



THE UNIVERSITY *of* EDINBURGH

Submission 1: Proposal

NN & MM

Astrud Elena Espinoza Ramírez, Fan Lin, Chengcheng Jiang,

Yuping Ke, Jingwen Deng, Yunqi Tang, Yutong Li

Digital Media Studio Project

February 13th 2025

Table of contents

1. Introduction.....	3
2. Plot.....	4
3. Art Style.....	7
4. Interaction and control.....	10
5. Sound Design.....	14
6. Technical specifications.....	17
Leap motion and Unity connection.....	17
Multiplayer in Wwise.....	17
7. Game mechanism.....	19
References.....	21

1. Introduction

In the enigmatic world of *Bleep & Blop*, perception is a puzzle, and reality is anything but certain. This third-person, two-player cooperative game challenges players to step into the roles of NN, a being with complete vision, and MM, a creature gifted with extraordinary hearing. Together, they must navigate a surreal maze garden, where cooperation and communication are not just strategies—they are the keys to survival.

As NN and MM delve deeper into this mysterious world, they begin to uncover unsettling inconsistencies in their memories. These discrepancies spark doubt, forcing them to question the very fabric of their reality. Is the maze all that exists, or is it merely a veil hiding something greater beyond its walls?

Inspired by Plato's "Allegory of the Cave," *Bleep & Blop* explores the tension between perception and truth. NN sees the world in vivid detail, while MM relies on sound to uncover hidden realities (Rauch, 1994). Their asymmetrical abilities create a unique gameplay experience, where players must learn to see the world through each other's senses. Together, they must bridge the gap between their differing perspectives, challenging the illusions they've come to accept as reality.

In NN & MM, the maze is more than a physical space—it's a metaphor for the barriers we construct in our minds. Can you and your partner unravel its mysteries, or will you remain trapped in the shadows of your own perceptions? The journey begins now.

2. Plot

In a vast, labyrinthine world that resembles an endless garden, a small creature named NN has lived for as long as he can remember. This surreal, maze-like garden is filled with cherished memories and friendships, and NN has always found comfort in its beauty. He loves to paint, capturing the vivid details of the world he sees, and he firmly believes in the reality of his visual perception. To NN, the labyrinth is everything—a complete and unchanging reality.

However, NN's friend, MM, has begun to notice something strange. MM, who possesses extraordinary hearing, starts picking up peculiar sounds that defy the logic of their world:

- On bright, windless days, MM hears the distant crash of ocean waves, though no ocean exists within the labyrinth.
- At night, he hears the chirping of birds—sounds that should only belong to the dawn.
- When recalling shared memories with NN, their recollections don't align. The locations, the environments, even the events themselves seem to shift and contradict.

These inconsistencies plant a seed of doubt in MM's mind. He begins to question the very fabric of their world. Is the labyrinth truly complete, or is it merely an illusion—a carefully constructed facade hiding a deeper truth? Driven by his sharp intuition and a growing desire for answers, MM starts to explore beyond the boundaries of the maze, using his acute hearing to uncover what lies hidden.

Meanwhile, NN remains deeply immersed in the world he sees. He paints tirelessly, capturing the beauty of the labyrinth and sharing his artwork with MM, hoping to convince him to stay. To NN, the labyrinth is real, tangible, and safe. He fears what lies beyond its walls and clings to the certainty of his visual perception.

As their journeys unfold, NN and MM find themselves perceiving completely different realities. NN, with his vivid 3D vision, sees a complex and intricate world, but struggles to interpret the sounds that MM hears. MM, on the other hand, perceives the world as a 2D plane, relying on his extraordinary hearing to piece together the hidden structure of the labyrinth. Together, they must navigate this surreal maze, using their unique abilities to solve its mysteries.

NN provides visual clues, painting the world as he sees it, while MM relies on sound cues to uncover hidden pathways and secrets. As they work together, their perspectives clash, forcing them to confront their differences and question the nature of their reality. Is the world they perceive real, or is it an elaborate illusion designed to keep them trapped?

In this labyrinth where reality and illusion intertwine, NN and MM must decide: will they stay within the confines of the maze, accepting the world as it appears, or will they venture beyond its boundaries in search of the unknown truth? Their choices will shape not only their fates but also the very nature of the world they inhabit.

Themes and Inspirations:

Plato's Allegory of the Cave: The game explores the tension between perception and reality, challenging players to question what they see and hear.

The Matrix: The labyrinth serves as a metaphor for a constructed reality, hiding a deeper truth.

Interstellar: The story delves into the emotional and philosophical struggles of characters grappling with the unknown.

Possible Endings:

- The Illusionist: NN convinces MM to stay in the labyrinth, accepting the world as it appears.
- The Seeker: MM convinces NN to leave the labyrinth and face the unknown truth beyond.
- The Divide: NN and MM go their separate ways—one staying, the other leaving—resulting in a profound change to the world itself.

Bleep & Blop is a journey of faith versus doubt, perception versus reality. It invites players to reflect on the nature of truth and the illusions we construct to protect ourselves. Will you cling to the comfort of what you see, or dare to listen for the truth hidden in the silence? The labyrinth awaits.

Character design draft



NN

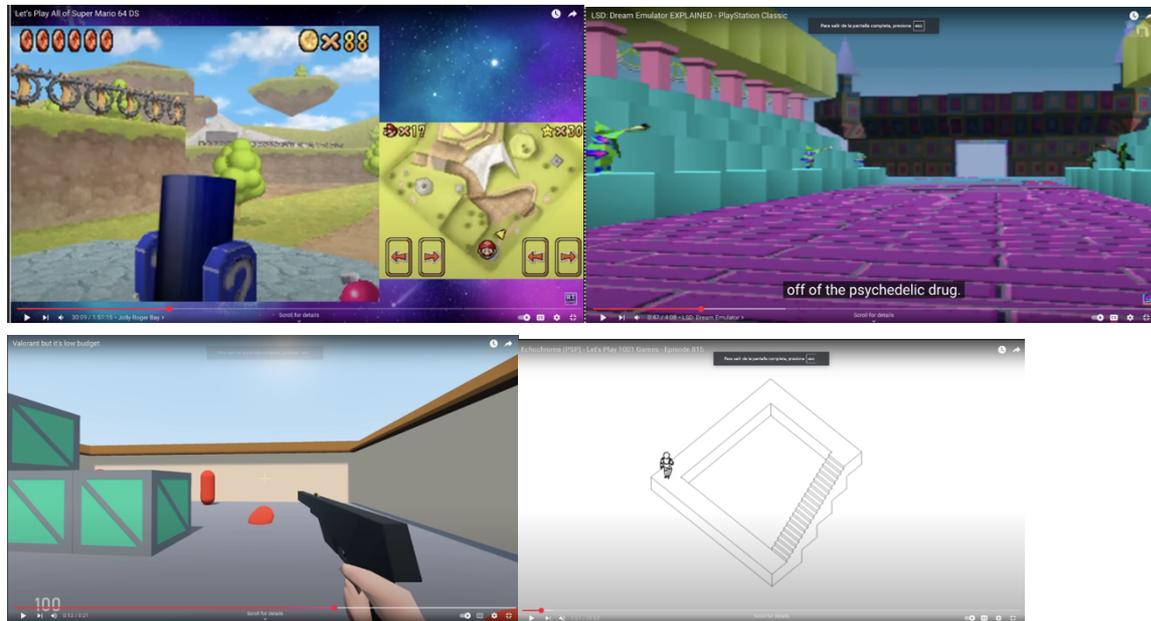


MM

3. Art Style

1. Visual Design (Overall Aesthetic)

low poly aesthetic, simple solid colors and textures



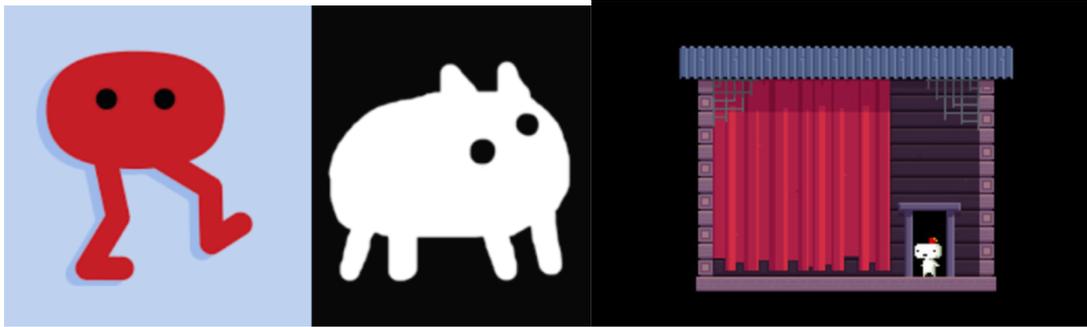
Considering the references to philosophical themes like Plato's Allegory of the *Cave* or *The Matrix*, the visual design might use **contrast** between **reality and illusion**. Using **surreal, abstract, or dream-like imagery** to represent the fractured perception of the characters.

2. Character Design

Only **two** characters, both protagonists.

NN: Loves painting, trusts what he sees, and fully believes in reality. His design could reflect a more grounded or **realistic** style, with **sharp details and a vibrant**, almost idealized appearance. He could be depicted with art-related elements like brushes or easels, reflecting his passion.

MM: Possesses extraordinary hearing, sharp intuition, and a strong desire to uncover the truth. MM's design might emphasize sensory elements, **like large eyes or ears**, or more **ethereal** characteristics that represent his deep intuition. His clothing and features could be more **abstract**, showing a deeper connection to the unseen or unknown aspects of the world.



3. Environment Design

The game could have **interconnected regions** that flow into one another in a **non-linear** fashion, with diverse **environments** — **natural gardens, chaotic urban settings, and abstract landscapes.**

The environment could also **mirror the inner states of the protagonists.** For example, a distorted, **fragmented** world could represent MM's **sharp intuition and desire to uncover the truth**, whereas more serene, **stable** environments could reflect NN's **belief in reality.**



4. Color Palette

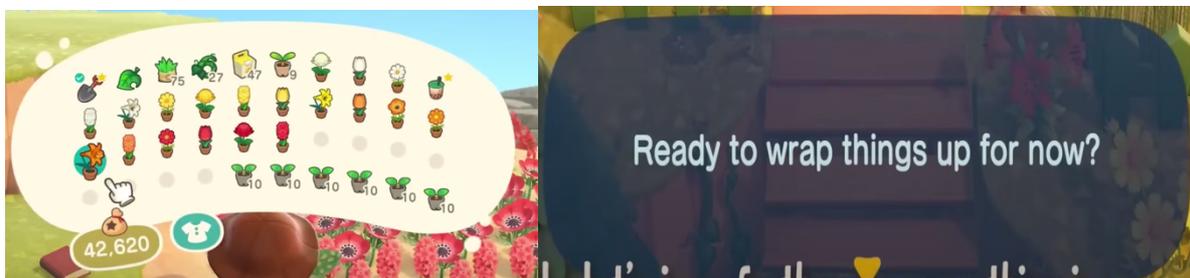
Bright, warm colors could be used in scenes that represent **NN's world**, filled with a sense of comfort or belief in the self.

On the other hand, **cooler, muted tones** could represent **MM's more questioning world**, filled with ambiguity and an exploration of truth. A mix of these tones could also signify the characters' contrasting perspectives on reality.



5. UI Design

UI should be **minimalistic**, stay consistent with the **visual design**



4. Interaction and control

Leap Motion is an advanced gesture control technology that allows users to interact with computers through hand movements. It captures hand and finger movements with extreme precision and responsiveness through a small sensor device. Leap Motion can be used in a wide range of applications such as virtual reality (VR), augmented reality (AR), gaming, design, healthcare, and more.

Leap Motion uses infrared sensors to capture hand movements. These sensors can accurately detect the position, speed, and trajectory of the fingers, and can even sense small changes in movement. Leap Motion analyzes this data to convert hand movements into computer-recognizable signals in real time, enabling gesture control.



It starts with a simple sensor that detects your hands.

The heart of a hand tracking device is two cameras and some infrared LEDs. Ultraleap's hand tracking modules the [Leap Motion Controller](#) and the [Stereoscopic IR 170 Camera Module](#) work on this principle, as do VR/AR headsets built using [Qualcomm's XR2 reference designs](#).

This is the official Leap motion tutorial for connecting to unity.

<https://docs.ultraleap.com/xr-and-tabletop/xr/unity/getting-started/index.html>

ultraleap
FOR DEVELOPERS

Search

Hand Tracking Overview
Haptics Overview
TouchFree
Automotive HMI Guidelines
Ultralab Blog

XR DEVELOPMENT, TOOLS & PLUGINS
XR Overview
XR Design Guidelines
Unity XR

Getting Started
Your First Project
Ultraleap Plugin Features
Further guidance
XRI Integration
MRTK3 Support
Frequently Asked Questions
Unreal XR

Set up hand tracking in Unity

How to install

1. Remove any existing Ultraleap Unity modules from your project

Note

We use OpenUPM to handle installing packages in this tutorial. This is our recommended method as it makes installing and upgrading packages easy.

You can also get the plugin as a .unitypackage or install using the git url. These can both be found on our [open source github repository](#)

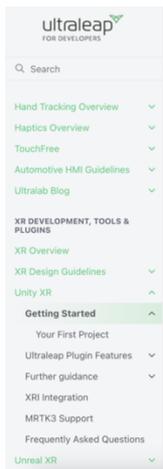
2. In Unity, select **Edit** -> **Project Settings** -> **Package Manager**, add a new scoped registry with the following details:
 - o Name: Ultraleap
 - o URL: <https://package.openupm.com>
 - o Scope(s): com.ultraleap

Project Settings

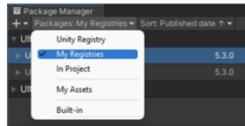
Package Manager

Scoped Registry	Name	URL	Scope(s)
Ultraleap	Ultraleap	https://package.openupm.com	com.ultraleap

UAD Meter &



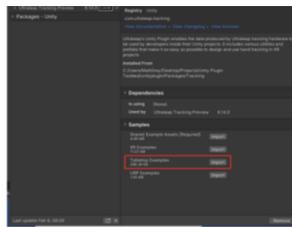
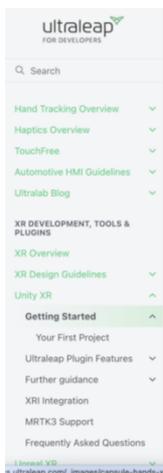
3. Open the Package Manager (Window -> Package Manager) and navigate to "My Registries" in the dropdown at the top left of the window.



4. Ultraleap UPM packages should be available in the list. Add the "Ultraleap Tracking" package by selecting it in the list on the left then clicking install in the bottom right.



5. Include our example content, which demonstrates a lot of the features of the plugin and can be copied and expanded for your project, select the Samples for the package and import them via the Package Manager as shown below.



6. To ensure that everything is set up and working, open the "Capsule Hands" scene in the package samples (Assets > Samples > Ultraleap Tracking > x.x.x > Examples > 1. XR Examples > 2. Building Blocks > 1. Basics > 1. Capsule Hands.unity). With these steps completed, when pressing play, you should be able to see your hands tracking in the scene.



This is the official YouTube tutorial for Leap motion

<https://www.youtube.com/user/LeapMotion>

We also found the official Leap motion forums for technical queries

<https://forums.leapmotion.com/latest>

Example of relative operation

Link: <https://www.youtube.com/watch?v=8KeZxfPh8TE&list=PLCWrkEpZHxqysGzvK-DvU4U8w5qtTimDn>

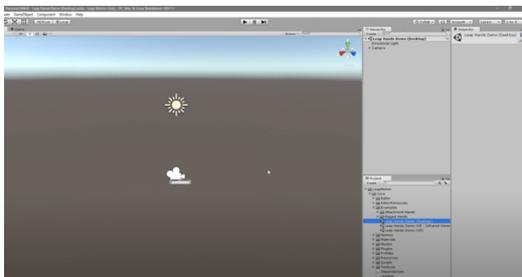
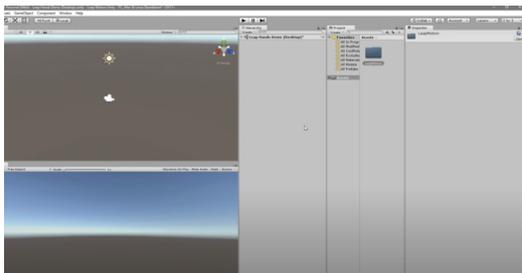
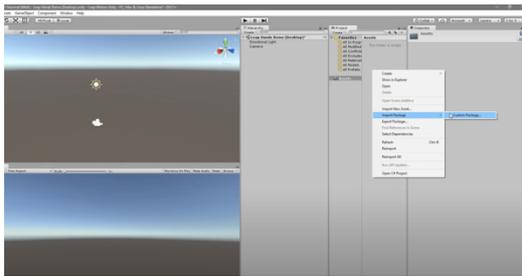
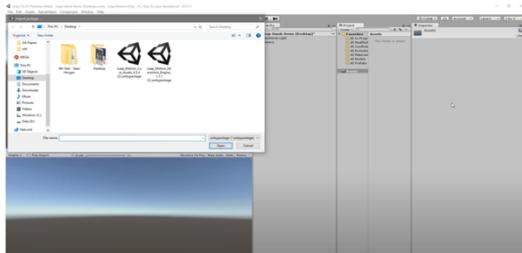
Specification of screenshot:

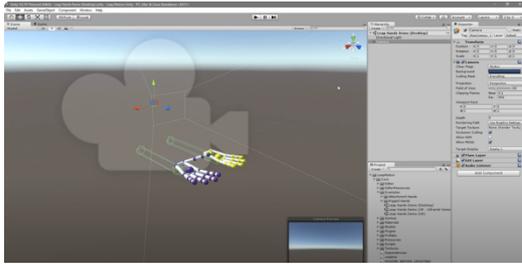
Step 1: download Leap motion, Complete the hardware setup and configuration



Step 2: Develop a custom game with gesture controls by integrating the Leap Motion SDK through Unity or Unreal Engine.

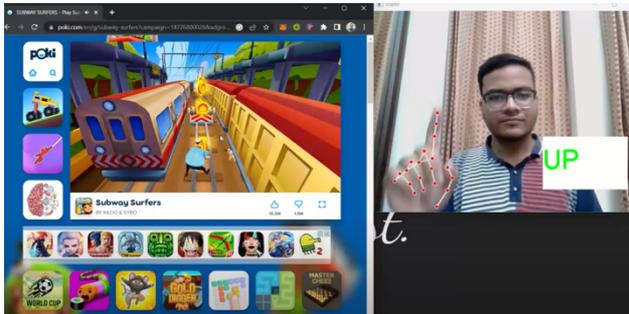
Step 3: Use Leap Motion in Unity





Specific examples of related games

An example of a similar implementation goal, such as using one gesture to control movement forward and a specific gesture to control the player's jump.



Relative video link:

https://www.youtube.com/watch?v=iODL_Xb4IS4

Tools preparation: Unity, Leap Motion



everything is ready to test !

5. Sound Design

1. Sound elements

The general sound design will be categorised in the next elements:

1. Ambience
 - a. Location-specific
2. SFX
 - a. NN's vibrations signs
 - b. Players feedback
 - c. NPC sounds
 - d. User Interface sounds
 - e. Environment
3. Music
 - a. Adaptative music
 - b. Interface/menu
4. Dubbing
 - a. Walas
 - b. NN's vocals

2. Sound description

Ambience

The labyrinth will have different ambiances: day and night ambience, different weather changes and other location-specific ambience.

SFX

NN makes sounds for MM to hear. The vibrations and vocals will be processed in two different ways: NN can barely hear, so she will have a muffled perception of the sound, but MM will hear everything perfectly.

Each character will produce feedback sounds like steps, fur and feathers movement, interaction with objects or creatures.

The dangers in the way can be due to the environment or the creatures around them, this will make determined sound that will be processed differently for the two perceptions.

As one of the characters approaches the correct path, the sound becomes clear and steady. When going in the wrong direction, the sound may gradually become distorted.

Music

Generative music will be created for different situations. When discovering new clues, a mysterious but gentle melody is added. When entering a dangerous area, music adds low, disturbing electronic sounds or ambient noise.

The main menu will have its own background music.

Dubbing

NN's language, walas and creatures' voices will be recorded and processed in post production to create fictional unique characters.

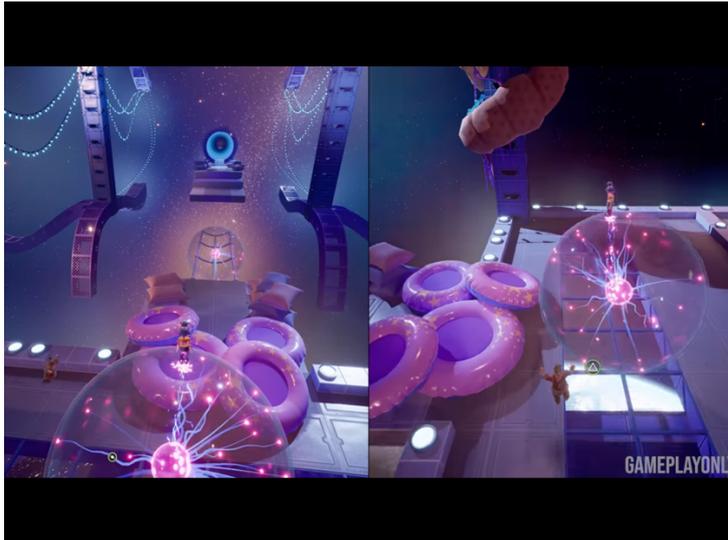
Sound references

1. unravel 2



https://www.youtube.com/live/iUeFObew5mI?si=0Q5hS9ioQjOA_94-

2. it takes two



<https://youtu.be/yZ2VB6nbsUI?si=ed-hYOd0d3ehk6ia>

Character Voice Reference

https://youtu.be/WHHGOYu6Fl0?si=W_yl0X3bdQSW0X93

6. Technical specifications

For the technical realization of the project, it is currently divided into three parts:

1. Leap motion and Unity connection
2. Multiplayer in Wwise

Leap motion and Unity connection

As the controller in the game, Leap Motion captures players' gestures and transfers them to trigger events in Unity. In this case, I am searching for instruction about Leap Motion and Unity connection and here are my results:

Leap Motion official instruction: <https://docs.ultraleap.com/xr-and-tabletop/xr/unity/getting-started/index.html>

Some settings that are good to know before started

1. Tracking Camera
2. Ultraleap Hand Tracking Software
3. Unity XR Plugin Management Package
4. OpenXR

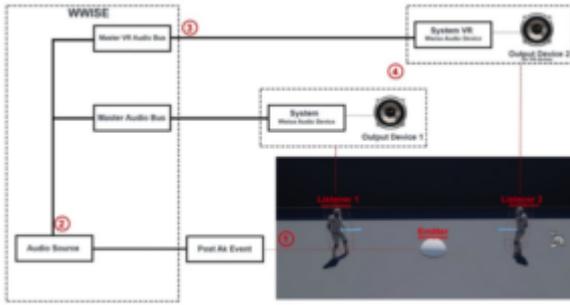
Leap Motion & Unity project blog: <https://felcjo-ringo.medium.com/leap-motion-unity3d-playing-with-virtual-blocks-using-my-real-hands-2329be3a07d6>

Multiplayer in Wwise

Since our project includes two players, there are some special settings and technical issues in Wwise requiring further research. Links below are references about multiplayer audio.

Audiokinetic Instruction: <https://www.audiokinetic.com/fr/library/edge/?source=WwiseFundamentalApproach&id=listeners>

Audiokinetic Blog: <https://www.audiokinetic.com/en/blog/implementing-two-audio-devices-to-your-ue-game-using-wwise/>



Implementing Two Audio Devices to your UE Game Using Wwise, ED KASHINSKY

Based on the game mechanism, most of the modulation for sound and even sound source vary between two players for their different perspectives. This means there are two listener in Wwise and emitters have to send to two output devices in different modulation simultaneously.

A plugin (**AkComponent**) mentioned in the Blog link worth further research. Some programming (plugin) needs to be done if we want two players to have their own sound / two sound devices. My further learning will focus on realization and selecting the best solution for our project.

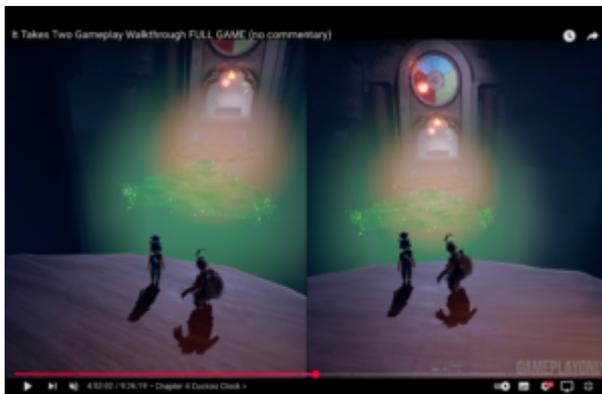
7. Game mechanism

The basic aim of the game mechanism is creating information difference for two players in to perspectives in order to make them cooperate. According to my learning and experience, in most of the recent multiplayer game, players shares equal information or have different information in one perspective (mostly visual). But we can learn from their logic in designing game levels and obstacles.

Two examples below are from It Takes Two.

<https://www.youtube.com/watch?v=yZ2VB6nbsUI>

Time Point:



1. Chapter 4, 4:51:55 Providing visual / aural information with timing challenge



2. Chapter 5, 5:31:38: Replace the magnet with visual / aural ability

I define some logic of game level design here: timing which has some information provided by certain objects as standard to adjust players' movement, rare power which only owned by one player and another player needs to help in some way.

For our project, instead of sharing information, it is always provided for one player due to two perspectives. Timing can be a crucial part in our game level design,

Issue: Two players share the same information on screen. In our project, we may want some information difference based on sound and vision. We need a set of clear and easy communication between the players who receives different perspective from each other.

References

Rauch, L. (1994) 'Imagery and allegory in philosophy', in **Smith, J.** (ed.) *Allegory Revisited: Ideals of Mankind*. Dordrecht: Springer Netherlands, pp. 315-324.

The Matrix (1999) Directed by Lana Wachowski and Lilly Wachowski. [Film] USA: Warner Bros. Pictures.

Interstellar (2014) Directed by Christopher Nolan. [Film] USA: Paramount Pictures & Warner Bros. Pictures.