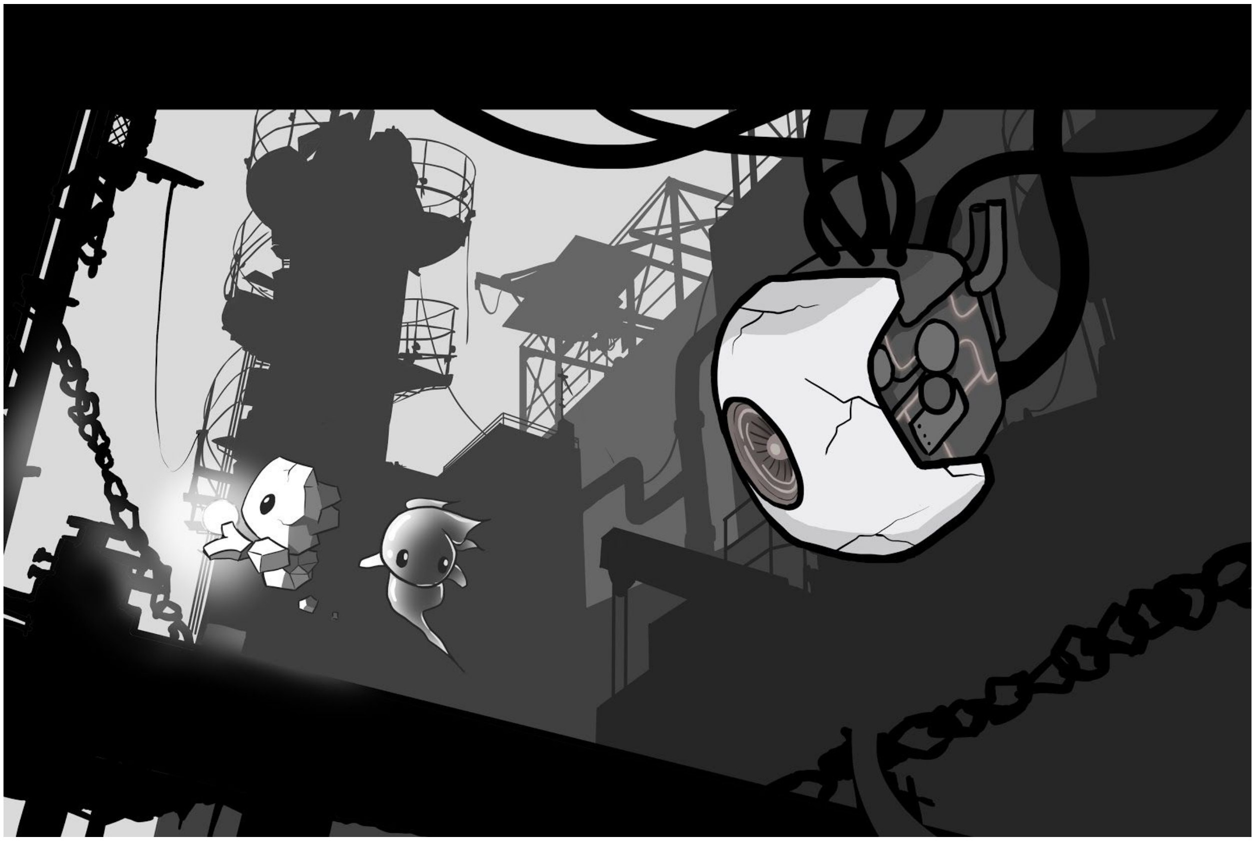
Resurgence

LEVEL DESIGN DOCUMENT



ANTHONY LIAO / GABRIELE DAL CENGIO / KEEGAN GEORGE / KENNETH JUANG / RYAN DONNELLY

Rationale & Outcomes:

What design questions you have intended to answer with this prototype? (focus on interface design and level features beyond Level 1)

The design questions we are trying to answer with the playtesting include:

- How much and how long do we have to guide players for?
    - Do players need constant reminder for the control layout?
    - At what point can we stop providing reminders?
- Are we implementing enough feedback so that players understand what it is they're doing?
    - Can players understand our teaching methods?
    - Is there enough feedback that relates to their in-game action to bring out excitement?
- Does a lack of UI make it difficult for players to understand the functionality of our game?
    - Is there too much ambiguity in our puzzle solution? Can more comprehensible tutorial UI solve this issue?
    - Can players distinguish what items are intractable or not? What can we do to make it more understandable?

How are those intended questions are clarified with playtesting (what works and what doesn't)?

For this submission, our digital prototype is still focusing on testing out the mechanics, kinesthetics and technology components of our game.

- Mechanics
    - How well players can use their individual abilities?
    - Can players recognize when to use their abilities?
- Kinesthetics
    - Are we providing enough visual affordances for players to understand interactions possibilities?
    - Can players recognize and understand our implemented visual cues?
- Technology
    - Do we need more UI elements to inform players of game functions and elements?
    - How long do players require reminders for the control mapping?

Any unexpected outcomes? Does it mean a new gameplay experience to include as a feature or a problem to resolve?

The biggest unexpected outcome came from the abuse of the slapping mechanic. It is a joint-play interaction we implemented that lets our players mess around while playing the game, but more often than not, it became more about who can cause the other to mess up the game, rather than a change of pace from the game itself.

Because our game is a casual intended for cooperative play, we are thinking of keeping this feature because the fun experience is unique when compared to the gameplay experience.

What problems need to be resolved with final polishing?

Some final polish we need to change in our game are:

- Sound effects for abilities for audible feedback
- Implement more animations as part of visual feedback
- Background music to drive the gameplay experience

Design Refinements as Per Internal Review

- Order of asset placement/drawing (having the platform on top of lasers, etc)
- Level showcase as animation intro for levels
    - show where the exit point is first and zoom into it, then slowly zoom out and pan to the players' starting point so they visually understand where their goal is located
- Hitboxes or button hovers need to be recalibrated so that when it hovers players can activate the ability
- Indicate to players that the catalyst needs to be dropped before using their abilities
- Power tracking bar for charging the platform throw (transpose ability)

Design refinements based on playtesting feedbacks and analyses

- Geyser ability art style
- Introducing the abilities prior to playing the level so players can understand the hovers actually mean
- Deceleration needs to be more instantaneous because the character is slipping and messing up player timing for jumps

Feedback Model:

Our game primarily uses a non-diegetic interface to provide feedback elements for the players. This interface is to teach our players, and this is done through a combination of button hovers and spatial components. Button hovers are triggered as players approach an interactable element, and if players recognize it, there can visually see a relationship between the hovered button(s) and the character. In terms of the spatial components,they are glow effects highlighted on these interactable elements. This acts as a visual affordance so the players can understand that this game asset differs from the game environment.

Our game is designed with a lack of information required to be tracked. As per the rule of functionality preservation, this allows us to have a lack in UI to better immerse players into our game, as there is no element which feels intrusive to the game.

Control Scheme Table:

Our control scheme employs the use of physical controllers for input. This is important because our game is a cooperative game. Hence, by utilizing controllers, players have the freedom and space to operate the controls for the game without hindrance to their partner. Specifically, the physical controllers used are Dualshock 3 controllers. The controls utilize the directional pad, shoulder, and action buttons. The d-pad is primarily used for movement. The shoulder buttons of L1 and R1 are used to control each of the player's two abilities. Furthermore, the action button of square is used for pickup or drop the catalyst, while the circle button is used for activating switches.

Macro Document (Constraints):

What this level is about?

Our first few levels are designed to educate players on cooperative play and introduce game elements such as enemies, electric zappers, etc.

For the beta iteration, we have added feature completion elements to our existing levels, and are currently building out the boss level as the climax for our game. Following the alpha submission, we are still testing out the same levels for our beta sessions, but our main focus is on the visual feedback for our players via animations in relation to the gameplay experience itself.

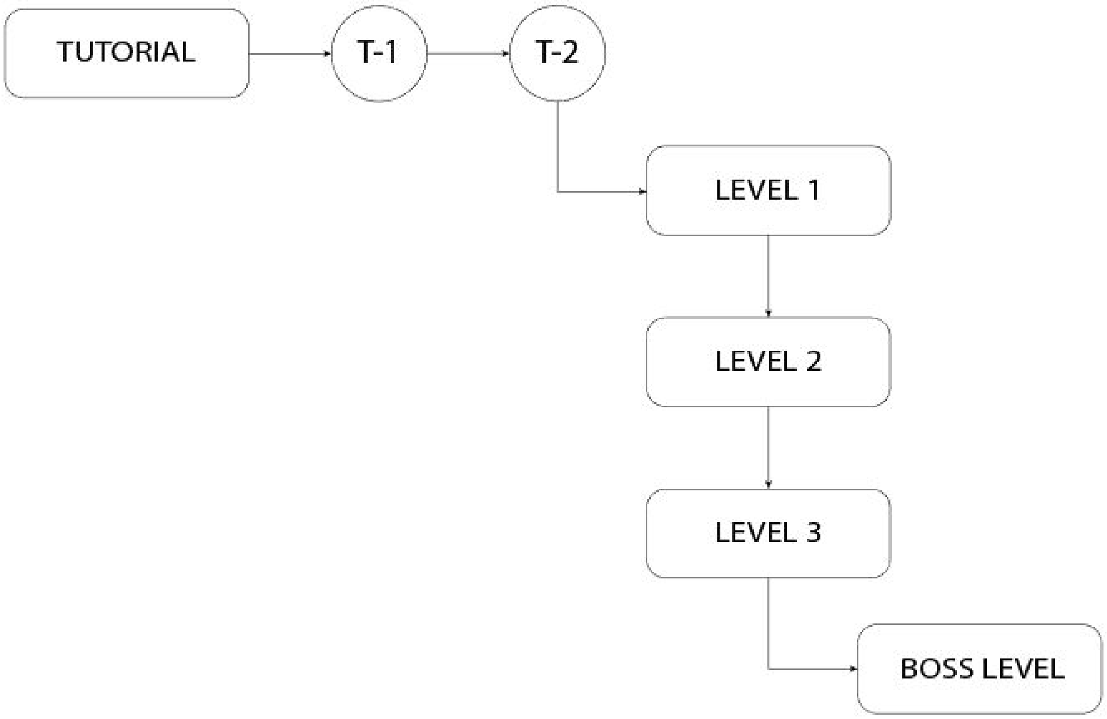
Basic archetypes of mechanics:

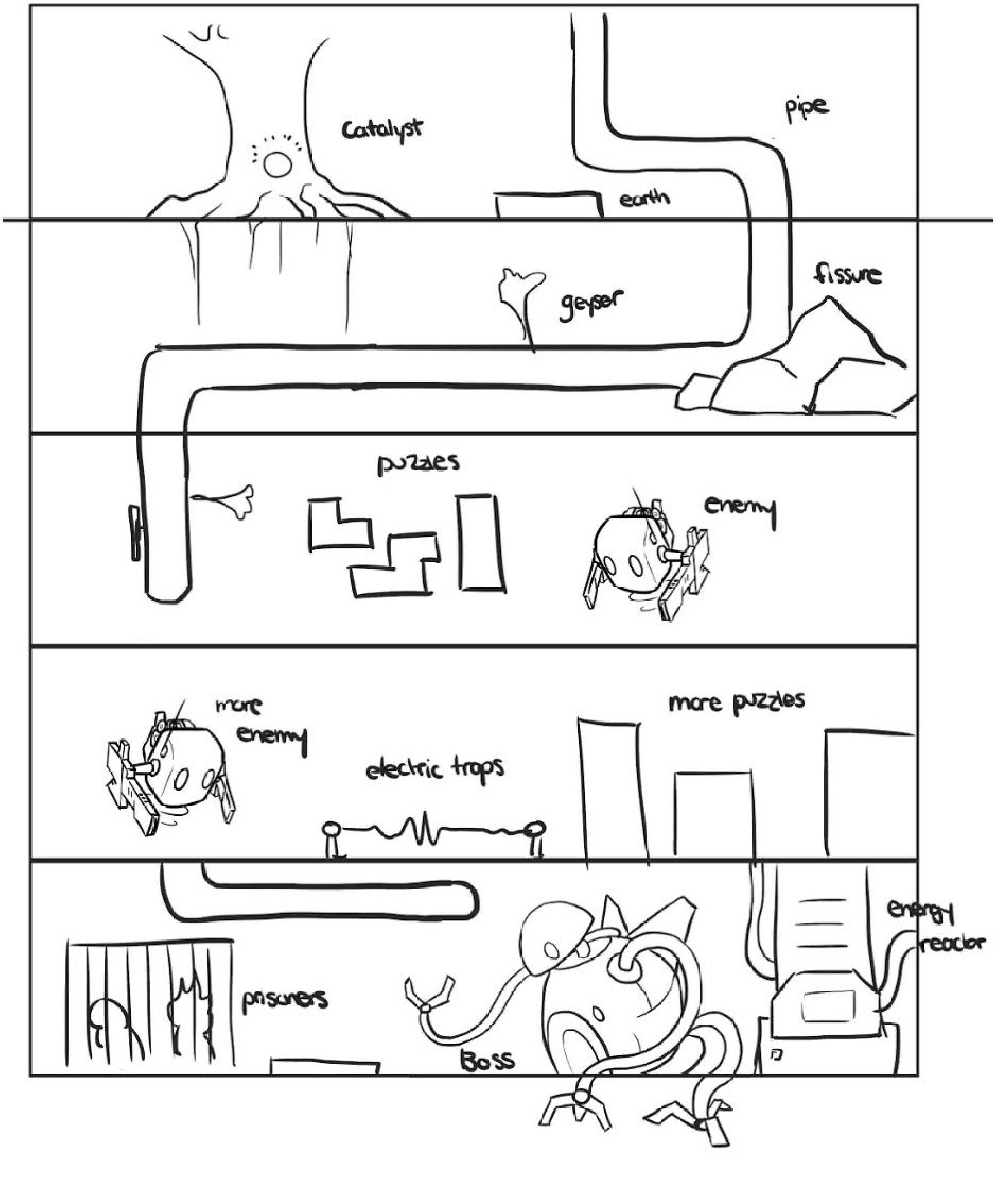
For this beta submission, we have five main archetypes for our level mechanics:

- **Terrain**: Each level features a unique layout of platforms, constructing puzzles of varying forms and difficulties.
- **Skill-interactables**: Interactive elements are strategically scattered throughout each level for the players to utilize their skills and solve puzzles with
- **Basic enemies**: These act as opposing forces to the player throughout the levels which appears in the form of Godots
- **Catalyst**: Players must bring the catalyst with them to the exit point to proceed to the next level.

How do mechanics archetype ramp up within each level and between levels?

The increasing difficulty is found in the progressive complexity of terrains. This includes the formation of platforms and the placements of other archetypes (skill-interactables, enemies, and catalyst) to construct challenge setups for a gradual ramp up. As players approach the end of the game, they enter the boss room where they must use the mechanics they've learned to defeat Azara.

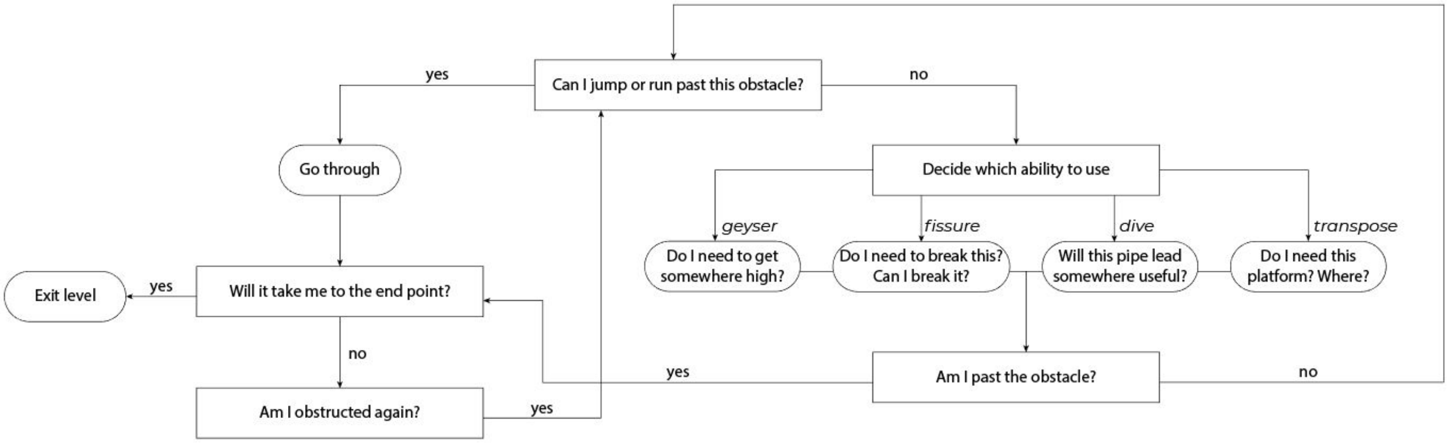



Catalyst
pipe
earth
geyser
fissure
puzzles
enemy
more enemy
electric traps
more puzzles
energy reactor
prisoners
Boss

Leading The Player

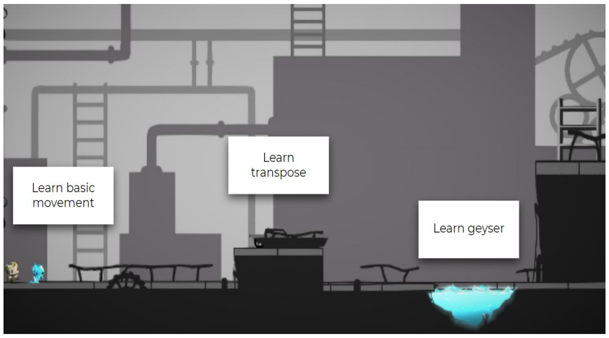
How do the “moment to moment” sub-objectives fit into the major objective of the level, which in turn fits into the final goal of the game?

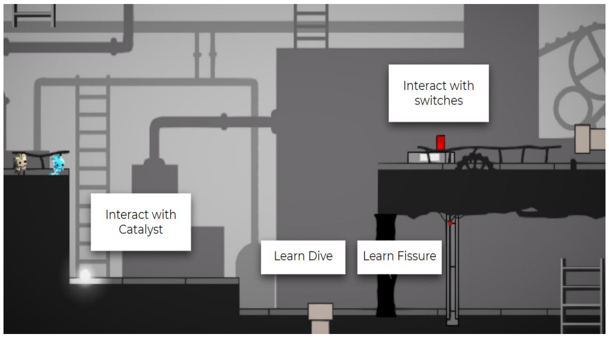
We lead our players physically, and showing them the start and end points of the level. It is up to the player to discover how to move about the level. By leading players through the levels and designing puzzles for them to solve, we create many moment to moment sub-objectives which consists of solving puzzles in a cooperative manner and must be accomplished to complete the level. When players trigger the Azara’s entrance, their goal becomes interrupted by this final hurdle that they must defeat.


Can I jump or run past this obstacle?
yes
no
Go through
Decide which ability to use
geyser
fissure
dive
transpose
Do I need to get somewhere high?
Do I need to break this? Can I break it?
Will this pipe lead somewhere useful?
Do I need this platform? Where?
Exit level
yes
Will it take me to the end point?
no
Am I obstructed again?
yes
yes
Am I past the obstacle?
no

How the paths are laid out, what layout pattern is followed?

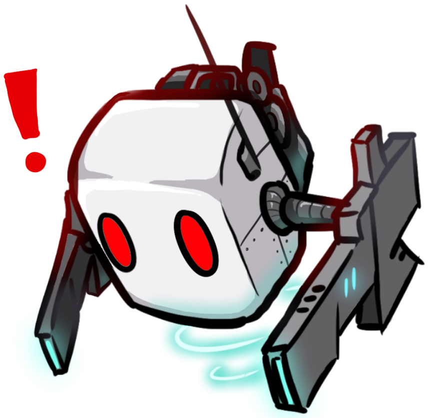
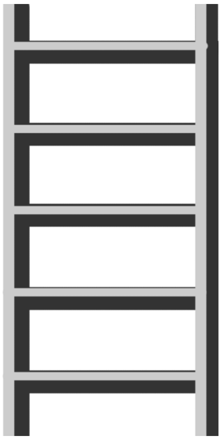
The layouts for our levels are linear. Players can see the level as a whole and decide how to approach it, but regardless, there is always a certain route/path to take that leads to solving the puzzles and reaching the level’s end point. At times, the level will allow for small offshoots for the players to choose.





Visual cues or any other means used to guide the player in terms of where they should go and what they should learn/do along the paths

- The glowing light that is emitted players come near interactable objects
- Enemies blocking certain areas (ex. player needs to get past enemy)
- Ladders
- Zappers which enemies must somehow advance across
- Appearance of button mapping during tutorial levels

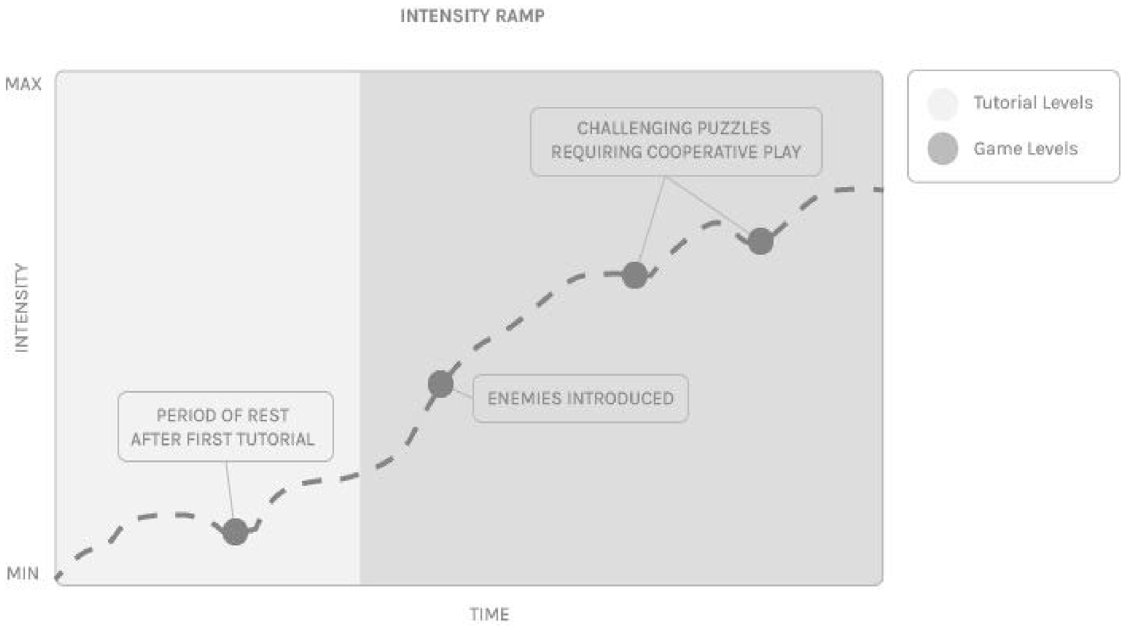
Major Beats and Pacing Rhythm

What key moments (in terms of both gameplay beats and narrative beats) have you highlighted in this level?

During easier levels, we implemented slower pacing schemes to match with its lack of danger. During later stages, when puzzles get harder, faster pacing is used when players are required to avoid an enemy or get past death-causing obstructions (zappers). Beyond this, we are feeding the players narrative to align with the gameplay beats and elevate a feeling of increasing intensity through gameplay.

Because the Godot's are minions seeking to destroy both intruders and the catalyst, we wanted the intensity between players and the Godot to be the more intense compared to the other assets. However, we still needed a balance between them, as Godots fundamentally are dynamic obstructions. Thus, we implemented a field of view for the Godots, so when players or the catalyst enters it, it begins chasing after them.

As for the battle between players and Azara, our plan for this boss level is sectioning this level into phases where Azara's attacks progressively get more intense. First, Azara will fire lasers that take time to charge up, and players can identify its angle of fire. This will be followed by a homing missile stage that players need to dodge, then a combination of the two. When Azara becomes enraged, the speed at which he fires his weapons, along with the quantity of it, also increases, creating a climax for the intensity ramp.



Reward Schedules

What reward schedules, among the four combined types, are used?

Our game mainly uses the Fixed Ratio reward schedule. These rewards include:

- Level progression
- Narrative progression
- Satisfaction after puzzle completion
- Safe progression of catalyst across map

For our fixed ratio schedule, we use negative reinforcement to reward the players. This was implemented with the puzzling aspect of our game - by figuring out the puzzles, players unlock and get pass obstructions until they finally reached the level's end point. This then unlocks the next level for players to continue with.

We have also implemented a narrative reward for players as they complete key completion points of our game. Progression is rewarded via context and narrative regarding the next level. We also introduce new game assets to the players as they continue through the game.

How they are laid out on the basis of per action or set of actions, and per second, minute, segment (area), level, per act etc. ?

Our fixed ratio schedule is rewarded based on a set of actions. Players are rewarded when they solve a puzzle, as they unlock another section of the map, and within a level there may be multiple puzzles to solve. Hence, as players complete puzzles in a segment of the map, they will be capable of advancing through the map (although still within the level). When doing so, players will be rewarded with satisfaction, especially for puzzles that require strictly cooperative play. Additionally, since the game hinges upon cooperative play, it is required that each player advance through puzzles in the same line of progression.

Although one player completes a puzzle and advances, they must wait for their partner in order to continue to further levels in the game, thus, the rewards are dependant upon cooperative play. There are also times where one player must help out the other for the two to continue on. In doing so, they must work together to reach the level's end point, and the reward is still dependent on cooperative play. Rewards are laid out in a specified and linear manner, but are very much dependent on cooperation.

Gameplay teaching component(s)

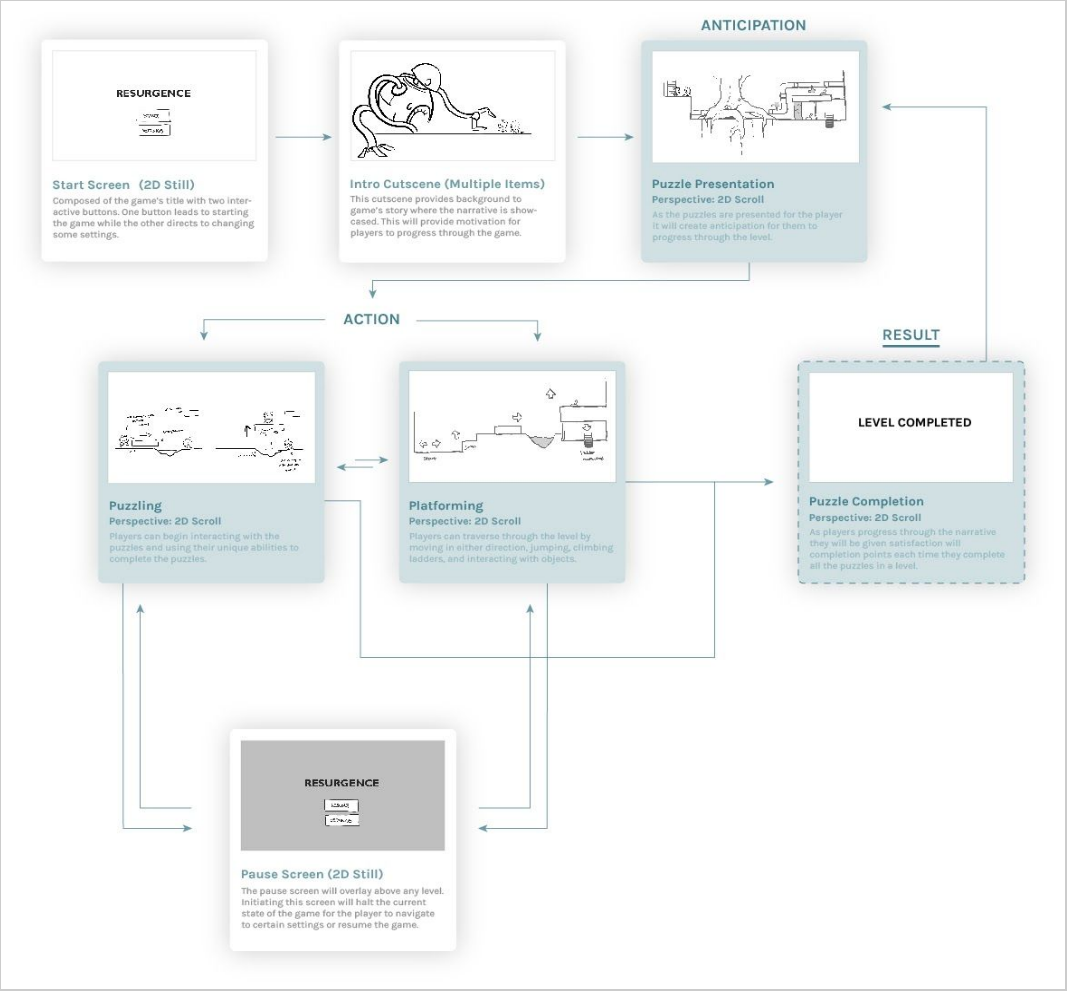
What type(s) of the four techniques as introduced in lecture has been used for teaching gameplay?

For our game, we first introduce our players to the mechanics through an overt formal tutorial. This tutorial educates the players (in a very basic way) the fundamentals of the game and how progression works. As players complete the tutorial, the following levels are designed to teach based on players experimenting. During these levels, our intention was to introduce game elements at an easier scale where players self-learn abilities and identify opposition mechanics. In the later stages of the game, we stop guiding the players as much and assume they understand the attributes of opposition mechanics

How effective it works for guiding player to learn the mechanics and apply for gameplay, issues that have been encountered, and resolution next

The tutorial works effectively in guiding the player to learn the mechanics and apply for gameplay. The players are educated gradually throughout the gameplay as they have opportunities to learn basic mechanics in the form of tutorial levels. The inclusion of formal tutorial levels at the onset focus on introducing specific mechanics. The first tutorial level for example, provides players with an introduction to movement and jumping. Additionally, each character gets a chance to learn one of their abilities. The second tutorial level later takes this further by introducing the player to their second ability as well as the catalyst. Moving forward from the tutorials to the actual levels themselves, players still have a natural progression of gameplay. For example, the first level introduces enemies and zappers. Overall this form of guiding the player works effectively in the game. However, there are at times some issues that arise. One primary issue with this that we encountered was that some players found the initial levels much too easy, while others found it easy, yet somewhat difficult in identifying the correct use of buttons. Thus, to remedy this, we ensured that button mapping for each individual character was present in the tutorial levels.

Refined Functional Flow



Start Screen (2D Still)
Composed of the game's title with two interactive buttons. One button leads to starting the game while the other directs to changing some settings.

Intro Cutscene (Multiple Items)
This cutscene provides background to game's story where the narrative is showcased. This will provide motivation for players to progress through the game.

Puzzle Presentation
Perspective: 2D Scroll
As the puzzles are presented for the player it will create anticipation for them to progress through the level.

Puzzling
Perspective: 2D Scroll
Players can begin interacting with the puzzles and using their unique abilities to complete the puzzles.

Platforming
Perspective: 2D Scroll
Players can traverse through the level by moving in either direction, jumping, climbing ladders, and interacting with objects.

Puzzle Completion
Perspective: 2D Scroll
As players progress through the narrative they will be given satisfaction will completion points each time they complete all the puzzles in a level.

Pause Screen (2D Still)
The pause screen will overlay above any level. Initiating this screen will halt the current state of the game for the player to navigate to certain settings or resume the game.

1. **Start Screen**
    a. Title
    b. Play
    c. image/concept art
2. **Intro Cutscene (Non-Interactive Mode)**
    a. Explains backstory
    b. Gives players motivation
3. **Primary Gameplay mode: Puzzling**
    a. *Sub-modes:*
        i. Combat
        ii. Logic Challenges
    b. *Player inputs:*
        i. Geyser

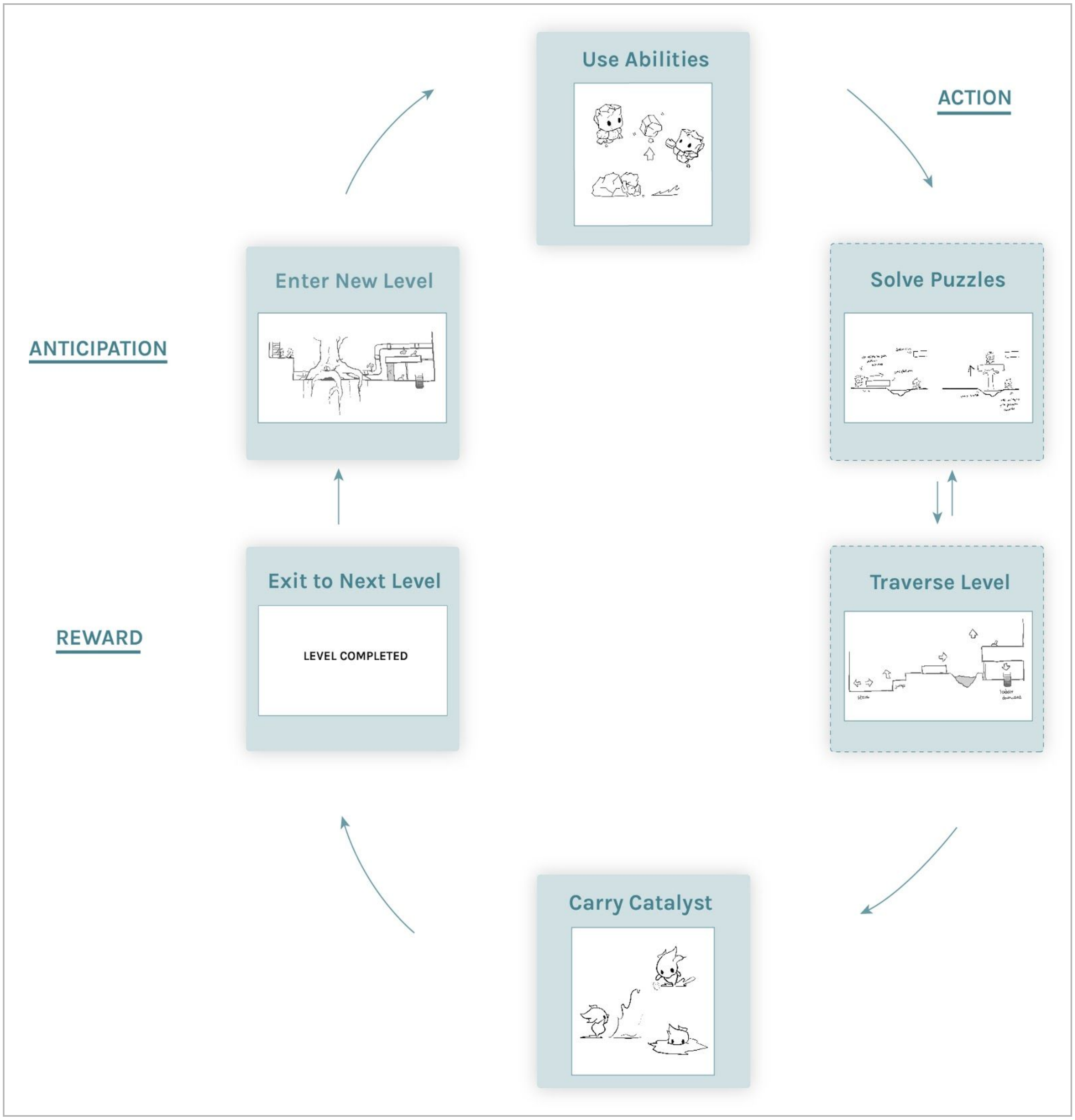
        ii. Dive
        iii. Transpose
        iv. Fissure

4. **Primary Gameplay mode: Platforming**
    a. *Sub-modes:*
        i. Timing Challenge
        ii. Level Progression
    b. *Player inputs:*
        i. Jumping
        ii. Throwing
        iii. Movement
5. **Progression Cutscene (non-interactive mode)**
    a. Narrative based
    b. Player progression
6. **Shell menus**
    a. Pause
    b. Volume

Refined Core Loop

The main core loop is between puzzling and platforming. The players will enter a level > solve puzzles using abilities > get to the end of the level with the catalyst > repeat.

Following Donald Norman's retention rules, players will instinctively want to get a feel for the character by jumping around and bumping into each other satisfying the visceral component. By learning properties of each character's abilities, players will be able to progress through the level while gaining knowledge to tackle harder levels, satisfying the behavioral component. The variation of the level design and different uses of character mechanics will draw the player to keep playing as a hook, satisfying the reflective component.



Use Abilities

ACTION

Solve Puzzles

Traverse Level

Carry Catalyst

REWARD

Exit to Next Level

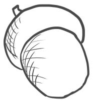
LEVEL COMPLETED

ANTICIPATION

Enter New Level

The Winning Play

- The winning play is to bring the catalyst passed the obstacles within the level and bring it to exit door with both characters



Refined Smart Depth

- There are three layers of smart depth on top of our core gameplay loop
- **Journal collectables:** small collectables scattered in extra difficult areas of the level. These will add extra story elements giving hints to what the captured characters are thinking

- **Combined Obstacles:** obstacles that restrict the player from progressing will be combined for added difficulty and cooperation between players
- **Open Endedness:** by using physics based abilities, puzzles may potentially have multiple solutions tied together with the journals
- Journal collectables do not necessarily lead to the winning play as they just give the player more insight to the story. The players must however, learn from previous obstacles and combine efforts into the winning play.

Game Concept and Narrative Description:

Concept:

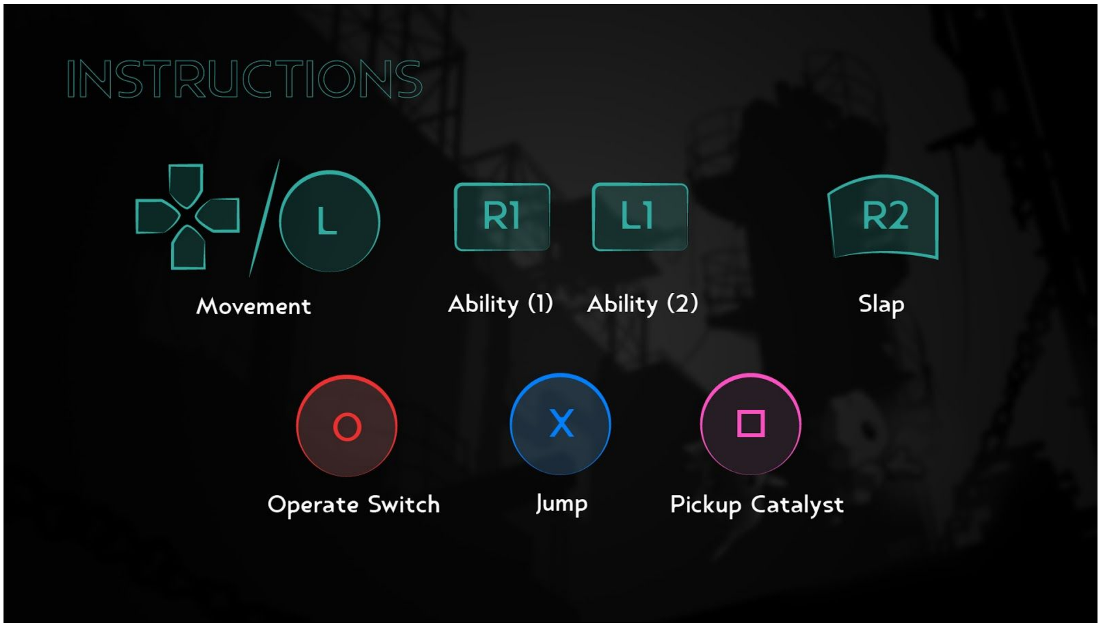
Resurgence is a 2D platformer in a fantasy setting where industrialization has razed the world. Using powers over earth and water in a collaborative gameplay environment, players use unique themed gameplay mechanics to complete challenging puzzles. While learning mechanics along the way, players complete puzzles to progress through the levels, in order to plant the Catalyst and restore the world to its natural state. As a puzzler first and a platformer second, *Resurgence* offers a distinct environment and a relevant theme.

Narrative:

Thousands of years ago, all elementals lived dividedly, but in peace. That was until one day, the fire elementals had discovered a hidden black stone entrapped deep within the Mines of Ignatius. This lone robot, Azara, was granted freedom, and with it, he slowly corrupted the world with industrialization and pollution. Azara's reign has gone long enough, as his rule is destroying this world. Itzli and Tlaloc's last hope lies in the catalyst and its power to restore nature's balance. Find the catalyst and plant it in the Tree of Mystery.

Game Instructions:

Control scheme:



- ***Movement***: directional pad + left joystick
- ***Individual abilities***: R1 + L1
- ***Slapping***: R2
- ***Jump***: X-key
- ***Pickup/Drop catalyst***: Square key
- ***Switch presses***: O-key

Objective and winning/losing conditions:

The objective of *Resurgence* is to defeat Azara and plant the catalyst in order to restore balance to their world. To win, players must solve environmental puzzles and bring the catalyst with them to each end point until the end of the game. In terms of losing conditions, we did not implement one into our game because it is meant to have a casual gameplay. When players are defeated, or when the catalyst is destroyed, then players restart and re-attempt their current level.

Variations of gameplay for each level:

The variation of gameplay is visibly seen through the puzzles and platform layout. The goals for each level remains the same, but it is up to the player to discover how to move about and solve the puzzles in a cooperative manner to complete the level. When players trigger Azara's entrance, their goal becomes interrupted by this final hurdle that they must defeat, and experiences a gameplay that differs from the prior levels. Rather than solving puzzles, players are now required to utilize the mechanics to defeat Azara, then plant the Catalyst.

Death/fail in-between a level:

At any point in the game, if the catalyst is destroyed, or if the players are defeated by environmental opposition, then the level resets and players must start the level over. Levels are setup so that each level's beginning acts as a saved checkpoint.

Implemented Features:

Based on the wish list provided in the first week of the course, we were able to implement many of the desired features.

1. **Sophisticated Animations**

   Throughout our game we ensured that there are animations implemented. These animations are done to help provide an additional layer of visual feedback, and create an association between character movement and inputs.

   Specifically, we incorporated animations for each character. These animations appear as players move around, run, jump, or interact with objects. The animations that appear on each character are also characteristic to the character. Hence, the water character will have fluid like animations, while the rock character's animations are more rigid.

   Another area we implemented animations was in the interactable objects that appear in the game. There are numerous interactable objects in the game that have animations. Switches flip back and forth to indicate whether it is activated or not. The catalyst is animated along with characters when picked up. Additionally, the lasers that appear throughout the game are animated. The water from the geyser animates upwards as the water character interacts with it. The water character is animated as it steps into pipes. The platforms will move in an animated manner as the earth character activates their fissure ability. Furthermore, the platforms are also animated as the earth character uses their transpose ability to move platforms.

The enemies are also animated in a unique manner. The godots are animated based on movement. They also become angry, appear as a missile, and explode. Azara, the boss villain, is animated as he changes states when he is idle, has full or half health, or is damaged.

2. AI Features

The game also implements artificial intelligence features throughout the game. One instance is with the godots. The godot enemies will rapidly move toward characters based on the character's proximity to the godot. Initially the godot will be animated showing that it is alerted, then immediately the godot will rush toward the direction the characters are in. The boss, Azara also utilizes AI features. Specifically, a finite state machine is used to control the boss' movements and actions.

3. Collaboration

Our game employs the use of collaboration on a variety of aspects. The entire game utilizes collaboration in the sense that both players must work together to play against the NPCs. Hence, players will utilize players vs NPC. Player vs player also occurs in the game. This takes place as players have the ability to slap one another and hinder each other's progress. NPC vs NPC also appears in the game as Azara utilizes missiles with godots attached to it. Thus, godots lives are sacrificed in attempts to attack incoming players.

4. Narrative Plots

Resurgence also employs unique narrative plots throughout the game. The game itself follows a linear progression in terms of narrative. However, the way in which the linear narrative is revealed can vary based on the sequence of actions that players take. Each aspect of the narrative is revealed throughout the game in the form of cut-scenes.

5. Diegetic UI

There are several forms of user interface feedback that appears both in the game world and narrative. Diegetic user interface occurs specifically through enemies. Godots, for example, become enraged and indicate through their appearance for the user that they are incoming. Additionally, the boss has a variety in lighting based on its health. Overall, these forms of diegetic UI help the player to understand information as they play the game.

Process Reflection:

Throughout the process of designing and developing *Resurgence,* we learned a lot of valuable information pertaining to game design. Specifically, we learned how to ease our game development process. Part of it was using prefabs, as we could drag and drop assets into the game directly. Thus, we didn't have to make new versions or instance of our each time we wanted to add a variation to gameplay, allowing us to refine and test more iterations. Within Unity, we also utilized animator components extensively. Since we used it so much, we became accustomed to it and learned how useful it could be.

Furthermore, we learned a lot about the interaction design principles. Implementing diegetic user interfaces as part of gameplay proved to be a compelling way to provide visual feedback to the player. This technique was useful to us, and is something we hope to use in future projects. This also worked hand-in-hand with some interaction design principles we learned. From this we saw how important it is.

Regarding the art implementation itself, we learned how to create a spritesheet to display all animated parts of characters. This helped to keep concise files and organized assets.

Finally, one of the most important things we learned throughout the entire course was playtesting. Each time we playtested using a variety of participants, we gained crucial feedback that could help to improve our game. Hence, we decided that this is something that should be mandatory during the game development process.