



LRW GDD

▼ Links to relevant docs:

-  [Game Design Brief](#)
-  [Retrospectives](#)
-  [Player Feedback Report](#)

Gameplay Overview

Left Right Wrong (LRW) is a 2D puzzle-platformer played by two players, attempting to complete a level by placing all required cubes into Checkpoints. Player One controls the Red Character, and Player Two controls the Blue Character. The players can move left or right, pick up objects, and Jump. They also have a limited degree of Gravity Manipulation (core mechanic).

Puzzles in LRW are designed to offer a range of creative solutions, in an effort to encourage experimentation and communication.

Developer Goals

Though the goal of this project is to create working prototype of this game, there are several personal goals I'd like to achieve as a developer.

- Create a polished and fully playable experience in 8 weeks, that would be ready for a beta launch on itch.io.
- Build puzzles that foster creative problem solving, through the use of modular systems leveraging the games core mechanic.
- Design mechanics focused on a level of interaction and combination with one another, in a system that feels expansive but not oversaturated.

Player Goals

Players in LRW attempt to work together to complete levels by using their gravity manipulation abilities in tandem with one another to safely transport Cubes to their respective checkpoints. Within a level there are also various items and hazards that either alter the way objects are effected by gravity or pose a threat that must be avoided.

The players can solve most of the problems in the game by synergizing their abilities to manipulate the gravitational pull on an object in the correct order, safely bringing it to the goal. The key concepts are communication, synergy, and patience

Mechanics

LRW is a game centered on teamwork and synergy. The mechanics deliberately require players to work together, while providing an equal opportunity for players to get in each other's way.

Movement

Direction Movement Type	Units/Second (Unencumbered)	Units/Second (Encumbered)
Left/Right	300	200
Jump	400	300

A player is considered "unencumbered", by default. If a player grabs or carries an object, they are considered "encumbered". This decreases their movement speed and jump height, as referenced in the table above.

The player should feel as though they have a good amount of air control while jumping, the exact numbers will be tested. In general, the player should have control over their momentum, however, if they let go of lateral input while in the air, they will not instantly come to a halt. The movement in LRW is designed to feel slow, in an effort to emphasize the importance of the gravity manipulation system.

Gravity Manipulation

The crux of gameplay is the players' ability to control **Channels of Gravity**. These Channels alter the direction of gravitational force to all objects within that Channel. Player one, the red player, controls the **Red Channel**, while player two controls the **Blue Channel**. These colors are internally referred to as the Native Colors. To enhance the emphasis on collaboration, the different colored Channels are also locked to specific directions. The Red Channel can send gravity up or down, and the Blue Channel sends objects left or right. Players can manipulate the direction of their gravity Channel via button input, though with the following restrictions in mind.

Restrictions

- Each player may only change the direction of their native gravity Channel (the original color of their character.)
- The players may only change the direction of their gravity Channel one time before touching the ground.

Tagging

Gravity Channels only effect objects within their Channel, but the players have the ability to assign Channels to characters and Grav Objects (any object with physics enabled). To assign a gravity Channel to an object, players can Tag an object with their corresponding native color. Tagging is a limited range raycast sent in the direction the player is facing. If the player hits a valid object or player, they successfully Tag that object. Tagged objects are assigned a transparent overlay matching their Tagged color, and have the gravity direction of their new Channel constantly applied.

Important Note: changing the direction of a gravity Channel applies for every Tagged object in that Channel. For example, if the blue player changed their gravity direction to Left, the blue player, along with every other actor Tagged in the blue Channel, will immediately receive that effect.

Players also have the ability to Untag objects, placing them in a Null Gravity Channel, with the default gravity direction (down). Any player can Untag any object, regardless of its gravity Channel, as long as they hit that object with ranged raycast. Players also have the ability to Untag themselves via input. If a player is Untagged, instead of moving into a null gravity Channel, they are moved to their Native Gravity Channel. Below is an example sequence of events.

Taggable objects feature a colorblind-accessible visualization component, toggleable in accessibility settings. This component simply adds a black O symbol on objects tagged blue, and an X on objects tagged red.

Movement in Alternate Gravity

In order to reduce player confusion regarding input rotation, the players movement inputs will be locked to match the orientation of a fixed camera, regardless of the orientation of the player. For example, while on the ground pressing left will move the player to the left of the screen, and pressing right will move them right. While on the ceiling, the left input will still move them to the left of the screen. While on either wall, to travel upwards, the player must press Up, and Down for down.

Grabbing/Carrying

Characters can grab and carry moveable objects. There are two types of moveable objects: Cubes and Hazardous Cubes (See Items.)

To grab an object, players must hold down an input. Releasing the input will drop the object. While holding the object, if something is wedged between the player and the object, the player will be forced to drop the object if the distance between the two exceeds a set range (Drop Radius).

While a player is carrying an object, it will overlap that player instead of colliding with it. This prevents players from abusing Cross Directional Bugs (see bugs).

Grab Stats

Grab Range	Drop Radius
95 Units	150 Units

Death

When a player or Cube comes in contact with a hazard (See Hazards), they are destroyed immediately, and respawn at their starting position automatically, after a few seconds. There are no other penalties for dying, as the game is meant to be difficult, not stressful. When players respawn, they are given a short amount of invincibility time, to prevent infinite death loops. Note: some hazards only destroy players, and don't destroy cubes. The specifics of each object will be listed in that item's description.

Death Stats

	Respawn Time	Invincibility Time
Player	0.5 Seconds	0.5 Seconds
Cube	1.5 Seconds	0 Seconds

Items

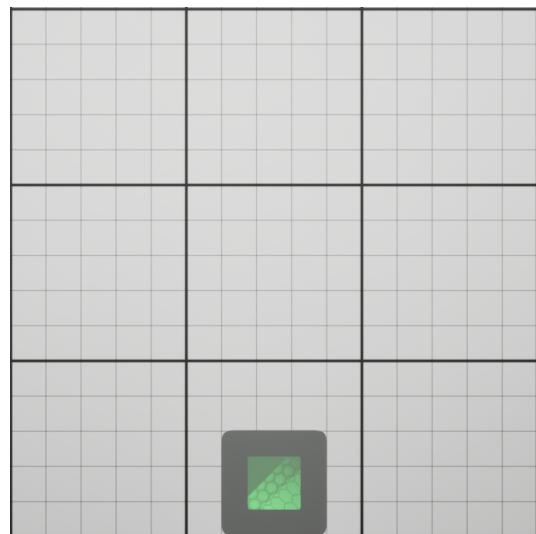
Items are the primary level design tool of LRW. Items are a critical aspect of gameplay, as they expand upon the gravity manipulation mechanic. Items are the content that make up levels. Each item has a unique behavior and asset, both of which will be expanded upon further in it's description.

Cubes

Description:

Cubes are the victory device for the game. They are **Grabbable**, and **Taggable** objects, that the player must find a way to bring to the **Checkpoints** in order to finish the level.

Size: 3×3 Units

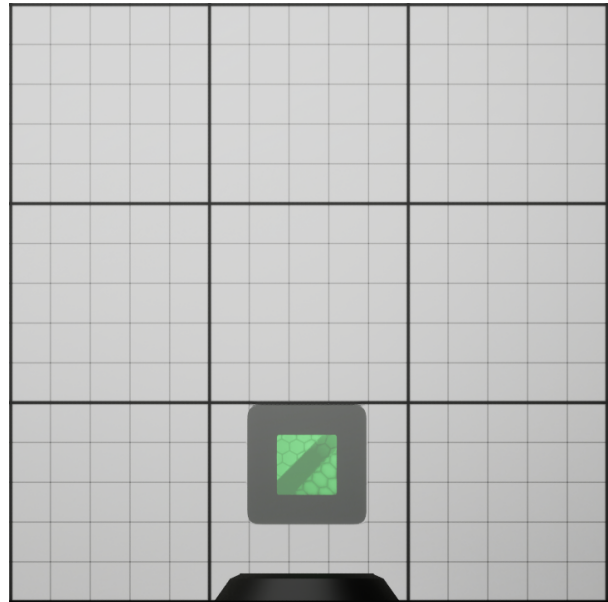


Spawners

Description:

Spawners each create 1 cube and store it in a static position until the cube it spawned (referred to as it's Child Cube) is **Tagged** or **Grabbed** by a player. If its child cube is destroyed, the spawner will generate a new one. Spawners can also be enabled to spawn a Force Bubble around its child cube.

Size: 1×5 Units

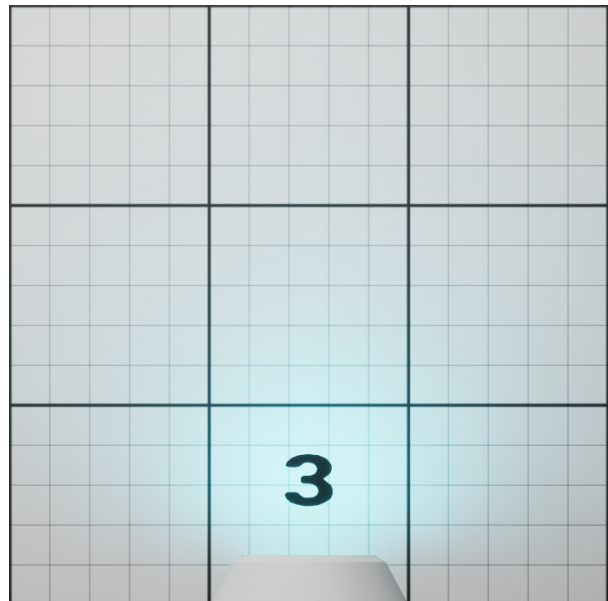


Checkpoints

Description:

Checkpoints are deposit points for cubes. They have a certain number of cubes they must receive until they are marked complete. Completing all checkpoints is the win condition for each level. When a checkpoint receives a cube, that cube is **permanently destroyed** and that cube's spawner will no longer spawn cubes. Some Checkpoints are colored orange, signifying that they work only for Hazardous Cubes. White or "default" checkpoints, do not accept Hazardous Cubes.

Size: 1.2×5 Units

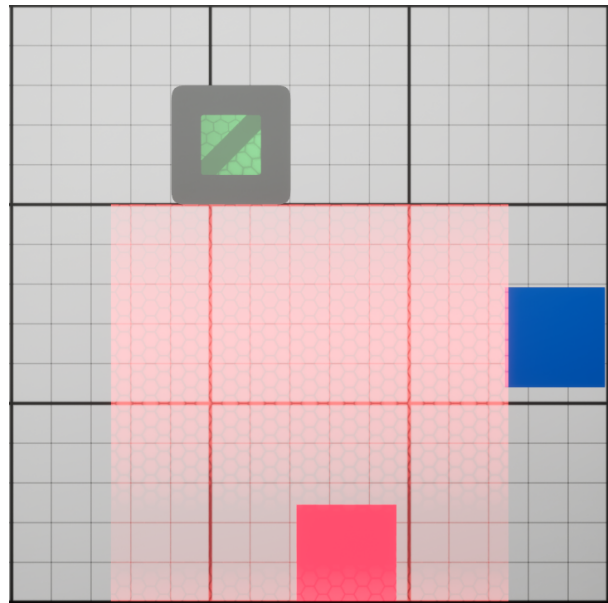


Shield Fields

Description:

Shield Fields are solid barriers only for objects that do not match their color. For example, a Red Shield Field is solid for blue objects, and has no collision against red players. **You cannot fire a tag inside or through a Shield Field, no matter your color.** This prevents objects from changing to the wrong color inside a Shield Field.

Size: Varies

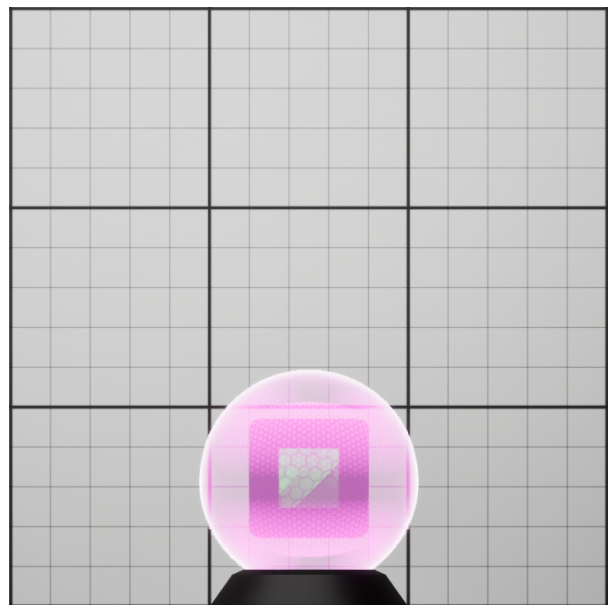


Force Bubbles

Description:

Force Bubbles are static barriers that can surround items. Force Bubbles block Tags as well as Grabs, rendering their object inaccessible to the players. Force Bubbles can be destroyed if they come into contact with a Hazardous Cube (see Hazards). If this occurs, both the Hazardous Cube and the Force Bubble are destroyed. If the Force Bubble was created from a Spawner, it will respawn if that Spawner's Child Cube is Destroyed.

Size: Varies (Surrounds An Object)

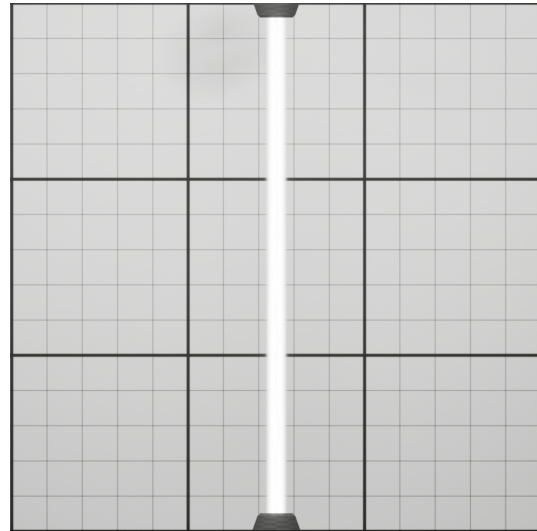


Nullifiers

Description:

Nullifier's are a type of Laser. Lasers are transparent items that objects can pass through, receiving an effect when they do so. Nullifiers cause any objects that pass through them to return to their native color. For example, if the blue player is tagged red, they will return to blue if they pass through a nullifier. Cubes and other objects always return to Null.

Size: 15×1 Units

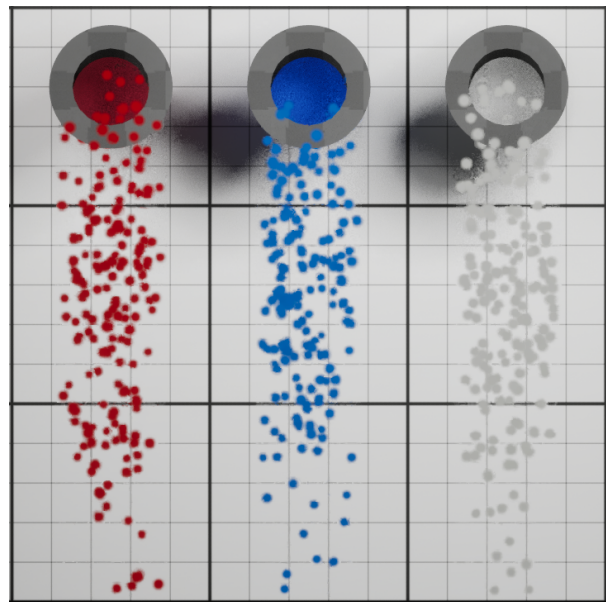


Fountains

Description:

Fountains spawn a stream of colored particles, that change the gravity channel of any object they come into contact with. Fountains can also be tagged or nullified, which changes the color of the fountain.

Size: 15×3 Units

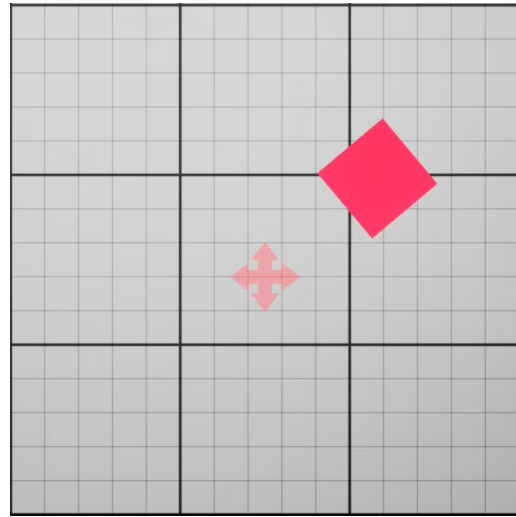


PushBack

Description:

PushBack is an item that causes player's channel direction to change to "In". This makes all objects tagged with that player's color to be pushed towards the back wall, creating a "top down" gameplay style. This effect lasts for 5 seconds, and the PushBack cannot be activated again until that timer expires.

Size: 2x2 Units

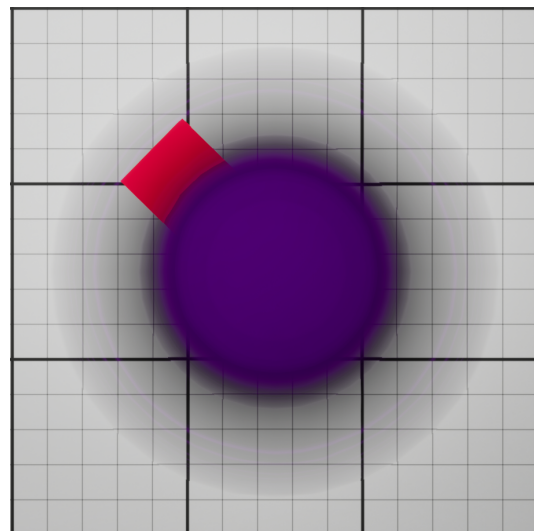


Black Hole

Description:

Black Hole's are unique object that override an objects gravity direction to point towards the center of the black hole. The center of Black Holes are solid objects, allowing players to move around them in a circle, or even jump out of them. The shaded area around the asset represents the radius of the effect.

Size: Varies



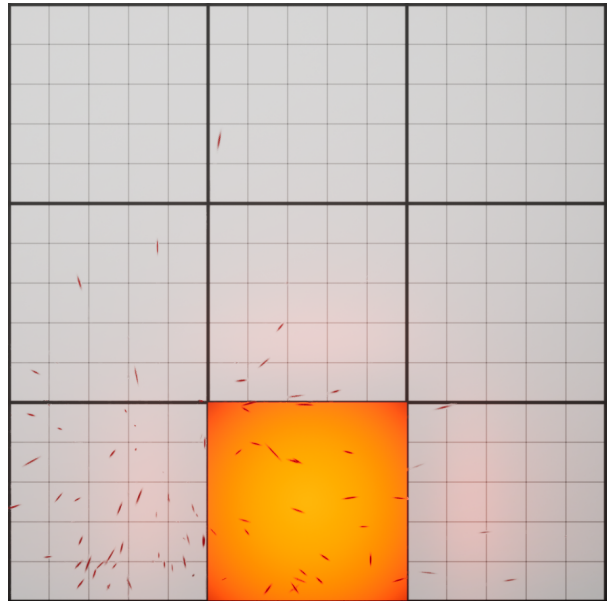
Hazards

Static Hazards

Description:

Static Hazards are barriers that apply an instance of damage when they come into contact with, instantly destroying that object.

Size: Varies

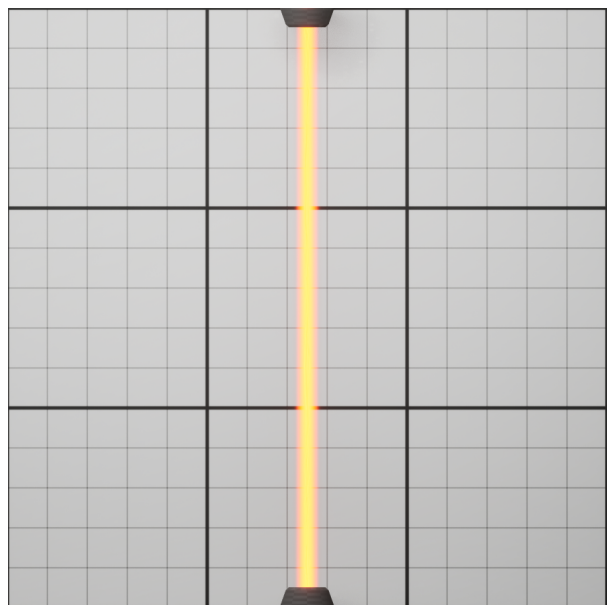


Death Lasers

Description:

Death Lasers are barriers that apply an instance of damage to players, but ignore cubes. They cannot be tagged through, nor can players grab an object on the other side of a death laser.

Size: 15×1 Units

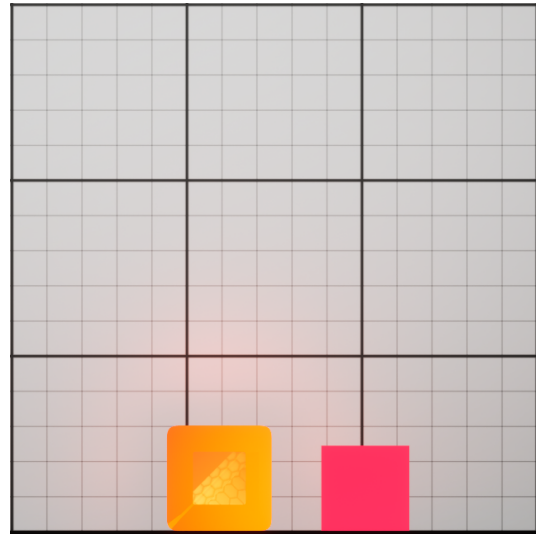


Hazardous Cubes

Description:

Hazardous Cubes are moveable hazards that players can carry. They apply damage as if they were a Static Hazard. They can be taken to orange checkpoints and they destroy force bubbles. While a player is carrying a Hazardous Cube, they cannot be damaged by it.

Size: 3×3 Units

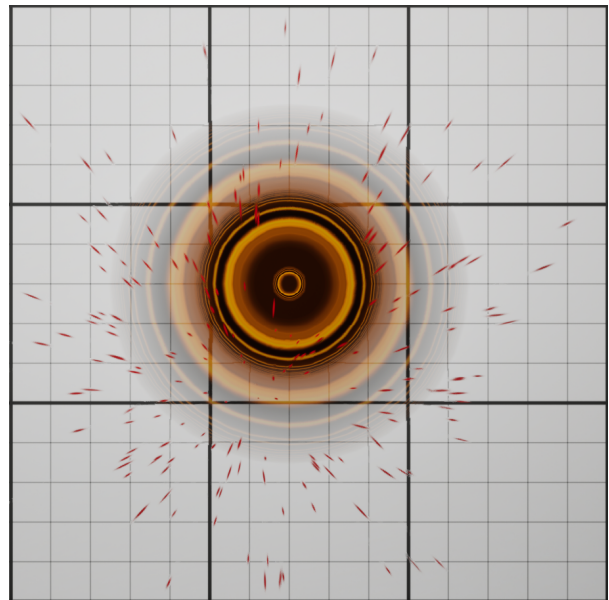


Hazardous Wells

Description:

Hazardous Wells are a type of Black Hole. However, their center is a Hazard, causing objects to receive damage if they come into contact with it.

Size: Varies



Level Design

Level Constraints

The levels in LRW will consist of a series of checkpoints that must be completed. Checkpoints contain a goal number, once they receive a number of cubes equal to their goal number, they are completed, and the level is completed. The challenge in each level is the act of bringing cubes to the checkpoints, as along the path there are a series of hazards and obstacles the players must work around using their

gravity alteration mechanics. **Both players must leverage their unique control of gravity in order to complete the level; this requires a great amount of both cooperation and communication.** The general rules for the level, or level design bible are as followed:

1. No solution to any problem will *require* the use of invincibility frames.

The i-frames are in place as a death-loop contingency, it is generally assumed by players that i-frame abuse is a form of “cheese”, and many players would not assume that abuse of this mechanic is a necessary element in the game, as it would be a “lame” solution to a puzzle.

2. Each level should contain an equal, or close to equal amount of work from both players.

The players should never feel like their character has significantly less to do with the other character. The level design will use alternating actions to ensure both players experience the same amount of engagement with the space.

3. There can be problems that require a player to die.

Some solutions to puzzles may require a player to die. Since the death stat is not tracked, and death is not a punishing event, this is acceptable.

4. Problems can contain multiple solutions.

Giving players control over a fundamental element of the space (gravity) opens the door for many interesting solutions to problem. The level design will not snuff out that experience by removing all alternative solutions as they are discovered.

5. Levels will not contain “red herring” elements.

Every level will only contain items that are part of a solution to a problem. Players may discover alternate routes that remove the need for some element, but no element will be placed in a level for the intended purpose of “tricking” a player into thinking it is necessary. This is both lame puzzle design, and wasteful of computer resources.

Level Contents

Currently, there are 15 levels planned for LRW, each offering a unique solution leveraging the different mechanics. The table below contains the order in which the elements appear, and how they are utilized as the **Focus** of that level. Each level also contains a name, which will appear at the beginning of the level, this may be a fun flavor element, or in some cases offer a subtle hint to the solution.

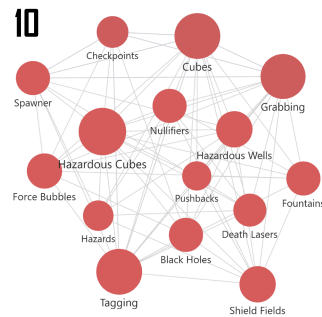
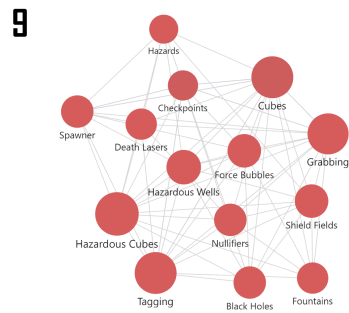
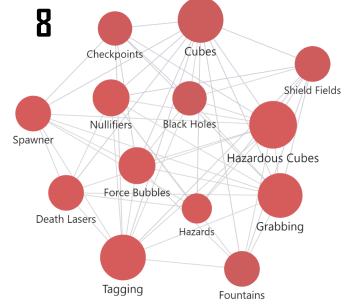
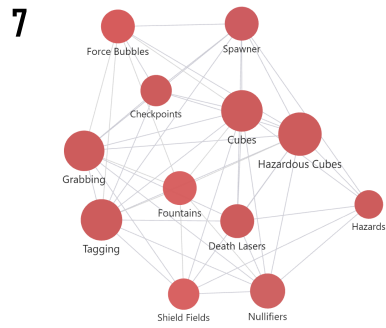
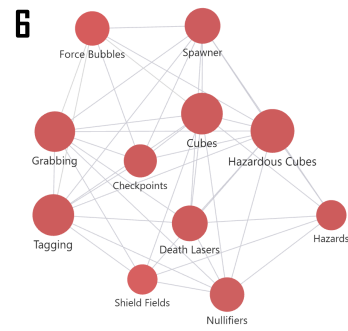
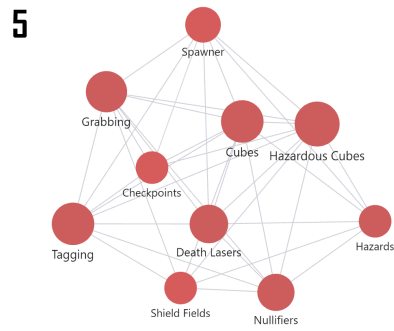
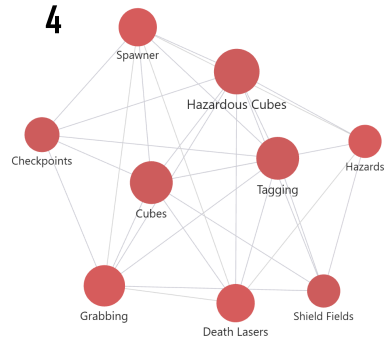
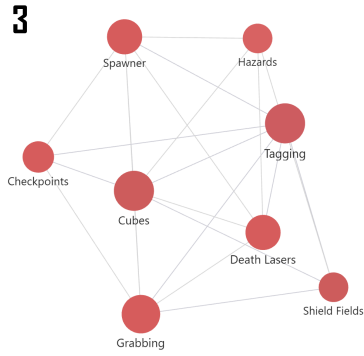
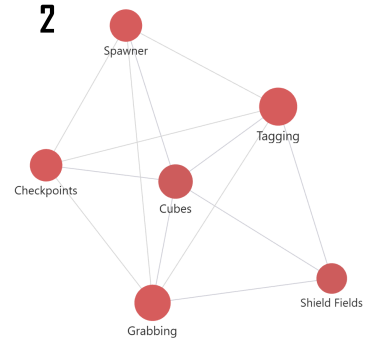
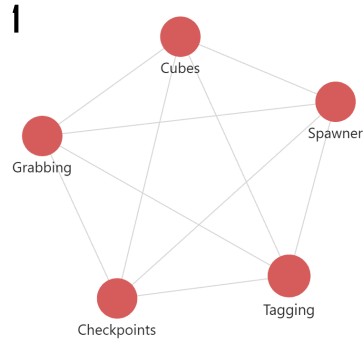
Tentative Progression System

Level #	Level Name	Focus	New Items	Difficulty	Status
1	Genesis of the Grand Finale	Grabbing Objects	Cubes, Spawners, Checkpoints	Easy	Complete
2	Energy is Born	Tagging Objects	Shield Fields	Easy	Complete
3	You're It!	Hazards	Static Hazards	Easy	Complete

Level #	Level Name	Focus	New Items	Difficulty	Status
4	Unprecedented Resilience	Hazards ii	Lasers	Easy	Complete
5	A Neutral Force	Nullification	Nullifiers, Moveable Hazards	Easy	Complete
6	Around the World	Hazards iii	None	Medium	Complete
7	Unstoppable Force // Immovable Object	Force Bubbles	Force Bubbles	Medium	Complete
8	Birth of New Matter	Fountains	Fountains	Medium	Complete
9	Grounded	Hyper Switching	None	Medium	Complete
10	Touch Water	Fountains x Shield Fields	None	Hard	Completed
11	Around Around the World	Black Holes	Black Holes	Medium	Completed
12	Objects in Motion	Hazardous Wells	Hazardous Wells	Medium	Completed
13	You and Me	Black Holes x Shield Fields	None	Hard	Completed
14	The Pushback	Pushbacks	Pushbacks	Hard	Completed
15	Grand Finale	Pushbacks x Tagging Objects	None	Hard	Completed

Progression System Visualized

Below is the lateral progression of mechanics in LRW, visualized by webs in which connecting nodes represent mechanics that interact or build off of one another. This data was used to emphasize the rapid growing complexity of the in-game puzzles, and helped determine when to space information out with levels that do not introduce new concepts.



Characters

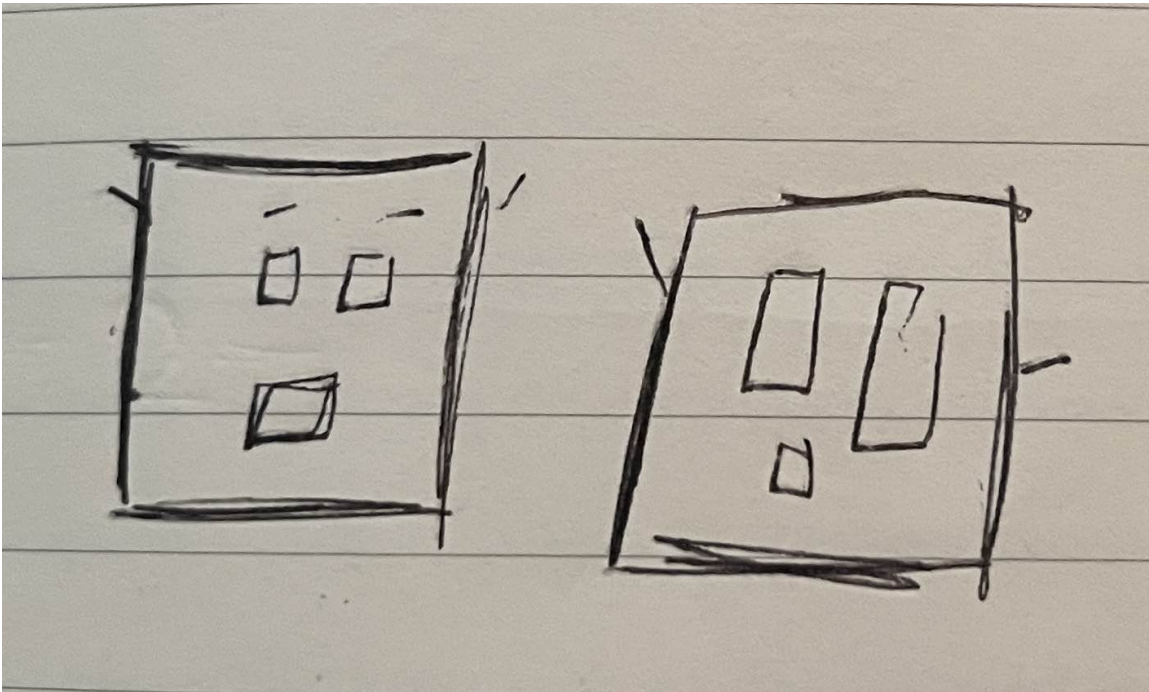
There are two characters directly in the game, the blue cube and the red cube respectively. There is also another mysterious character that is referenced in the games lore, referred to as the Origin.

The cubes are silent protagonist, and their personalities and traits are unconfirmed. It is only confirmed that they are friends.

Design

The characters in LRW are cubic objects with a small directional arrow near the front. This helps the players easily understand their orientation. Though the characters are full cubes, their hitboxes are two dimensional, which allows them to make tight turns without clipping their corners against objects.

Lofi Concept Art



What did you expect? They're cubes.

Story

The story players are able to put together by reading in game messages is intentionally vague and means almost nothing as an intentional design decision. However, these in game messages hint at an incredibly deep story, taking place at the end of the universe and the beginning of a new one. This decision was made because I thought it would be funny.

Structure

Some levels contain messages in the form of static text renders. Below is a table that describes each message and the level in which they appear.

Message	Level#
[Once] there [was] two cubes, and they [were] friends.	6
The laws [were yet to be] determined.	7
As the ripples crashed across the sky, the Origin calls into the vastness.	8
There [was] no response.	
The cubes developed an instinct for their great purpose.	
The Origin was pleased, and the sky became viole(n?)t.	10
You see, Cubes do not have the capacity to evaluate ethics, though nor does the one who wrote them.	11
What is ethical is what Will happen, and what won't is not.	
And so, the Cubes carried on.	
As the End drew near, the Beginning closes in.	12
See what I did there? Of course not.	
All began (to fall)	13
The Cubes looked around across The Vastness The Origin smiled,	15

Message	Level#
as they faded into the purple	

Bugs

Bugs Table

Aa Type	≡ Description	📉 Priority	☀ Status
<u>Collision Issues on Physics Meshes</u>	Physics meshes have incorrect collisions and can be pushed into the walls. They also build up speed quickly and launch the player.	High	Bug Fixed
<u>Carrying Objects Out of Bounds</u>	Players can pick up objects, then drop them mid-rotation, allowing them to be dropped out of the level.	High	Bug Fixed
<u>Untagging Issue</u>	Untagging players sends them to the null tag instead of their native color	High	Bug Fixed
<u>Respawning Issue</u>	Respawning while tagged does not clear that player's overlay	High	Bug Fixed
<u>Respawning Issue</u>	Respawning does not reset players collision channel	High	Bug Fixed
<u>Infinite Death Loop</u>	Players can place a moveable hazard onto the spawn point	High	Bug Fixed
<u>Out of Bounds</u>	Players can perform reliable out of bounds glitches on the ceiling.	High	Bug Fixed
<u>Grabbing Items</u>	Both Players can carry the same object, causing it to interpolate between both players positions	Medium	Bug Fixed
<u>Cube Respawning</u>	Cubes destroyed by checkpoints respawn.	Medium	Bug Fixed
<u>Particle Collision</u>	Objects that block tags do not block the particle	Low	Bug Fixed
<u>Black Hole Orientation</u>	Black Holes create orientation issues while holding Cubes	High	Bug Fixed
<u>Grabbing incorrect objects</u>	players can grab any object that is moveable	High	Bug Fixed
<u>Invincibility frames</u>	If a player is touching a moveable hazard while their i-frames expire, their overlap check returns null.	Medium	Bug Fixed
<u>Invincibility frames</u>	Players can abuse i-frames to pass through death lasers	Medium	Bug Fixed
<u>Carrying Hazards</u>	Hazards being carried by the player, do not apply damage to the other player	Low	Bug Fixed
<u>Carrying Objects Clip</u>	Objects Carried by the player are able to clip through some barriers when the player quickly turns	Medium	Bug Fixed

Aa Type	☰ Description	📍 Priority	⚙ Status
<u>Spawner Object Location</u>	Spawners spawn objects at a height that barely misses the players grab trace	Medium	Bug Fixed
<u>Hazardous Gravity Wells improper destruction</u>	Hazardous Gravity Wells do not destroy hazardous cubes	High	Bug Fixed
<u>Player hitboxes clip corners</u>	Because the players hitboxes are 3D, their corners clip the edge of hazards when the player rotates in certain instances	Medium	Bug Fixed

High Priority Bug Types

This is a physics puzzle game. Thus, there is a ton of opportunity for bugs or interesting actions between items. Since I am one person, and this is a 8 week project, the likely that the finished project will be 100% bug free is very low. However, there are certain types of bugs considered to be high priority, despite still offering a playable experience.

Cross Directional Bugs

Cross-directional bugs are one of the main challenges in the design of LRW. These refer to instances where players interact with an object or player of which has a gravity force moving in the opposite direction. In certain circumstances, this can cause player's to move in unintended ways, such as infinite jump glitches, or massive acceleration boosts.

Desyncs

There should never be an instance where a player is able to desync from their gravity channel. This means that an untagged player's gravity is moving in one direction, but objects tagged in their color are moving in another direction, in other words, the player has separated from their gravity channel. (The only exceptions for desyncs are black holes / hazardous wells, as that is their intended function).

I-Frame Abuse

Simply put, I-Frame Abuse scenarios are when the player is able to leverage the short invincibility they gain when they respawn as a way to "cheat" in puzzles.

Audio

The general vibe of the audio in LRW is soft. For example, objects should make deep but soft thuds when they hit the ground. Below is a table that describes the different sound effects needed for the game, as well as a short video clip of the game mechanic that should be paired with that sound.

AudioTable

Aa Name	📄 Description	🎥 Video	⚙ Status
<u>Movement</u>	Simple repetitive "dooty" noise. Should be subtle.	<u>Move.mp4</u>	In progress

Aa Name	Description	Video	Status
<u>Grab</u>	Player grabs an object	<u>Grab.mp4</u>	Done
<u>Jump</u>	The jump is pretty floaty. Maybe something deep. Deep vocalization might be funny?	<u>Jump.mp4</u>	Done
<u>Tag</u>	This is a projectile the player fires. Something like a laser gun or a short firework burst. There should be a sound for the tag being fired, and one for the tag impacting an object. Something that represents a new effect being applied.	<u>Tag.mp4</u>	Done
<u>Untag</u>	This is also a projectile, but it reverses the effect of the Tag, a reversed sound effect would be cool	<u>Untag.mp4</u>	Done
<u>Untag_Self</u>	This removes a tag placed on a player. Something cleansing or refreshing, non-vocalized	<u>Untag_Self.mp4</u>	Done
<u>Grav Switch</u>	You know those blastic tubes that make that UUUAAHHHP noise when you flip them? Both versions of that.	<u>Gravity_Switch.mp4</u>	Done
<u>Death</u>	The player explodes into glass-like shards. Something like the death sound in celeste	<u>Death.mp4</u>	Done
<u>Fountain</u>	Fountains making a subtle pitter patter on the ground	<u>Fountain.mp4</u>	Done
<u>Force Bubble Destroyed</u>	A popping sound, with electric undertones.	<u>ForceBubble_Destroyed.mp4</u>	Done
<u>Force Bubble Bounce</u>	Objects interact with a forcefield and bounce off with an electric ZAP		Done
<u>Nullify</u>	See Untag	<u>Nullify.mp4</u>	Done
<u>Black Hole</u>	Player gets sucked into a black hole. Something forceful but soft	<u>BlackHole.mp4</u>	In progress
<u>PushBack</u>	This is a powerful item that temporarily alters the gravity of the user for a limited time. there should be an effect for it being applied, and one for it expiring	<u>PushBack.mp4</u>	In progress
<u>Hazardous Wells</u>	like the black hole, but dangerous.	<u>Hazardous Well.mp4</u>	In progress
<u>object lands on ground</u>	deep, but soft impact noise		Done

Music

LRW needs about 3-5 different tracks, including the title theme.

In general, the music should increase in complexity throughout the game, but not in intensity. The music should be gentle and slow, rarely picking up in volume.

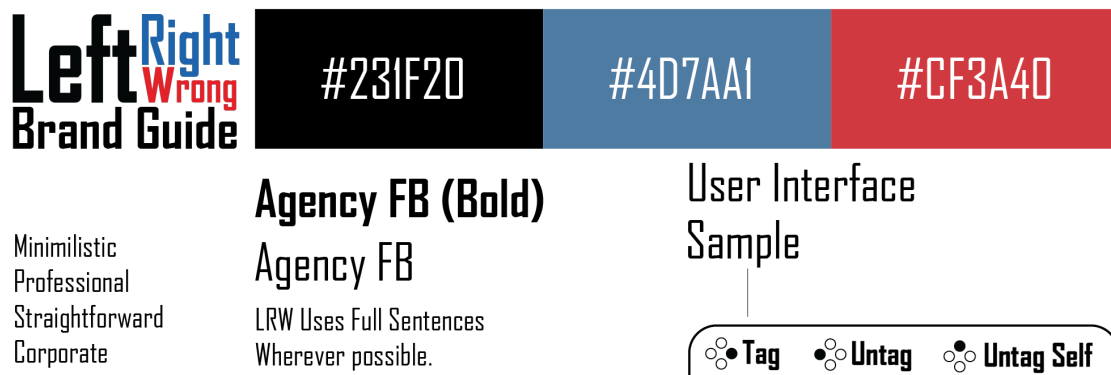
Ideally, the main melody of the game slips into a minor key.

UI

LRW uses a minimalistic UI setup, in which information is presented through direct instruction, often in full sentences. The full guide for UI design can be viewed below.

The gameplay is not constructed to give many hints to the players, but there is a simple toggleable graphic at the bottom of the screen, reminding players of the game's basic control scheme. Additionally, there will be a small spatial UI graphic that appears above a player's head if they enter the grab range of a carryable object, notifying players that they may grab that object. There will never be a spatialized object notifying a player that they may tag an object, because this would both cause visual clutter, and defeat the purpose of a few key discovery moments.

Guide



Prototyping

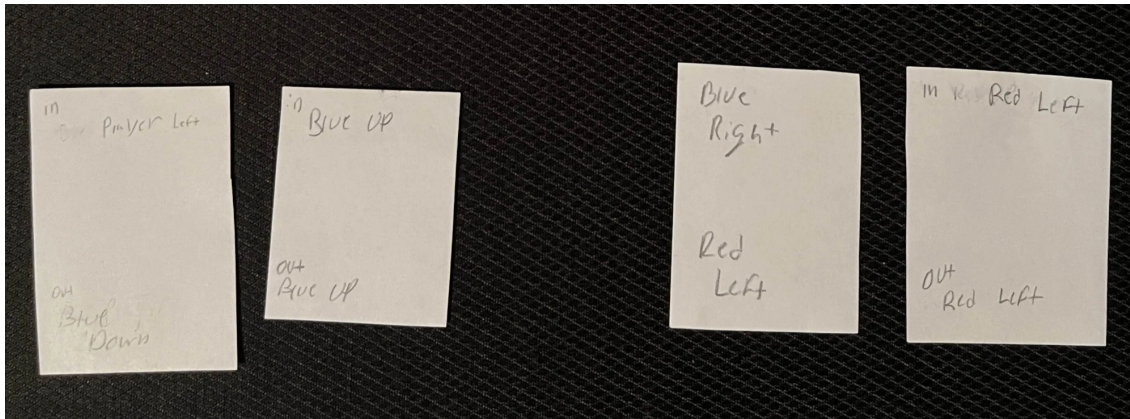
The final section outlines the game's various prototypes and playtest. Eight total prototypes were tested, including five paper prototypes and three digital. Each playtest had a unique purpose in regards to the final prototype.

Prototype 1

Purpose: to abstract and test the core loop of the game, to determine its viability.

Description: Prototype 1 was a very heavy abstraction of the initial core loop. In the initial concept for the game had players interacting with objects that changed the gravitational direction of a specific color into a specific direction. This paper prototype considered the challenge posed to players of the order in which to interact with objects.

The goal was for players to match a random set of conditions by playing cards that have a random input and output. This could be compared to a slightly more complex version of uno, in which two player cooperated to complete their hands.



Feedback and Decisions:

"I love you man, but this is a little rough." - My friend matt.

The ultimate decision made from this prototype was to push forward the amount of control the players had on the environment.

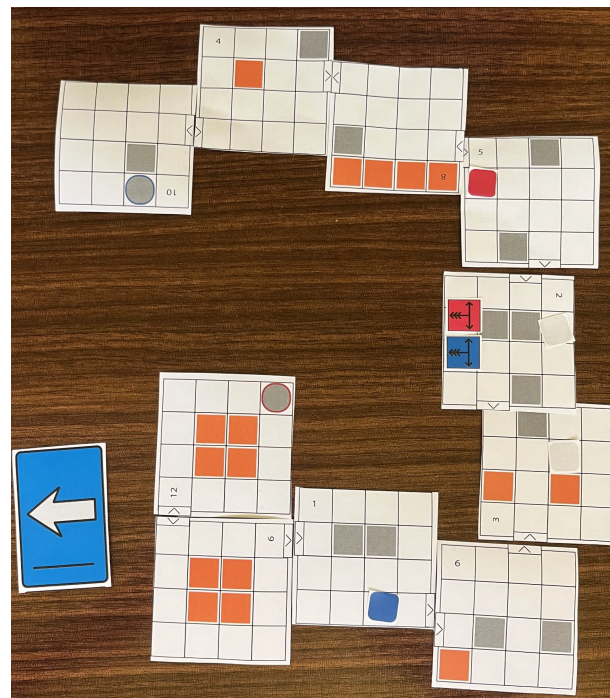
The concept became a game in which players could control the gravitational direction of objects that matched their color. Players could also change the color of objects they came into contact with.

Prototypes 2 & 3

Purpose: to refine the core loop based on feedback from the previous versions

Description: the next two iterations built upon the previous version by introducing the concept of tagging objects, and then splitting the players control of gravity in half. This was done to force the players to cooperate in order to progress, rather than allowing a single player to be able to complete a puzzle by themselves.

These prototypes were slightly higher fidelity as well, as they contained components representing the physical space in which the game took place. In order to get this to function, I had to move up a few layers of abstraction into an experience that would represent gameplay slightly more accurately.



Feedback and Decisions: The final consensus on this series of prototypes is that the newly introduced mechanics directly improve the experience.

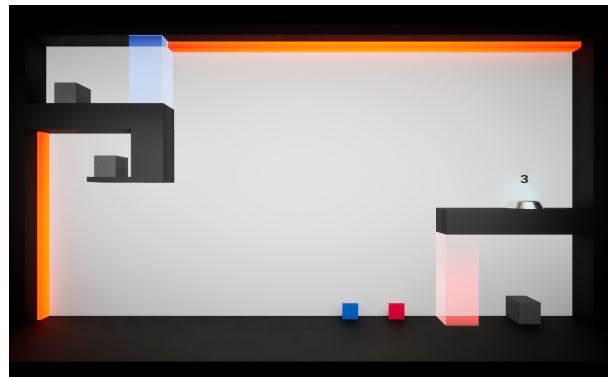
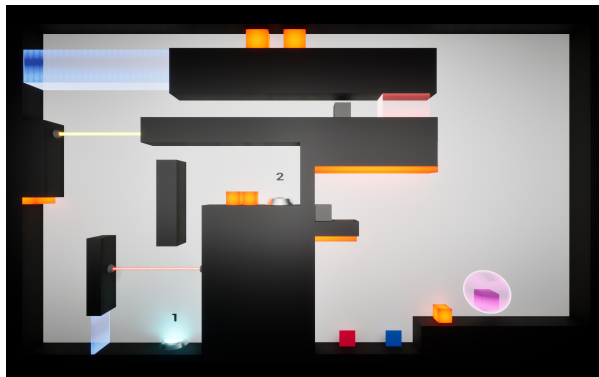
Tagging and splitting the mechanic were very crucial to the development of LRW. This is because it allows players a kit that can both be used to benefit and disrupt the efforts of the team.

Prototype 4

Purpose: to test the core loop's implementation into a digital space.

Description: The core loop prototype introduced a few elements to the game, in addition to a fully functional gravity manipulation system with local multiplayer capabilities. Shield fields, death lasers, nullifiers, moveable hazards, force bubbles, checkpoints, and static hazards were all introduced in this prototype.

The game was quite a bit harder than the final version, as cubes did not respawn, and shield fields destroyed incorrectly colored objects instead of blocking them.



Feedback and Decisions:

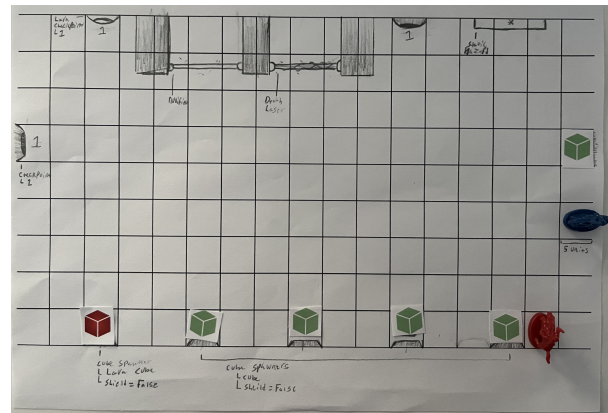
This playtests, while being broadly successful, brought forward an important shift in my design thinking. One tester mentioned that they felt the game should be "difficult, but not stressful". This really influenced my thinking moving forward.

Prototype 5

Purpose: test playground area in a paper prototype.

Description: this was the lightest playtest session of the series, as there was not much to *do* in the digital space, as much as there was for me to describe/explain. There were initially two spaces, each with a small area to display the game's mechanics and demonstrate how they work.

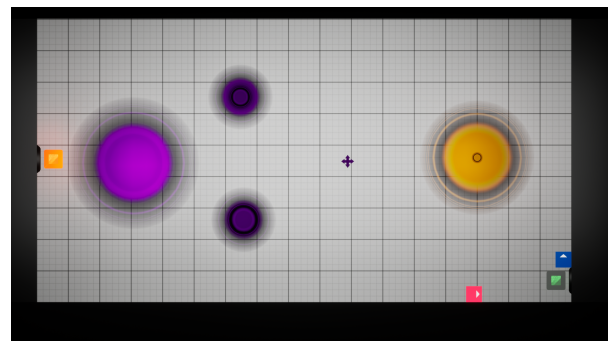
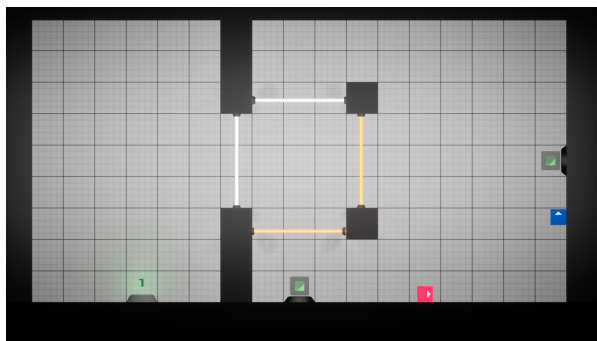
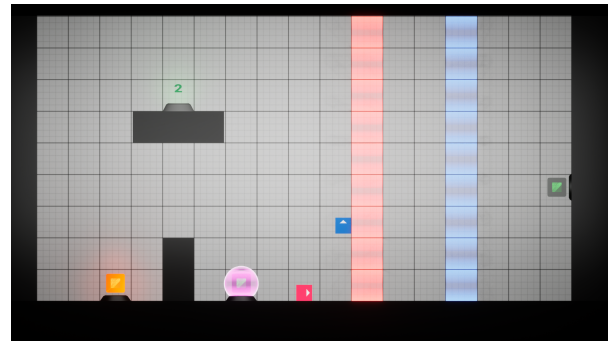
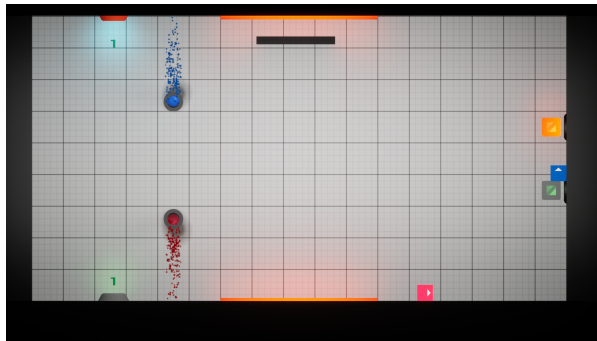
Feedback and Decisions: These areas did not contain enough space for the player to adequately experiment with the kit, so a third space was added to better distribute the existing mechanics, as well as a fourth for new mechanics such as the black holes and pushbacks.



Prototype 6

Purpose: test new playground areas in a digital space.

Description: The final version of the playground area was organized into 4 spaces, each focusing on different mechanics with a central focus. The first space (top left) focused on gravity manipulation, the second space (top right) focused on blocking devices, the third (bottom left) focused on laser gates, and the fourth (bottom right) focused on gravitational overrides.



Feedback and Decisions:

The feedback on the playground area was all positive. I did get a lot of feedback, however, on visual clarity within my gameplay elements. This led to a lot of improvements in the game's UI and visual cues. Specifics include orientation references, colorblind accessibility, and both diegetic and non-diegetic UI elements.

Prototype 7

Purpose: determine effective order of lateral progression based on feedback from previous players

Description: For this micro-test, I printed out cards of each of my mechanics, and had play testers of my prototype 6 rank each mechanic in order of complexity in their opinion.

Feedback and Decisions: The rankings of each item directly influenced the final progression system of the game (see Progression System Visualized). Below are the rankings from the players, ranked in ascending order from least complex to most complex.

1. Cubes	1. Cubes
2. Spawners	2. Spawners
3. Tagging	3. Tagging
4. Checkpoints	4. Checkpoints
5. Hazards	5. Hazards
6. Hazardous Cubes	6. Shield Fields
7. Force Bubbles	7. Fountains
8. Shield Fields	8. Hazardous Cubes
9. Nullifiers	9. Death Lasers
10. Death Lasers	10. Nullifiers
11. Fountains	11. Force Bubbles
12. Black Holes	12. Pushbacks
13. Hazardous Wells	13. Black Holes
14. Pushbacks	14. Hazardous Wells

Prototype 8

Purpose: test digital progression system, and provide final notes for polish before beta release.

Description: This test was performed for two separate groups, each played through the 15 levels I designed following the progression system. These players were given no instruction apart from what was provided by the game. I observed play carefully and took notes as they played.

Feedback and Decisions: The majority of changes to come out of prototype 8 were minor softlocks possible within levels, usually revolving around cubes making there way into unreachable areas.

There were also some minor clipping glitches that were relatively easy to iron out.

The biggest update to come from these playtests actually had more to do with how controllers are detected and assigned in the main menu backend. Riveting stuff.