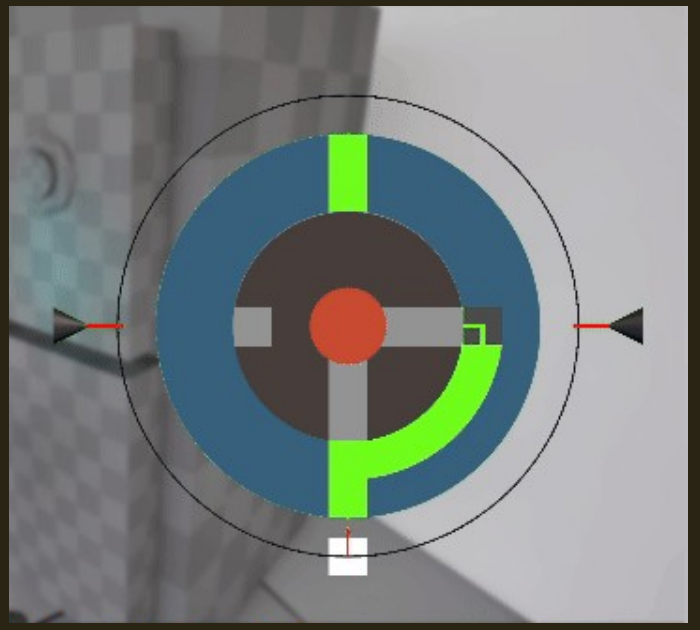
- **Rings** and **objects** automatically **align** with possible target splines.
- **Objects travel** through the puzzle, **respond to** then **modify** puzzle configuration.
- **Lasers** find their way inside the puzzle and **respond to configuration changes.**
- Puzzle have **specific main and side objectives**, tracked by the system.
- **Transition** from player-character to puzzle mode.
- **Rudimentary UI** for **objectives** and **success/failure feedback.**

Production

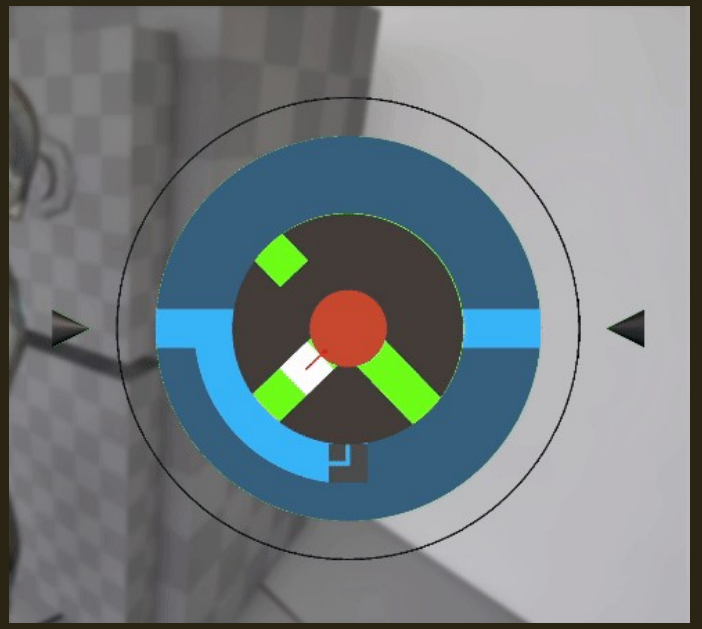
Prototype



Rings automatic position adjustment



Objects traveling and attaching to rings



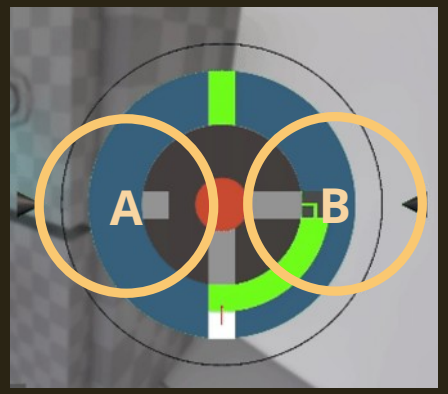
Lasers reacting to evolving puzzle configuration

Production

Controls

Prototype Issues

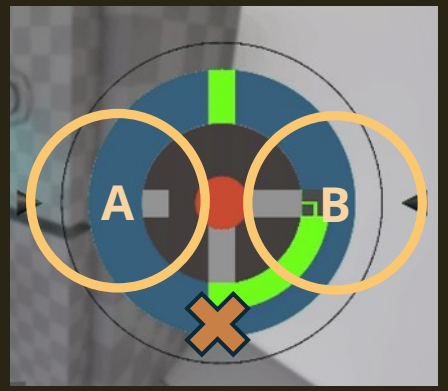
Up to this point, controls had only been functional, allowing for self-testing of the systems. With the circular design, **controls** became **unexpectedly challenging** to design for. For example:



Selecting Rings
Situation: the outer ring is selected; player wants to select the inner ring.

Expected behaviour:
- If they are looking at (A): Push Right button
- If they are looking at (B): Push Left button

If the 3 rings system (outer, inner, drop), this doesn't work!



Rotating Rings
Situation: Player wants to rotate the outer ring so (A) or (B) is aligned with **X**

Expected behaviour:
- If they are looking at (A): Push Down button
- If they are looking at (B): Push Down button

If the ring movement being built on a clockwise, counterclockwise system, this doesn't work!

Production

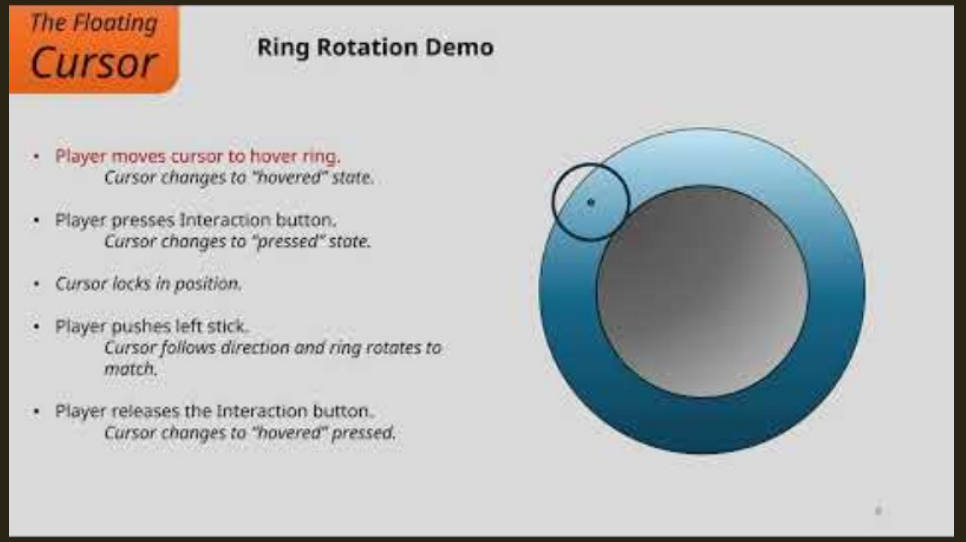
Controls

The Floating Cursor Solution

To solve these issues, we originally researched and designed a **floating cursor** control scheme. It would allow to keep the same controls on pc and console, and more importantly, would greatly **limit the number of buttons** the player has to use to play the minigame.

Some setbacks however meant that the solution could not be implemented. To allow for clean and uninterrupted transition between main gameplay and minigame, but also to be able to use 3D assets to represent the puzzle, we made the choice to have the actual puzzle somewhere else in the level and render it on the UI using a **target render.** While this works as expected, it also means that the object is not actually there, and the cursor could not “see” which part of the puzzle it was over (as it is just all just a texture).

Keeping the overall prototype architecture, it meant that **another solution** had to be designed, and the effects on the player experience monitored in playtesting.

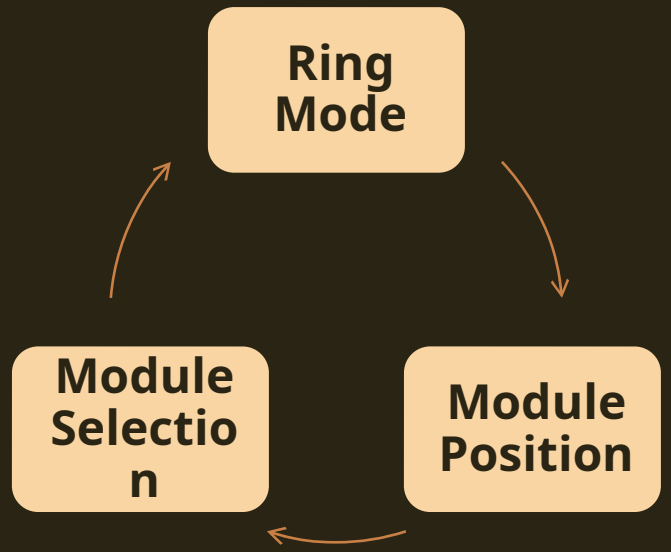


Production

Controls

Modes

With a more traditional control scheme needing to be used, we decided to divide **NEXUS** into several **phases**. This would allow to **limit the number of necessary keys** but also help to streamline the experience.



- **Ring Mode**
  - Select Rings
  - Rotate Rings
- **Module Position**
  - Move the indicator along the drop ring to select drop position
- **Module Selection**
  - Select the module to drop

Production

Controls

Buttons*	Modes		
	Ring Mode	Module Position	Module Selection
A		Enters Module Selection	Confirms Module Selection
Y	Enters Module Position	Enters Ring Mode	
B	Reverts to last step		
D-Pad Up/Down	Select Other Ring	Move Selector	
D-Pad Left/Right			Select Module to left/right
Left Bumper	Rotate Selected Ring Counterclockwise		
Right Bumper	Rotate Selected Ring clockwise		
Menu	Open Pause Menu		
View	Open Reset Menu		

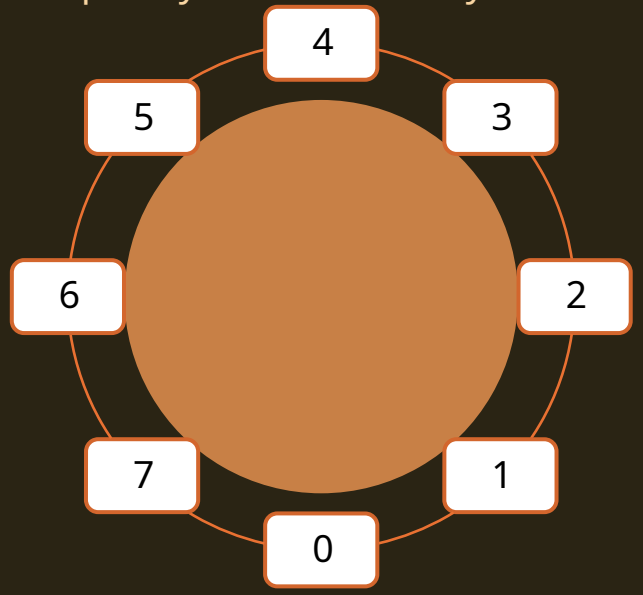
*Xbox Series X/S Wireless Controller

Production

Controls

A circular, spline-based design can bring a level of **complexity** to the controls. Since these must stay **intuitive** for the player, that complexity, while still present, should be **hidden** from them as much as possible.

For example, the diagram below represents the 8 positions where the module indicator can stop. This is, in fact, a spline (growing numbers indicating direction). But for the player, for instance moving from (3) to (2), it would be natural to press down. Alternatively, it would be natural to also press down to go from (6) to (7)... even though these are opposite directions on the spline! The solution was to customize controls for each position, to respect expected behaviour in a completely unintrusive way:



Button	Drop Position							
	0	1	2	3	4	5	6	7
Up	*	C	C	C	*	Cc	Cc	Cc
Down	*	Cc	Cc	Cc	*	C	C	C
Left	Cc	Cc	*	C	C	C	*	Cc
Right	C	C	*	Cc	Cc	Cc	*	C

C= Clockwise / Cc= Counterclockwise / *= No Movement

Production

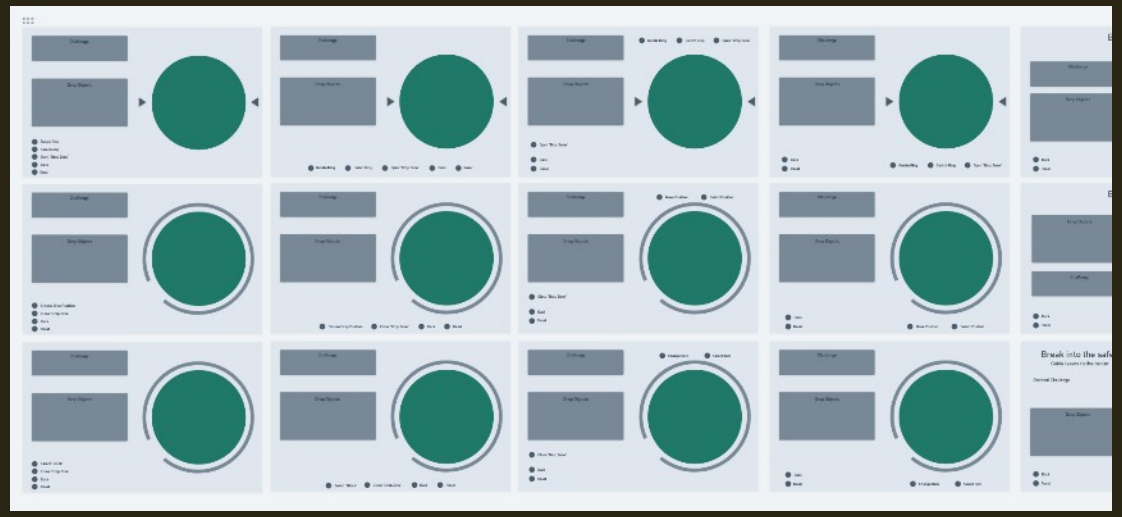
User Interface

Wireframe Iteration

The user interface for NEXUS needed to include the following elements:

- Clear, always visible **instructions.**
- Space for the **optional challenge** text and/or visuals
- The core puzzle.
- The Drop **object panel**, always visible even when not in direct use
- Always visible **on-screen controls**, adapting to the current mode)
- Enough space to still see the safe and player-character interacting.

Several wireframe variations were tested for the project, each time for each of the different modes in the minigame, with the different elements positioned and grouped differently.



Production

User Interface

Chosen Wireframe

- The top-right of the screen features the **objectives** of the minigame (set and changing optional challenge).
- The middle of the screen features the elements the player should mostly focus on: the puzzle itself, and the module panel.
- On screen controls are split into two:
  - Under the puzzle, the player can see the directly applicable **controls for the current phase** of the puzzle. These change and adapt depending on the current phase.
  - At the bottom-right are controls that are available in every phase. These were separated to make the other controls easier to read and to handle.

Production

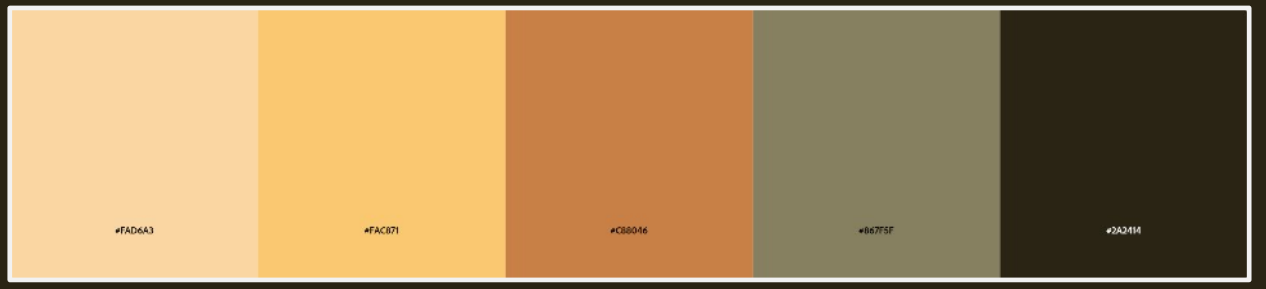
Visual Identity

Concept

If **NEXUS** had been created as a minigame for a specific title, the visual identity would have been greatly influenced by said title. But since this title does not exist, we had more freedom on the matter.

With the player looking to break into a safe, but also with puzzle part being quite bulky (and so not very modern), we started to look at **antique bank safes**, which would often feature dark background, and **rich, golden accent colours**. They also usually featured **ornate frames and fonts.**

These elements could be then brought into the puzzle itself and the user interface, for a cohesive and immersive experience.



Production

Visual Identity

The result of visual concept and UI research all came together. It allows NEXUS to look and feel more cohesive, and more importantly, a lot less abstract than at the prototype stage.

Production

Tutorials

To support onboarding, a series of **tutorial screens** were created on the following subjects:

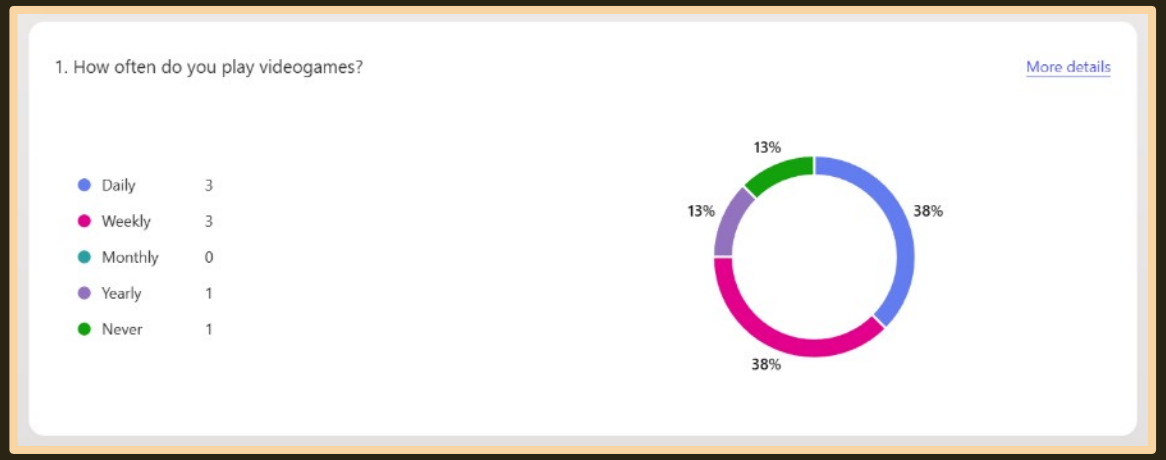
- Minigame Introduction
- Rotating Rings
- Modules
- Module Selection
- Optional Challenge
- Back/reset function

Each screen is supported by a video demonstrating the information.
Rather than all trigger at the start of the game, they are **progressively introduced** through the first puzzles to help with information retention.

Production

User Testing

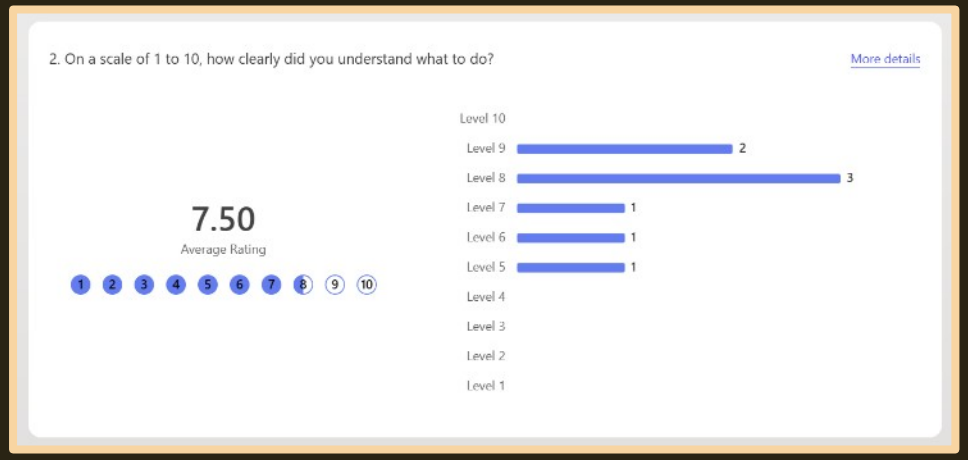
User testing for this project was conducted between Feb 3rd and 9th 2025.
The build was composed of 2 puzzles and all features integrated at this point (including all tutorials).
After playing, testers were provided with an online form regarding their experience. Below are the results of the study:

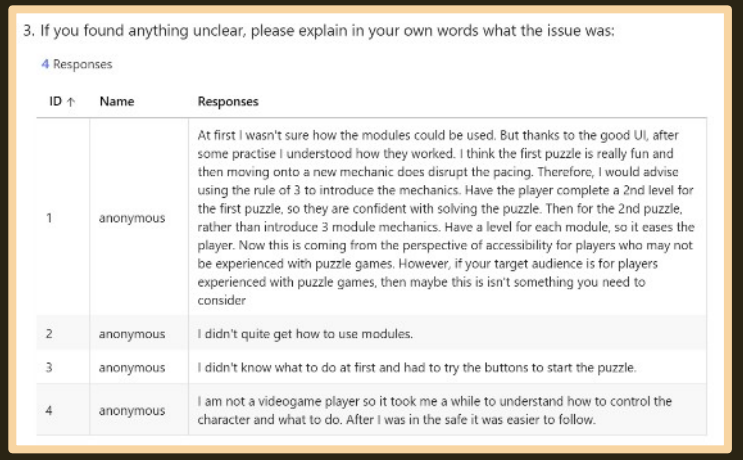


With this minigame looking to fit in AAA, it was important to try to have a broad range of player. Some (very) occasional players were included in the study.

Production

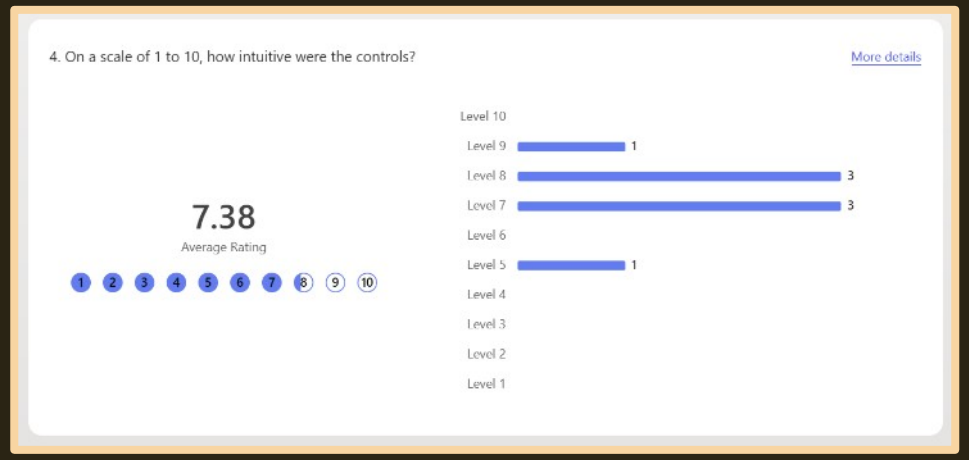
User Testing

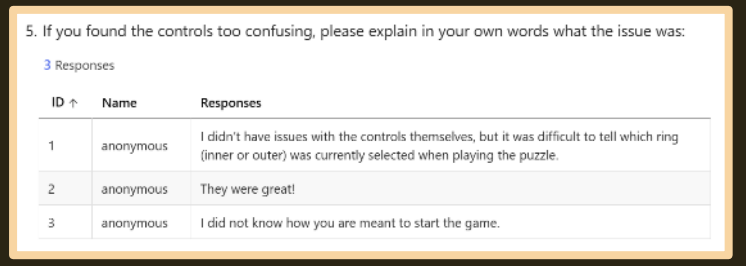




Takeaways:

- *Adding a button prompt to show how to interact with safe.*
- *Modules are introduced too quickly. As suggested, adding an extra puzzle in-between the two present here will help.*





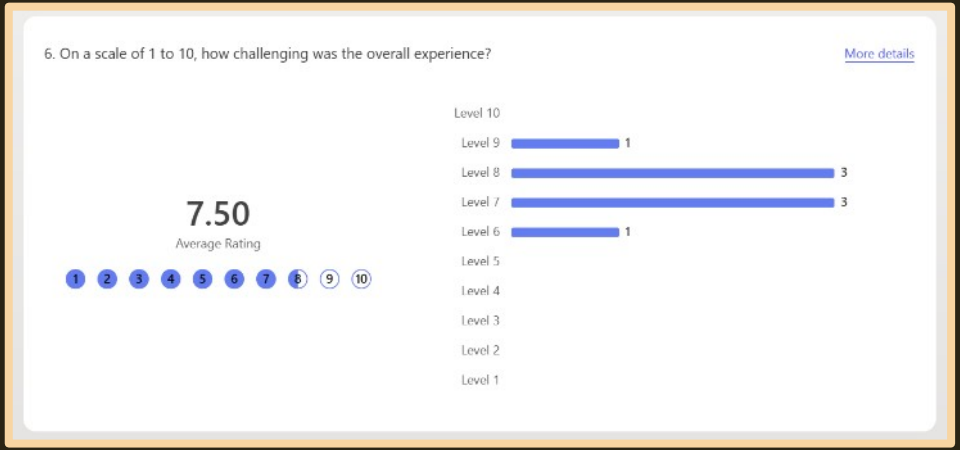
Takeaways:

- *Adding a button prompt to show how to interact with safe.*
- *Letting player use D-pad Up/Down too to select rings*

As a real design challenge during production, I was pleasantly surprised with the results!

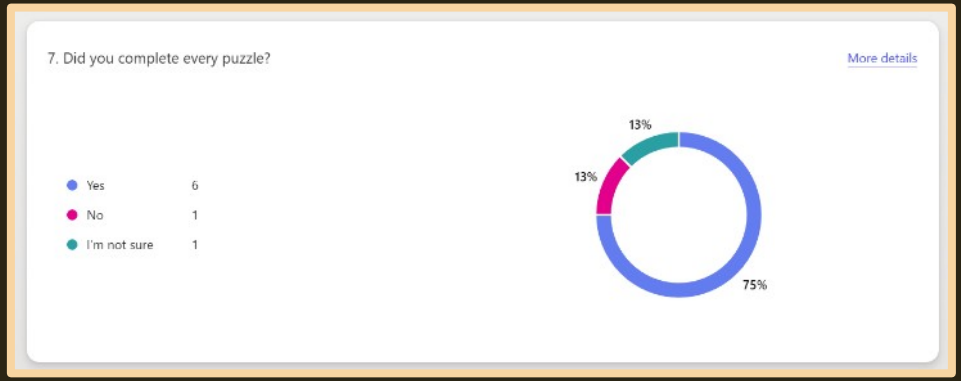
Production

User Testing



Takeaways:

- *7.5 is a bit too high for this project's goals*
- *Making the introduction of modules more gradual should lower this number.*

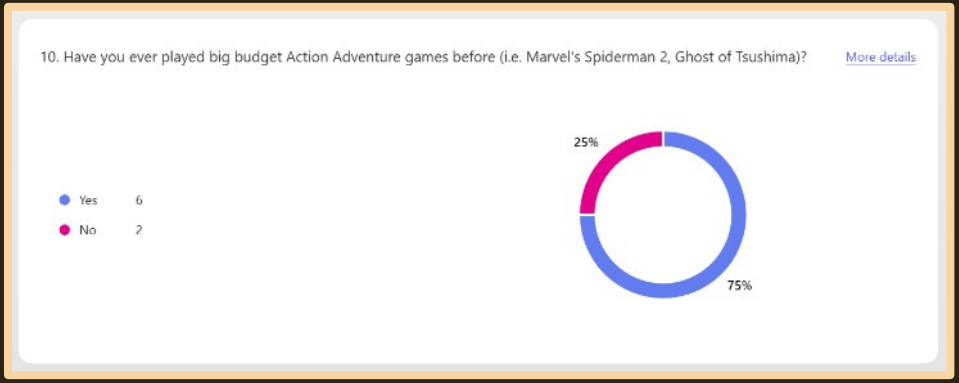


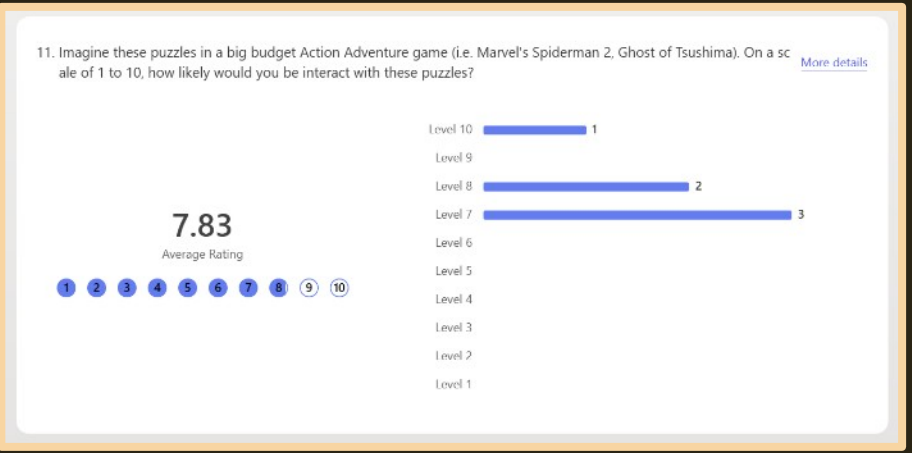
Takeaways:

- *The restart and reset functions have proven to be buggier than expected!*

Production

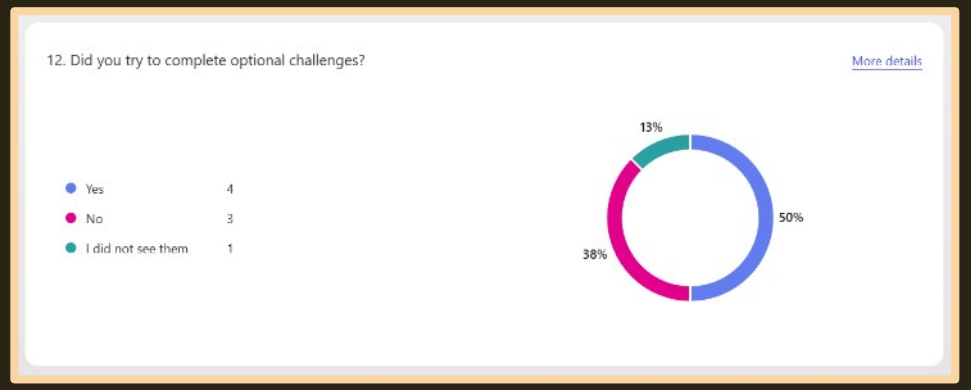
User Testing

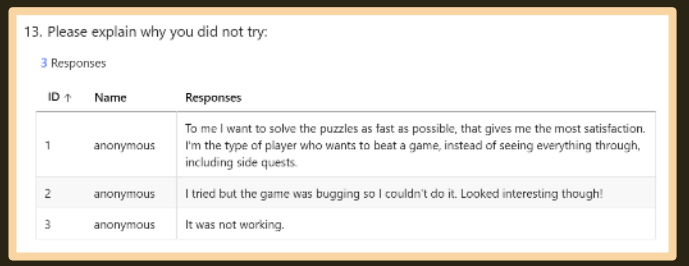


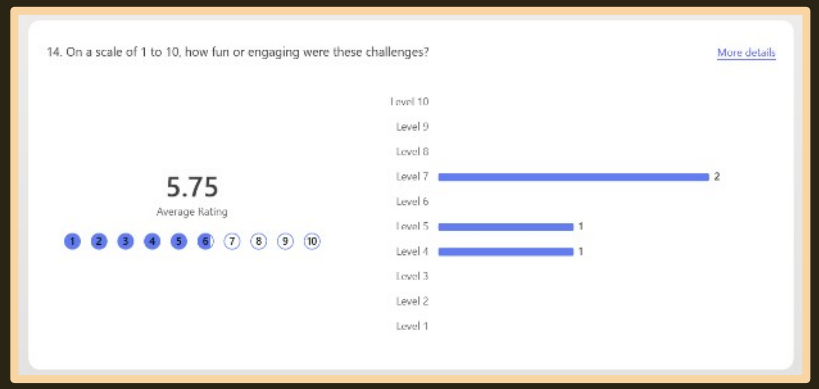


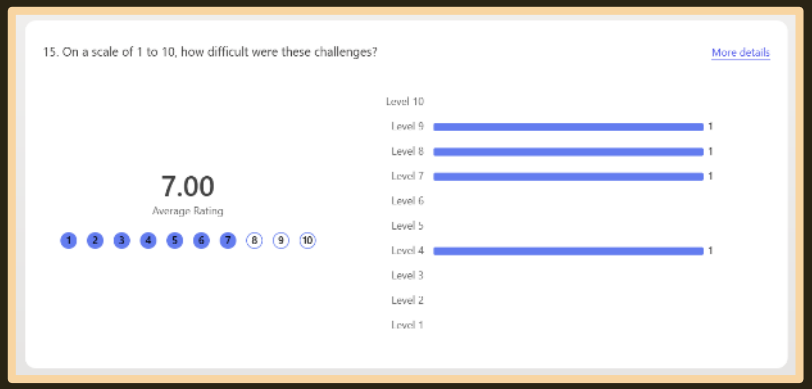
Production

User Testing









Takeaways:

- *Once more, the real issue were the buggy restart/reset functions, which prevented some players from being able to achieve full completion of the puzzles.*

Production

User Testing

16. Is there anything you would like to see added to the game?

6 Responses

ID ↑	Name	Responses
1	anonymous	More levels, that ease the player into each new mechanic, i.e. rule of 3. UI is really good and guides the player into solving the puzzle themselves. Overall this was fun, thank you!!
2	anonymous	I can see these minigames being used to open big rooms or big chests, but for something as often as Skyrim utilises lock breaking, I feel it could be too big of a minigame for the players rewards. Either way its a fantastic project which was challenging, well explained and executed. As for something being added, maybe a prompt that tells the player to press A on the safes. :)
3	anonymous	A UI prompt when you first go up to the safe would be helpful. I think the UI for the puzzles looks clean and concise. With the optional challenge I completed it without really doing much so maybe that was a bug or it needs to be more challenging. Overall I had a very enjoyable experience.
4	anonymous	it was difficult to manage the second puzzle and I felt like I had too much to do.
5	anonymous	I feel like the second puzzle was quite overwhelming compared to the first.
6	anonymous	I think there is a lot of bugs in the game.

Takeaway Summary

- A button prompt for interacting with the safe.
- Debugging the reset/restart function.

These have been worked-on for Build2

- An intermediary puzzle to lead into the 3rd puzzle.

To be worked worked-on for later Builds

Final Artifact


Break into the safe
Guide both lasers to the center
Optional Challenge
MODULES

Evaluation

Evaluation

Initial Proposal

"a puzzle minigame system for AAA action-adventure titles, that caters to puzzle fans and actions fans alike."

The addition of **back/reset functions,** along with extensive **onboarding** features, are meant to keep the design **accessible to non-puzzle fans**. Thanks to the addition of the **optional challenges**, the puzzle minigame system is designed to also **cater to puzzle fans.** In its current form, the project still **needs to develop its number of level** in order to truly evaluate the success of the proposal. Playtesting showed that the **difficulty curve** for the critical path was **too high**, and further iterations on the project will improve it.

Deliverables

For this project, our initial deliverables consisted of :

- *"A UE5 **artifact** for the puzzle minigame system, industry-matching in size and scope"*

We were able to deliver a functioning artifact for the puzzle minigame system. At the time of writing (Feb. 2025), it is not yet of industry-matching scope, as some of the **priorities shifted** during development. As part of our research, we determined that integration within the world, and visual feedback, were very important to the success of a minigame in this context. With the original plan of working on visuals between the February deadline and GradEX, we had to shift tasks, to ensure that visuals and onboarding were to a certain standards before playtesting. Otherwise, the puzzle system would have been **too abstract.**

- *"Supported by a **research** paper;*
- *Design **documentation.**"*

A **9000+ words research document** was produced through the making of this project. For the design documentation, with the need to present new developments to our supervisor, we took the approach of creating **smaller presentation decks.**

Evaluation

Unexpected Challenges

Controls were a real sticking point throughout the development. The circular/spline-based design meant that it took **a lot of trial and error** and iteration to land on something which would still feel **intuitive** to the player. I believe that some of these issues and setbacks could have been avoided if the controls had been considered earlier on, before setting out the architecture, to allow for the preferred floating cursor solution.
I was frankly still expecting players to struggle with the controls, but it was important to take the current solution to playtesting to determine how much of a shift needed to be done. **Playtesting** results however showed that they seemed to work well, and no player complained about them.
A challenge remains on making the game playable on keyboard/mouse, as it is currently not available.

Going Forward

In the next 3 months leading to GradEX 2025, we are planning to work on:

- Creating **more puzzles** and playtest them
- **Improved visuals**, for example regarding the laser
- Adding **sound**/music
- Adding **keyboard/mouse controls**
- **Improved stability** and bug squashing

Bibliography

Journals and Books

- Adams, E. (2014) *Fundamentals of Puzzle and Casual Game Design*
- Schell, J. (2020) *The art of game design: a book of lenses*

GDC Talks

- Brett, T. (2024) *The Puzzle Vocabulary Toolbox: How to Think and Communicate in the Language of Puzzle Design*
- Traynor, P. (2024) *System-Centric Puzzle Design in "Patrick's Paradox"*
- Blow, J. (2011) *Truth in Game Design*
- Menzel, J. (2016) *Level Design Workshop: Solving Puzzle Design*
- Elyot, G. (2021) *30 Puzzle Design Lessons From The World's Greatest Puzzle Communities*

Engine Documentation

- Blueprint Splines, accessed November 2024, <Blueprint Splines in Unreal Engine | Unreal Engine 5.5 Documentation | Epic Developer Community>
- Landscape Splines, accessed November 2024, <Landscape Splines in Unreal Engine | Unreal Engine 5.5 Documentation | Epic Developer Community>

YouTube Tutorials

- How to make Splines using Blueprints | Spline Meshes | Unreal Engine 5 Tutorial
- Splines And Spline Meshes | Creating A Road System Using Splines - Unreal Engine Tutorial
- Populating Meshes Along a Spline | Tips & Tricks | Unreal Engine
- UE5 Tutorial - Add Video to a Widget

Outsourced Assets

- Monteiro, A. *Xbox Controller Icons Free*
- Sorkin Type Co., Limelight Font (licensed under the SIL Open Font License, Version 1.1)

Thank you for reading.