

UNIVERSITAT
JAUME I

mimicry

An experimental puzzle game mixed with emergent
narrative

Sergio Ferreras Lozano

Final Degree Work
Bachelor's Degree in
Video Game Design and Development
Universitat Jaume I

May 27, 2021

Supervised by: Marta Martín Núñez



To my parents, my sister and all my friends. . . Without you this would not be the same.

ACKNOWLEDGMENTS

I would like to thank my supervisor Marta Martín for her help with my final degree work, and my friends and family for their support throughout this years.

ABSTRACT

In this final degree work I've created a puzzle game with emergent narrative called Mimicry. Mimicry is a game created with the aim of mixing emergent narrative with puzzles, thus reaching an experience that can be enjoyable both for people looking for a meaning to the game and for those who just want a challenge. The results obtained have been quite satisfactory in terms of gameplay and narrative, and the game has been finished in the 300 hours of development that was requested.

CONTENTS

Contents	v
1 Introduction	1
1.1 Objectives	1
1.2 Justification	1
1.3 Related subjects	2
1.4 Initial planification	2
2 References	3
2.1 Game design references	3
2.2 Artistic design references	6
3 Game design	9
3.1 Overview	9
3.2 Gameplay and mechanics	11
3.3 Interface	17
4 Narrative design	19
4.1 Emergent narrative	19
4.2 Characters	20
4.3 Narrative structure	20
5 Artistic design	23
5.1 Characters	23
5.2 Game world	25
5.3 Sound effects / Soundtrack	26
6 Functional and technical specifications	27
6.1 Camera positioning system	27
6.2 Player movement	29
6.3 Scripting for puzzles	30
6.4 Fish tank scene (first scene)	30
6.5 Exam scene (second scene)	30
6.6 Broken screens scene (third scene)	31

7	Project management	33
8	Results	35
9	Conclusions	41
	Bibliography	43
A	References	45
B	Annex	47

INTRODUCTION

Contents

1.1	Objectives	1
1.2	Justification	1
1.3	Related subjects	2
1.4	Initial planification	2

1.1 Objectives

The main objective of this final degree work is to make a puzzle game with emerging narrative with an interesting message behind its gameplay and, as a good videogame, it should be enjoyable both for people who will only play it for its puzzles and for those who are looking for a more narrative experience.

Needless to say, all of the above should have a polished finish, resulting in a game with few bugs and a satisfactory gaming experience.

1.2 Justification

For many years I have seen how people end up looking on the internet for solutions to problems that they do not understand or that were not explained to them properly, in fact, I myself started this degree looking for the answers to my homework on the internet rather than trying to understand the problem I was facing. This has led us to a world in which we believe that all the problems we can face are solved on the net,

without realizing that the internet should not be a tool that gives us a magic solution for everything, but a support tool to find the answer on our own.

All this ended up being the trigger for Mimicry.

1.3 Related subjects

- Programación II
- Narrativa hipertexto y análisis de videojuegos
- Motores de juegos
- Informática gráfica
- Inteligencia artificial

1.4 Initial planification

- Design and program the puzzle mechanics: 60 hours
- Develop the narrative design of the game: 30 hours
- Program the player controls: 55 hours
- Search for assets to use in the game: 15 hours
- Setup the scenes: 10 hours
- Program an NPC to solve predefined puzzles: 30 hours
- Create at least 3 different puzzles: 15 hours
- Write dialogs or text: 5 hours
- Design and program the UI: 10 hours
- Search for music and sound effects to use in the game: 10 hours
- Write the document for my final degree work: 50 hours
- Prepare the presentation of my final degree work: 10 hours

CHAPTER 

REFERENCES

Contents

2.1	Game design references	3
2.2	Artistic design references	6

Before I start talking about my references within the video game industry I think I should also mention some books or articles of great interest that have been a great inspiration to deal with narrative in video games. Among some of them, I recommend taking a look at an article that deals with emergent narrative[1] and a book that talks about procedural rhetoric[2] in which it's told how to use the video game to create narrative out of nothing (usually through game mechanics). An article on the relationship between learning and video games[3] has also been reviewed since it is a topic that is very present in this final degree work.

Finally, I would like to comment that one of the decisions that led me to name my game "Mimicry" is given by the four types of games defined by Roger Caillois in his book *Man, play and games*[4], being one of them Mimicry (aka the action or skill of imitating someone or something).

2.1 Game design references

I'm going to talk a little bit about some of the topics I tried to research more in depth to start the development of the game and also about the video games that influenced me in one way or another.

The idea for a game whose puzzle mechanics included redirecting lasers came from *Archaica: The path of light* (Two Mammoths, 2017). I have to admit that this is not even one of my favorite puzzle games, but I had been thinking about the idea of making my own version of a puzzle game for a while and *Archaica* was one of the freshest ideas I had in my head. In fact, one of the first prototypes for the puzzle gameplay in this final degree work was very similar to *Archaica*'s redirection of lasers with mirrors, with the small addition that the lasers had their own color (just like the mirrors) and the laser changed color depending on which mirror it bounced off (i.e. the laser color was interpolated with the mirror color when reflected).

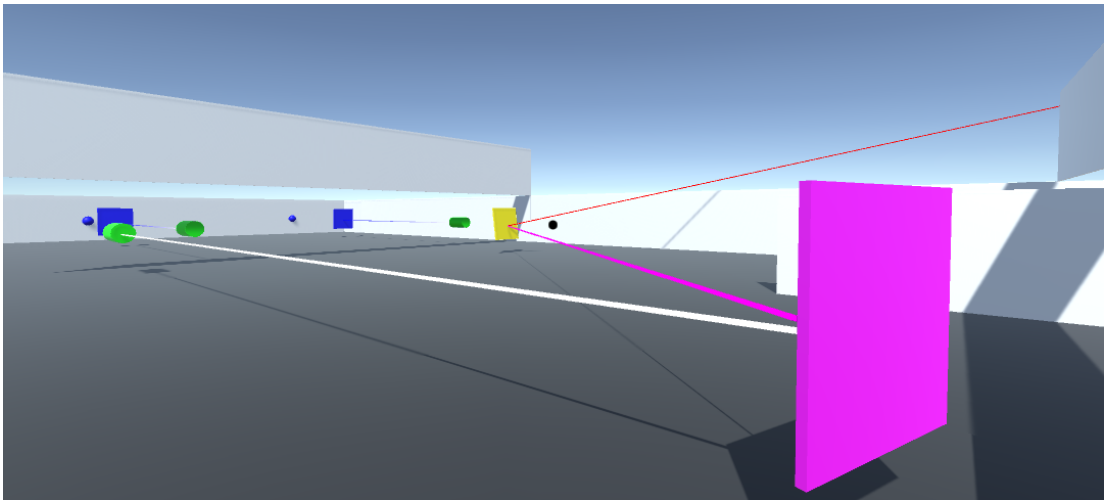


Figure 2.1: Laser game prototype



Figure 2.2: Archaica: The path of light

But *Archaica* was not the only puzzle game that inspired me, in fact, I could say that one of my biggest references when creating *Mimicry* was *The Witness* (Thekla, 2016). *The Witness* is another puzzle game based on solving small panels scattered around a mysterious and beautiful island. In these panels lies the grace of the game since the player will only start knowing how to use them, but the mechanics of future puzzles will have to deduce them on their own.

This is a game with a rather complex and experimental narrative (since at no point is it explained to the player what's going on) that can be completely ignored for those who are just looking for a challenge, but with enough depth for those who like to draw theories or find meaning in their world (which is exactly what I intend to achieve with this final degree work).

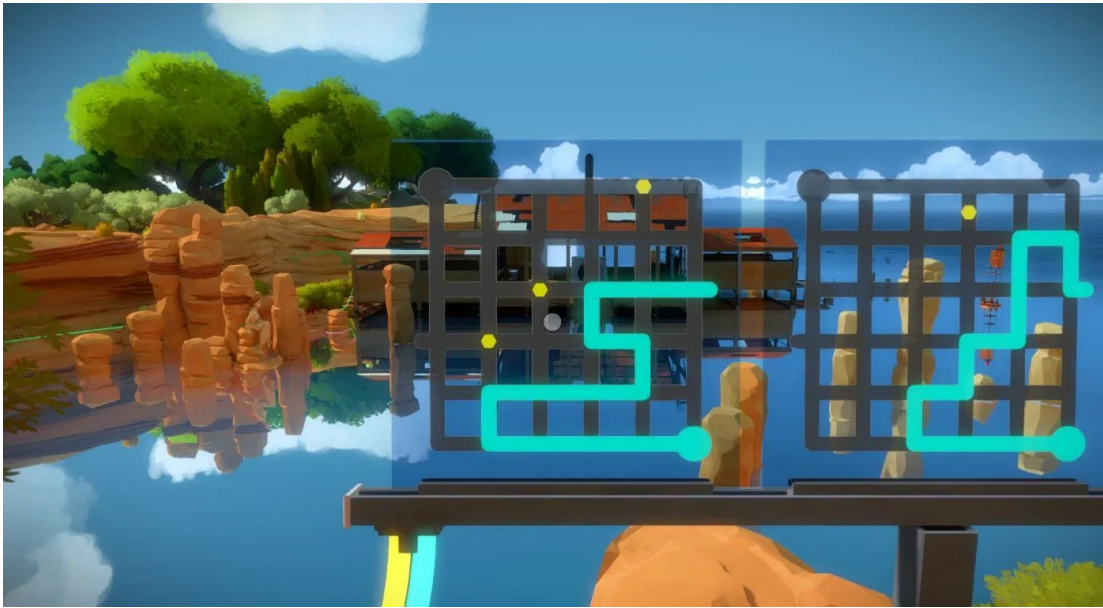


Figure 2.3: The Witness

2.2 Artistic design references

Finally, I have to mention *Portal* (Valve, 2007) because it is not only one of my favorite puzzle games (the portal mechanics are original and super fun), but it also has a very simple and minimalist art style, which has served me as a reference when it comes to build my game visually.

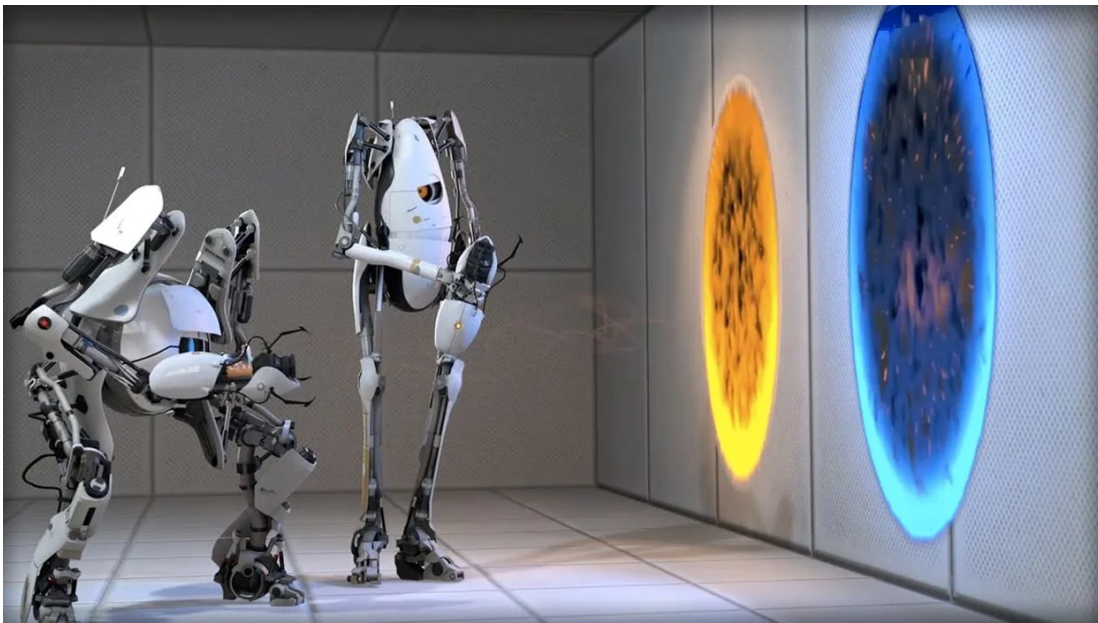


Figure 2.4: Portal

CHAPTER 

GAME DESIGN

Contents

3.1	Overview	9
3.2	Gameplay and mechanics	11
3.3	Interface	17

3.1 Overview

3.1.1 Game concept

The objective of Mimicry is to make the player feel, in an entertaining way, how people learn today to integrate into society. This path develops from childhood to adulthood and has a certain critical view of the culture of copying and memorization that is increasingly present as we advance in our education. Although all this would be useless if the player is not entertained, so the puzzles are intended to be entertaining and challenging while maintaining the narrative through the gameplay, that is, the difficulty should be staggered as the challenges we face with age increase. To achieve this, it is necessary to build puzzle mechanics that are intuitive and interesting enough to make the player want to continue with the 9 puzzles in total that will last the game. In addition, there will be intermediate scenes that will serve as temporal contextualizations (about the age or stage of the character's life) to follow the story correctly. The puzzles will be based on the manipulation of cubes of different colors which will represent the different mechanics that the player will use to overcome the challenges. These cubes will be discussed in more detail in the section 3.2, but I can advance that the main mechanics are based on

the replication of lasers between some of these cubes.

Finally, I have to mention the importance of creating a figure that teaches you how to play and at the same time serves as a narrative vehicle. In this case that figure has been called "the mentor", and is an NPC that will briefly show you how the mechanics of the puzzles works before the player begins to solve them. The narrative importance of the mentor will be explained in future sections.

3.1.2 Genres

- Puzzles
- Single player
- Rich story

3.1.3 Target audience

It's a game that tries to deal with somewhat mature themes (probably a 16+ audience), but if this game should have a PEGI rating, it would probably be PEGI 7.

3.1.4 Game experience

Mimicry is a game that seeks to simulate a learning experience, that is, it tries to start puzzles with basic mechanics and with which any person could manage naturally, to reach puzzles with a greater number of factors to take into account. All these factors and mechanics will be explained later in section 3.2, but the important thing for now is to know that the player will learn in different ways to solve the puzzles. The general idea of this game is to represent the passage of a person from childhood to adulthood, so the game experience in the first areas will focus on the player to grasp the controls naturally and not have too many difficulties to advance (although always trying to introduce a small challenge), and in the later areas the player must interpret the new mechanics introduced on their own (without reaching the limit of just guessing them).

As already mentioned in section 3.1.1, Mimicry tries to criticize how people often try to copy others to overcome a challenge, and that is why in the introductory mechanics an important point is revealed when it comes to understanding the final puzzle. This will allow that only people who really pay attention to what they are taught or people with a great ability to understand problems can finish the game (because even if you don't pay attention to the explanation it is possible to understand how this secret mechanic works, which as I said before, will be explained along with the rest in section 3.2).

Finally, I think I should point out (again) that between some puzzles there are certain "in-between scenes" that contextualize the story and work as a relief for the player

(it's not my goal to have the player 30 minutes just solving puzzles).

The meaning of these scenes will be explained in section 3.2.5.

3.2 Gameplay and mechanics

3.2.1 Gameplay overview

The gameplay of Mimicry is not complex at the level of player mechanics, all that complexity is in the puzzles. The player controls during these puzzles will be the classic WASD keys to move around the environment, and approaching a cube looking in the direction you want to send it will push it a certain distance in that direction. The rest of the mechanics in the puzzles that the player will encounter won't change his controls, but he will have to think about new ways to solve new challenges. The pace of this gameplay is designed to be quite slow so that the player has time to think about the moves he wants to make (it is not intended to motivate the player to solve the puzzles as quickly as possible).

It should be said that in the intermediate scenes the gameplay is slightly different. In the first scene, the player's control is based on moving an arm which follows the movement of the player's mouse, and clicking with the mouse will make the hand touch a fish tank in front of it and scare the fish nearby (it is a quite kinesthetic control since the movement of the arm and touching the fish tank correspond in our world with the movement of the mouse and the click). In the second scene, the mouse control is maintained, only now it is used to move the head to look around (although the viewing angles are limited); in addition, the exam mechanic is added, which basically asks the player to enter with the WASD keys the direction that appears randomly on the projector screen in front of him. The third scene is mechanically a simple first-person control, i.e. you can look in all directions (the viewing angle is no longer restricted as in the second scene) and you can use WASD to move horizontally in any direction.

Note: WASD refers to the computer keys. Commonly in video games the W key is used to go forward/up, A to go left, S to go back/down and D to go right.

3.2.2 Puzzle mechanics

The puzzles in Mimicry are based on taking a laser through cubes until they reach their final receivers. This means that in Mimicry puzzles there will always be at least two elements, the blue cubes and the yellow cubes:

The blue cubes are the cubes that the player will be able to push around the room, and will be in charge of redirecting the lasers to their destination. These blue cubes have two or more white squares (which from now on I will call `laserDisplay`); these `laserDisplay` are the areas through which the blue cubes can receive and emit laser, so

if one of them receives a laser beam then all other laserDisplay of that cube will emit it (so if a cube has more than 2 laserDisplays, in addition to redirect it, you can divide the laser into several more).

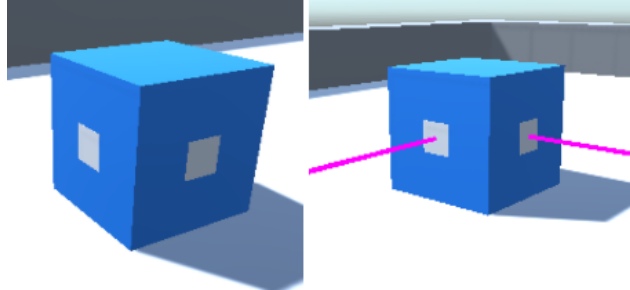


Figure 3.1: Blue cube

Yellow cubes are the wincondition in Mimicry. If a laser hits all the yellow cubes in the puzzle (which have a laserDisplay as a receiver) then you have won and move on to the next area.

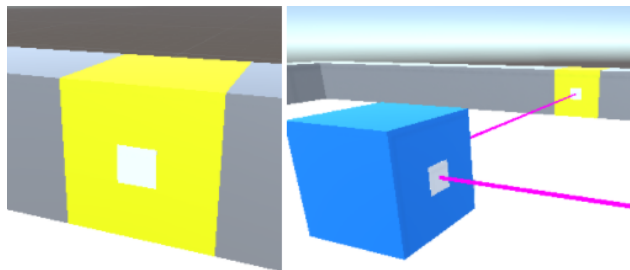


Figure 3.2: Yellow cube

The fact that these are the cubes that are in all the puzzles doesn't mean that the others are less important, in fact, the black and pink cubes have an essential role for the following reasons:

The pink and black cubes are what are known as "generators". Have you ever wondered where the main laser comes from in a puzzle? Well, it is thanks to one of these two. The mission of both of them is to generate lasers but they do it in different ways.

The black cube is a generator without laserDisplays, so it will not be able to generate anything on its own and will need a blue cube in order to work properly. The idea behind the black cube is something like an activator; if a blue cube sticks to the black cube, then this blue cube will start generating lasers through all its laserDisplays.

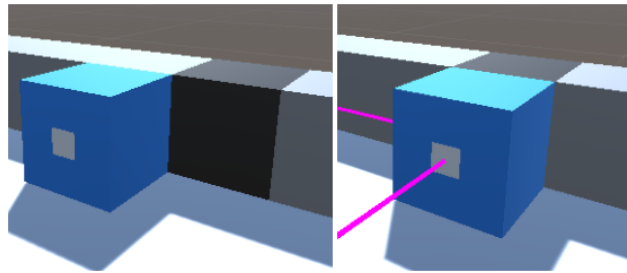


Figure 3.3: Black cube

On the other hand, the pink cube does have a laserDisplay and it will be a cube that permanently generates a laser from the laserDisplay... well, that's not quite true. The pink cube can be permanently active, but it can also be turned off. In the second case, it will need the help of a new cube to activate.

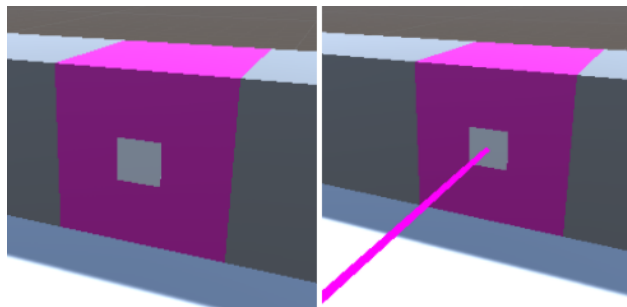


Figure 3.4: Pink cube

The orange cube works as a "receptor" and is in charge of activating the pink cube in case it is turned off. This cube has a single laserDisplay and a single associated pink cube, so if it receives a laser then it will activate that generator (and if it stops receiving it, then the pink cube will be turned off again).

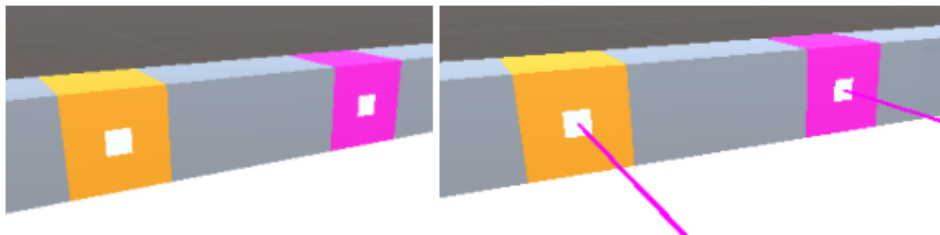


Figure 3.5: Orange cube

These orange cubes have a small trick, and that is that they will not activate their associated cube if the same blue cube lasers two different orange cubes (this mechanic

is the one that brings a big difficulty spike to the final puzzle of the game).

The green cube has no laserDisplay (just like the black cube). This cube works as a kind of "fan" that turns on and off depending on whether a black cube in the room is active or not. Their mission is to push a large amount of distance to the cube in front of them at the moment they are activated. If this cube is active, it won't be possible to push a blue cube next to it until it is turned off. The way to know if it is activated or not will be by its color, being green the color of the cube off and red on.

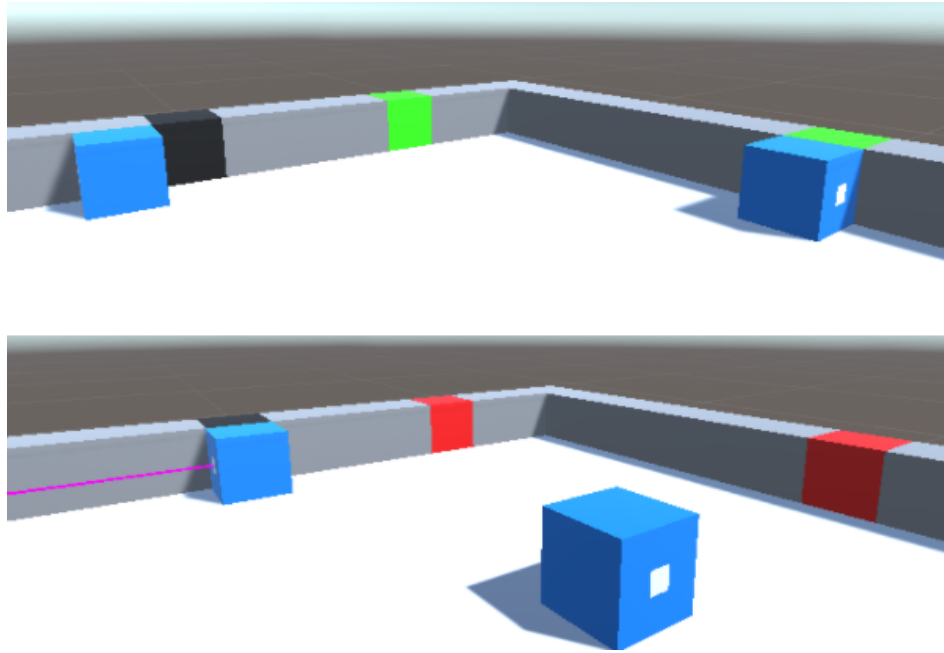


Figure 3.6: Green cube

The violet cubes have a single laserDisplay and function as an "inhibitor". If one of these cubes receives a laser, the puzzle cannot be completed, so the player's goal in a puzzle with this type of cube is to bring the laser to all the yellow cubes trying to prevent the beam from touching a violet cube.

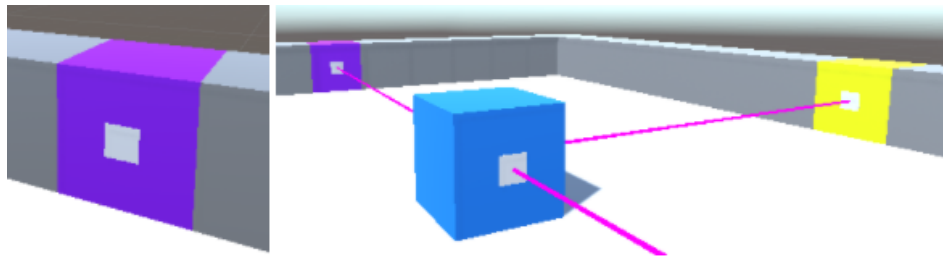


Figure 3.7: Violet cube

The gray cubes are like the blue cubes but fixed, i.e., they can have several laserDisplays and replicate the lasers that reach them, but they cannot be pushed to move them around the area.

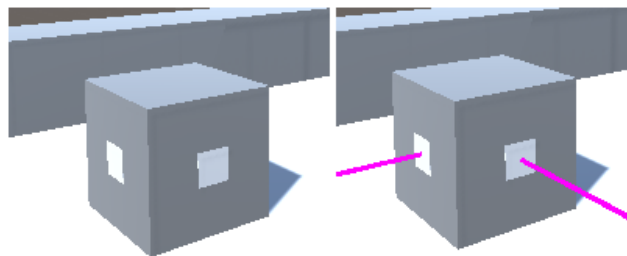


Figure 3.8: Gray cube

3.2.3 Objectives

Through the help of the mentor the player will know that to beat the levels he needs to bring the lasers in the puzzles to the yellow cubes, and the intermediate scenes will be completed in a natural way. The ultimate goal of the player should be, as it is obvious, to complete all the puzzles in the game, but at a narrative level it is turned more by the idea that the character becomes the figure of his own mentor, that is, to achieve that he only needs himself to continue (which is a transformation that the character will undergo along with the player at the end of the game).

3.2.4 Game progression

The game is designed to increase in difficulty in a stepwise fashion, meaning that the puzzles become more complex and challenging as the player solves them. By introducing new mechanics in almost all of these new puzzles and increasing their difficulty a bit more, I manage to make the game progression balanced and satisfying for the player.

3.2.5 Level structure and meaning

The gameplay of *Mimicry* is designed to have a slow tempo. You don't want a puzzle to be solved too quickly, even if you rapidly find the answer. The feel of the game varies between the different first-person stages, and even between the different puzzles.

And before I start explaining each part of the game, I must define what I mean by a "scene". When I talk about a scene in this final degree work, I mean an "in-game" scenario in which the player is in control, and therefore relies on himself to progress with the game.

The first in-game scene that the player will be able to see is a small fishbowl on a table where fish swim freely, and where the player can control the protagonist's hand to interact with them. This scene is designed so that the controls are understood in a natural way, without the need to tell you what to do. This is why the way to move the arm is by moving the mouse and the way to touch the fish tank is by clicking, because it is a universal handling that everyone has internalized when using a computer, and therefore motivates the ease to give free rein to our curiosity (as in the scene, the player behaves like a curious child and starts trying to touch everything, and this is possible because the actions, like the controls, are simple and universal).

The second first-person scene we will encounter is the exam scene. This scene tries to reflect a random exam in a random school, but with a more somber visual style trying to give a certain sense of alienation. The objective here is to answer the questions that are put in front of you, but this time the player doesn't have such a clear and instantaneous answer (like the fishbowl scene) as to what and how to overcome this screen. The exam in itself already gives the player a clue about what to do with the keys, but it won't be until the player by pure inertia starts moving the mouse (it will be the natural reaction since that's what he did in the fishbowl scene) and realizes that he can look at his sides. Here we will see on both sides a row of your peers performing the same test (each one at the speed that suits them best), and it is expected that the player's reaction is to observe (copy) the actions of his peers before performing his own to confirm that his ideas are correct. In short, it is a scene that tries to show how natural it is for human beings to copy the people who are close to them (in this case it is copying in an exam after seeing that there is no one watching over you, but everything could change if an authoritarian figure walking behind you was introduced) and, therefore, tries to show how little by little it is demonstrated that our education often rewards more to reproduce what others do than to think for ourselves. As for the test, it is very simple; you just have to use WASD to select the direction the arrow is pointing to. After doing this several times (the projector restarts at the end of a slide), the scene will end.

The third and last scene in first person takes us to adulthood. The player will find himself in a white corridor in the middle of nowhere and his control will now be complete, i.e. he will be able to move in first person (note that in the first scene the movement

of the camera with the mouse is introduced so that in the second scene he has already learned it, and in the second scene the movement keys are introduced so that in the third scene the player has already completely learned the controls of a first-person movement gradually, and symbolizing in each phase a certain age, being childhood the time when we are more innocent and we move out of pure curiosity, and finally adulthood in which we have learned and mastered all the aspects that we have been taught). The scene is designed so that the player simply walks forward while some screens begin to flood the place until finally finding the figure of the mentor at the end of the corridor; after the final screen (the largest) is turned on, it shows the place as a mirror, and after a blink, the mentor disappears. All this tries to symbolize how the advent of new technologies move that figure of the mentor away from us when we reach adult ages, and suddenly we find ourselves asking more questions on the internet than to the people who have taught us everything when we were little (Mentor (childhood): Parents), or the people who have taught us what we always wanted to know (Mentor (adolescence): Teachers). After the disappearance of the mentor, the player will pass to the final set of puzzles.

The puzzles are of great importance for several reasons, and that's because they are the main mechanical attraction of the game and, therefore, they are the reason why a person will start playing Mimicry. These puzzles follow a tiered level of difficulty that increases as the age of the game character increases. This means that the childhood zone will be of easy difficulty, the teenage zone will be medium, and the adult age will be difficult; but in addition, within each zone the difficulty will also increase, so that level 1.3 will be the most difficult within the easy levels and 3.1 will be the easiest within the difficult levels. When talking about the puzzles in Mimicry, a very important figure must be mentioned: the mentor. The mentor will be the one who will teach you how the puzzles work and will introduce you to the new mechanics that you will encounter when you pass to another level. The mentor will try to be representative of the age in which the player is (as I mentioned above), so in the final zone there will not be a mentor to help you. Also say that the new mechanics that are shown in the levels are dragged to subsequent levels to mix them with new ones and thus create more complex and interesting puzzles (almost every level introduces a new mechanic).

3.3 Interface

From the beginning it has been pursued that Mimicry wouldn't have a HUD to improve immersion in the game, but it does have a main menu and a pause menu.

The main menu will give the player the option to start the game or exit the game, and the in-game pause menu will give the player the option to continue the game, exit the game, and most importantly, restart the current level (in case the player gets stuck in a dead end or simply prefers to leave the puzzle as it was to clear the ideas in his head).

CHAPTER 

NARRATIVE DESIGN

Contents

4.1	Emergent narrative	19
4.2	Characters	20
4.3	Narrative structure	20

4.1 Emergent narrative

Mimicry is a game that was built on the idea that the story should be ambiguous enough to create some mystery and capture the player’s attention, but also clear enough that the player could easily take on the role of the protagonist, and thus reflect the events of his life in the events he observes. This, as you might expect, gives rise to an emergent narrative; why? Because Mimicry does not try to tell a fixed story, but one that molds itself to each person and emerges from the actions we have all experienced during our lives (the idea is that the player sees himself reflected in what is happening and reflects on his own world through the one I create). In this way the narrative that one person may experience may be very different from the one experienced by another... it all depends on each person’s point of view.

That’s why the characters you see don’t have any special characteristics that make them stand out, but function as vessels that our mind can shape.

Even so, the narrative of Mimicry will have a common nexus for all players, and is that we will see the passage of the years from the point of view of a person through the story of his life, starting with the curiosity of a child, through the tasks of a teenager,

and reaching the independence of adulthood; all wrapped with a critical message to the current education systems through the use of game mechanics and audiovisual language.

4.2 Characters

This story will only have two characters (three if you count the people at your side in the second scene of the game), and these will be the protagonist and the mentor. The protagonist, as mentioned before, should be a reflection of the player, while the mentor should impersonate the person who has taught the protagonist/player at each stage of his life, that is, in the stage of childhood symbolizes our parents, in adolescence to our teachers, and in adulthood he leaves us, highlighting the independence of the protagonist from that point on, but also his new responsibility, since now there will be no hand to guide the player in the following levels, and he will have to understand the new mechanics and challenges by himself.

4.3 Narrative structure

The reason for choosing a staggered progression in the difficulty of the game was mostly to reflect how life gets harder and harder the further your mentors get away from you, the people who have guided you throughout your life, until it's time to face new challenges, and if you follow that urge to "copy" someone else to learn how to overcome them, you may never truthfully make it.

As for how the story was structured, Joseph Campbell's Hero's Journey (*The Hero with a Thousand Faces*, Joseph Campbell, 1949)[5] was taken as an example. All the phases of the protagonist's life coincide with what would be a classic structure of introduction, knot and denouement.

Not all the elements that make up the hero's journey were taken because emergent narrative cannot afford to fix all narrative structures (or there will be nothing for the player to imagine beyond what he sees). For example, the call of adventure in Mimicry is an event that does not take place, and this is because it is about making a story that fits like the life of anyone who can play the game. However we can find a transformation in the character (who learns at the same pace as the player) and a crisis stage, which is the moment when the mentor disappears, leaving the player alone to learn in a more proactive way. The puzzles to be solved throughout the adventure would be the obstacles the hero has to face in order to achieve his goal.

The final puzzle is considered to be the ultimate test since both the character and the player himself are forced to remember the lessons the mentor taught them in the past and put them to work.

At the end, the protagonist is reunited with his mentor in a final scene that functions as a reward to the player for completing the game.

CHAPTER 

ARTISTIC DESIGN

Contents

5.1	Characters	23
5.2	Game world	25
5.3	Sound effects / Soundtrack	26

This final degree work does not focus on artistic development, so most of the models have been free assets searched on the internet. Even so, I've tried to maintain a certain aesthetic to keep the game visually coherent and pleasing to the eye of the player.

5.1 Characters

The characters in Mimicry (the player and the mentor) are the test bots offered by mixamo.com for testing animations. The reason why these models were chosen for the characters derives from the search for a simple and not too detailed aesthetic, without looking for visual realism. In this way it is much easier for anyone to see themselves reflected in the main character (since a more realistic character, apart from creating a figure of a character that shouldn't even have a name, would be more difficult to animate and, in general, to make it look good).

The main character will be the one in blue, and will represent the player in all the stages of the game. These will be childhood in the introduction, adolescence in the middle and adulthood in the denouement.

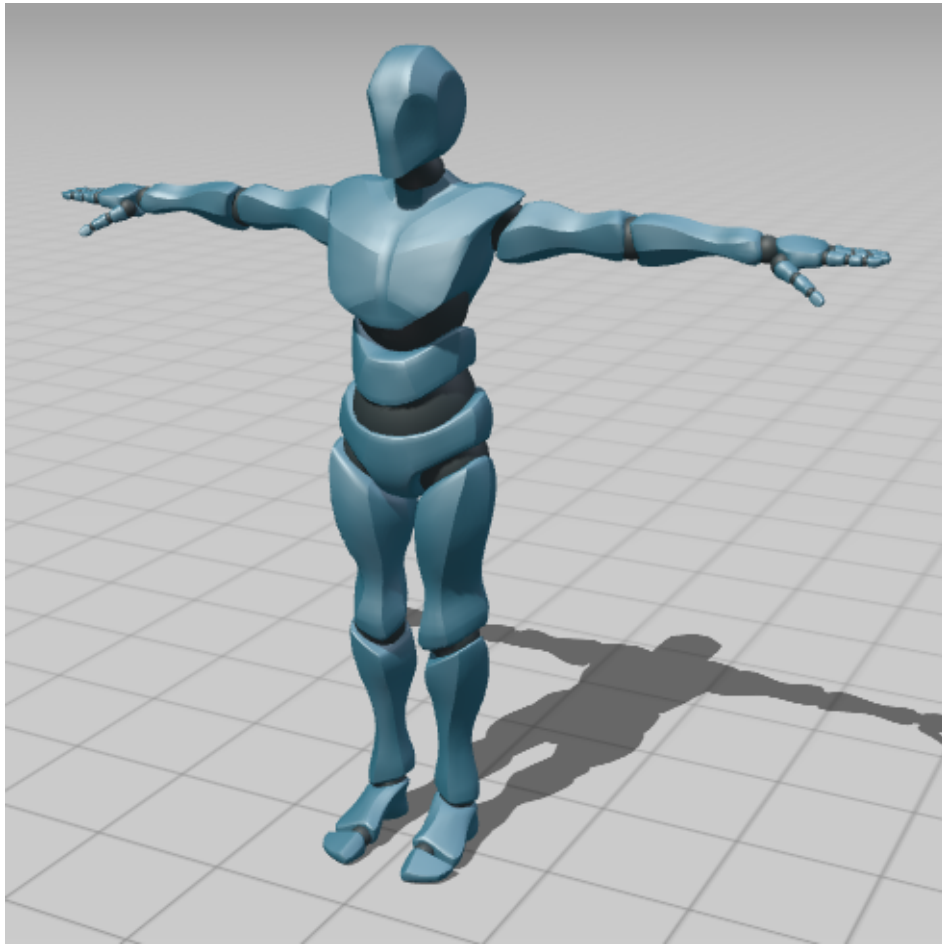


Figure 5.1: Main character

The mentor will be the character in red, and will represent the different types of teachers that the player has throughout the game. These will be the parents in the introduction and the teachers in the knot (since in adulthood it's replaced by the internet, which is represented by a large number of digital screens).

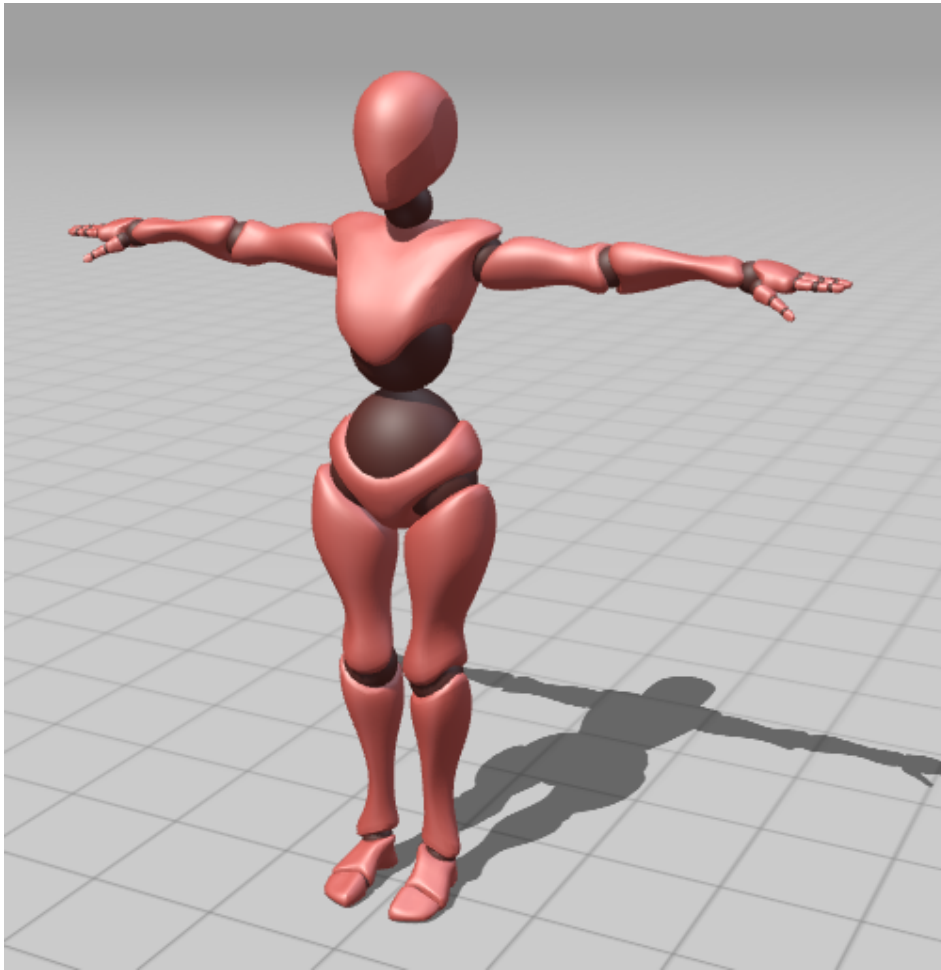


Figure 5.2: Mentor character

5.2 Game world

Following what was explained in characters, the world also had to be simplistic in order to keep the general aesthetics of the game. That is why Portal was chosen as a reference for the models of the walls and floors of the different areas.

Something to comment especially about the floor of the areas in the puzzles is that it was chosen in the form of tiles to improve the gameplay experience. The reason for this is because the movement of the cubes is fixed (they move 1.5 units each time they are pushed), and a perspective view from the cameras sometimes gave rise to confusion when it came to, for example, understanding how deep a blue cube was in relation to a yellow cube (i.e., it was not very clear at a glance if they were aligned). In this way, by making the floor tiles perfectly match the positions and sizes of the cubes, the player

can use the lines on the floor as a visual reference like a "grid" to make sure that a cube is in the desired position.

It is also interesting to say that the puzzles are in outdoor areas because the lighting of the puzzles in enclosed spaces was somewhat difficult to perform, plus a greater amount of light (typical of open spaces) felt much more pleasant when concentrating to solve the puzzles.

5.3 Sound effects / Soundtrack

The sounds used have been mostly taken from the internet (their respective authors will be linked in the bibliography [A]). In general, I have looked for sounds to mark transitions and music to play in the background while solving the puzzles. This music has been searched trying not to distract or disconcentrate the player in the puzzles, but to be somewhat relaxing and enjoyable (except in some specific scenes where it serves more to enhance the atmosphere).

FUNCTIONAL AND TECHNICAL SPECIFICATIONS

Contents

6.1	Camera positioning system	27
6.2	Player movement	29
6.3	Scripting for puzzles	30
6.4	Fish tank scene (first scene)	30
6.5	Exam scene (second scene)	30
6.6	Broken screens scene (third scene)	31

All the code for what I'm going to explain can be found in the GitHub repository linked in section 8.

6.1 Camera positioning system

The cameras in Mimicry depend on whether or not the player is in a puzzle area. In the affirmative case, the scene will be seen with a fixed third person camera, and in the other case it will be a first person camera (and here it can be fixed, free but with limiters, or completely free).

I will proceed to discuss how the cameras work in each type of scene:

6.1.1 Camera in puzzles

In the puzzle areas the cameras work with the help of a free Unity camera management package called Cinemachine[6].

Thanks to Cinemachine I can get in a very simple way the camera following the character after passing invisible thresholds on the screen (see Figure 6.1).

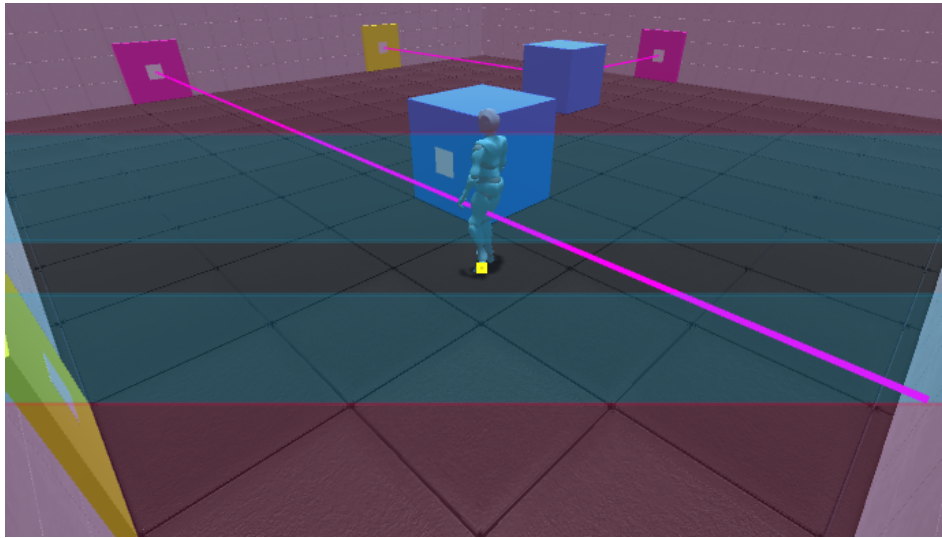


Figure 6.1: Camera in puzzles

Finally it is worth mentioning how to switch between the two cameras. The system is quite simple: there are two cube-shaped triggers in the scene. From here I can control the priority of the cameras created by Cinemachine to switch the position of the main camera between one place and another when the player collides with a trigger. It will also recalculate the value of the player's inputs so that the controls adapt to the new camera position. These values will be recalculated not only when the trigger is hit, but also when the player lifts a finger off the key they were moving on (this was the way it was done in the days of consoles like the first PlayStation, and is done this way so that the player can continue in a fixed direction even when changing cameras).

6.1.2 Camera in second and third first person scene

The camera of this scene is quite similar to the camera of any first-person game. Basically it reads each frame where you want to move the mouse on the X and Y axis (the mouse is locked in the center of the screen), and based on these values it calculates how much to rotate the player's camera. To this basic script was added the possibility to limit the angle of movement of the camera, and also it is detected if only move the camera or also rotate the player's body (since in the third scene the player will need the body to rotate so that the movement is kept in the right direction, while in the second

scene I have no body to move and I just want to rotate the camera).

I won't talk about the camera in the first scene since it is a fixed camera without any kind of control.

6.2 Player movement

Again, I will divide this section into first and third person control:

6.2.1 Control in third person

This control depends entirely on the position of the camera, so it needs the help of two vectors that match the forward direction and the direction perpendicularly horizontal to the camera.

The movement of the character is activated by pressing the WASD keys, and thanks to Unity's input system I can detect the position on the coordinate axis that each button represents (for example, the W represents the value 1 on the Y axis, and the S represents the value -1 on the same axis). In this way, and with the help of the vectors calculated from the position of the camera, I can know the direction to which the player should move according to the key he presses.

It should be said that from the beginning it was intended to have a realistic movement, so I tried to avoid having the player always moving at a linear velocity. For this purpose, acceleration and deceleration values were created, and in this way it was achieved that the character's body had a uniformly accelerated movement, that is to say, when starting to walk it takes a few tenths of a second to reach the maximum speed, and when stopping it takes another little time to slow down completely.

6.2.2 Control in first person (first scene)

The control of the arm is quite simple, the most complicated thing was to get the camera and the arm in a natural position for the player's view. It is based simply on moving the fingertip to the same place that the mouse occupies on the screen, and then check if the player clicks to make the fish respond to that input (although this will be discussed in more detail in section 6.4).

6.2.3 Control in first person (third scene)

This control was much easier to achieve since it is based on the Unity component `CharacterController`[7]. With its help I can simply move the character in the direction relative to the first-person camera (which was explained above).

6.3 Scripting for puzzles

As explained above, the puzzles in Mimicry are based on getting lasers to specific locations. These lasers are implemented thanks to the use of the Unity LineRenderer[8] component. The general idea is to use a programming pattern called ObjectPool to load a number of lasers into the scene and update their position every frame to make them work.

The way to make the player interact with the blue cubes (to push them) has to do with the distance between the player and a cube, his direction with respect to the cube he is looking at (is he looking at it from the side or is he really facing a face of the cube?) and whether or not it is moving. Obviously it is also taken into account when pushing a cube if there is something behind it, so you won't be able to push it in a direction with an obstacle.

The types of cubes have already been discussed in section 3.2.2, so I don't think it's necessary to talk about all of them again. I think it is only necessary to mention that both the black cube and the green cube need a trigger to know which cube should be affected when they are activated, and that all laserDisplays in every cube uses Raycast[9] for detecting what's in front of them (which helps them detect when they stop receiving a laser from another cube or if the cube in front of them is not turned on).

Finally, I wanna add that the way to know when a puzzle is solved is through a GameManager that is in charge of checking if all the yellow cubes are receiving a laser (and depending on the case, that no violet cube is receiving it).

All the scripts for what I just explained and what I'm about to explain can be found in the GitHub repository (which is linked in section 8).

6.4 Fish tank scene (first scene)

This is the only scene in the game in which an artificial intelligence algorithm is executed in real time. In this case, the algorithm called Flocking is in charge of giving life to the fish in the tank, giving them a natural movement (as if they were real fish). The interaction between the player and the fish is given thanks to an object that changes to the position in which the player clicks and accelerates the fish that are in its range for a short period of time.

6.5 Exam scene (second scene)

The exam being taken in the second scene is actually the following image (see Figure 6.2) being modified in real time.

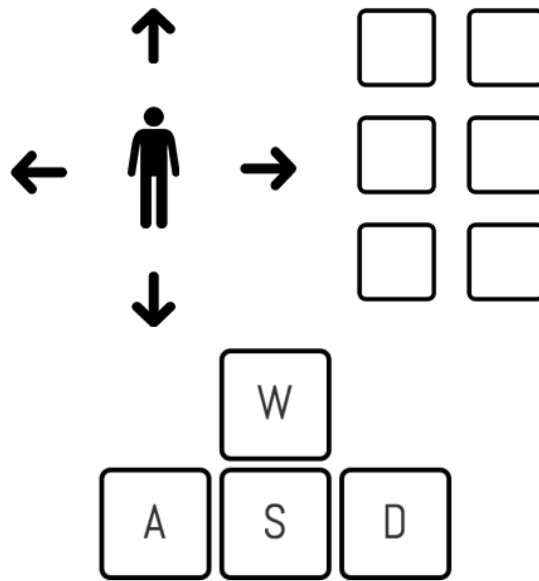


Figure 6.2: Base exam of the second scene

For this purpose, blank images have been placed in front of each arrow, which change randomly as the player gives an answer. The boxes on the right also have a blank image in front of them, but this will not disappear to reveal something behind (as with the arrows), but will change to an X or a check depending on whether the answer given is wrong or right.

The bots that the player sees on either side will simply give random answers (although each at a different speed).

6.6 Broken screens scene (third scene)

The biggest challenge when creating this scene was to program the noise shader for the screens that were going to fill the place. During the corridor there are several triggers that make the screens appear in a random order, and at the end of the corridor (when you reach the mentor) there is a final trigger that executes that kind of "blinking" in which the mentor character disappears and the big screen shows an image of the place.

CHAPTER



PROJECT MANAGEMENT

Trello has been used to monitor the project. Trello is a tool that allows you to divide tasks into cards and move them between tables as if they were posits. In addition, in each card you can put lists of subtasks to complete (to keep a progress within each task) as well as due dates, to give a couple of examples. This has allowed me to divide the work early on into the tasks I needed and give them a deadline in development based on how important or difficult I felt that task would be, which has resulted in tasks being completed earlier than expected and tasks being delayed, but with them successfully organizing the development of the game.

Trello was also used to note bugs and errors in the game, as well as possible improvements to the player experience.

To avoid losing motivation in the project, I decided to build a version of the game every week to be able to test it and that the progress made would gradually become visible. In addition, in a folder shared with several people, some of these builds were uploaded for testing sessions (in case these people could). In this way, over time, the game was taking shape and progressing at a good pace.

Although from the beginning I tried to set realistic goals for the development of the game, there are certain aspects that underwent changes during the development stage due to lack of time. For example, the mentor in the initial concepts could have been in the first scenes of puzzles as a real-time tutorial, but eventually the mentor tutorials became cutscenes that are played before each puzzle.

On reflection, I think the initial plan has been followed satisfactorily. The conceptualization phase of the game did not take more than a couple of weeks and all the tasks that were written at the beginning were developed within the expected timeframe.

CHAPTER



RESULTS

The work has paid off, and the result of the development of this final degree work is a demo of Mimicry that, according to the data obtained from the people who have tried it, has a duration of between 15 and 45 minutes. It allows the player to try all the intermediate scenes in first person, in addition to the 9 puzzles designed for the game, and all this in the order of events already explained earlier in this document... basically the full experience that was intended to achieve at the beginning of the year.

As for the goals described in the initial planning, the vast majority of tasks have been accomplished in the expected time, with the small difference that the task of writing dialogues was eliminated as well as the task of creating an NPC to solve predefined puzzles. The remaining hours were invested in tripling the number of playable puzzles and improving the overall game experience.

The final hours dedicated to each task looks like this:

- **Design and program the puzzle mechanics:** 70 hours (+10 hours from initial planning)
- **Develop the narrative design of the game:** 30 hours
- **Program the player controls:** 50 hours (-5 hours from initial planning)
- **Search for assets to use in the game:** 10 hours (-5 hours from initial planning)
- **Setup the scenes:** 25 hours (+15 hours from initial planning)

- **Setup, record and program NPC cutscenes:** 10 hours (-20 hours from original task)
- **Create 9 different puzzles:** 30 hours (+15 hours from original task)
- **Design and program the UI:** 15 hours (+5 hours from initial planning)
- **Search for music and sound effects to use in the game:** 5 hours (-5 hours from initial planning)
- **Write the document for my final degree work:** 50 hours
- **Prepare the presentation of my final degree work:** 10 hours

Total time: **305** hours.

Below, I attach some links of interest to check the results:

Link to a folder where you can download the game and watch a gameplay video:

<https://drive.google.com/drive/folders/1cBMmVhp3hhLqmfiB1ARBERnJhaoBQcAm?usp=sharing>

GitHub repository: <https://github.com/SergioPucela/Project-Mimicry>.

Some screenshots of the game:



Figure 8.1: Fishbowl (first scene)

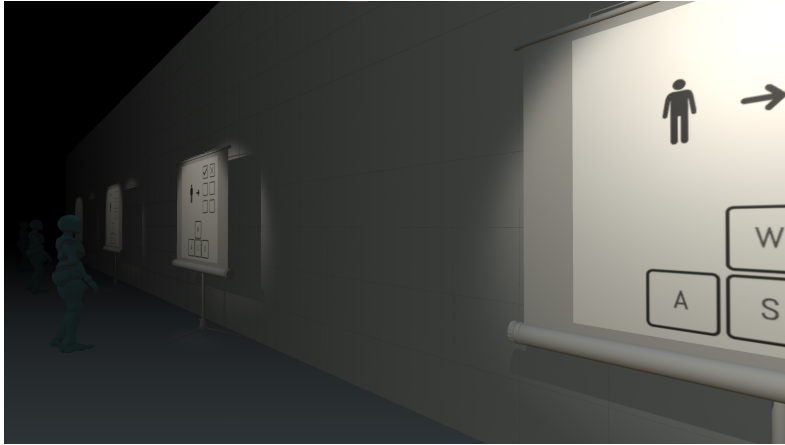


Figure 8.2: Exam (second scene)

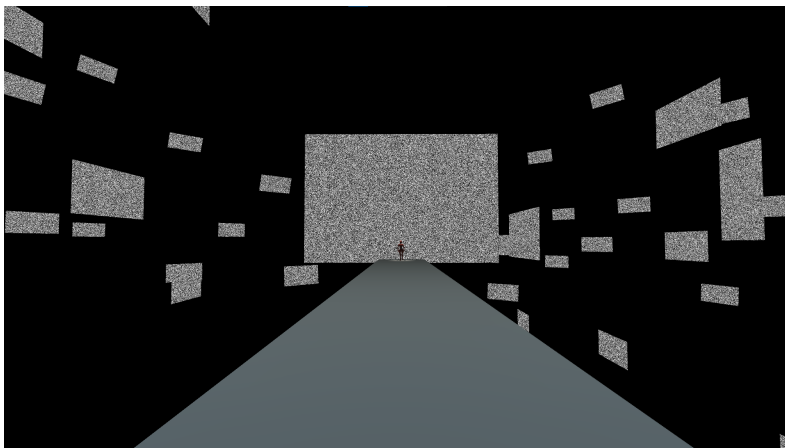


Figure 8.3: Broken screens (third scene)

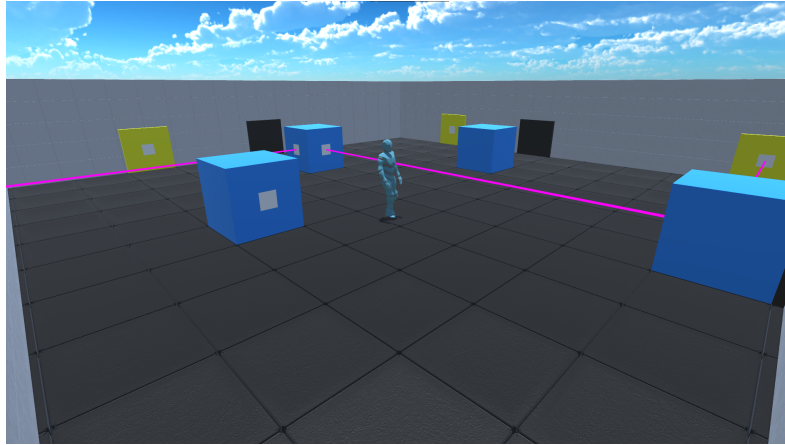


Figure 8.4: Level 1.3 (puzzle scene)

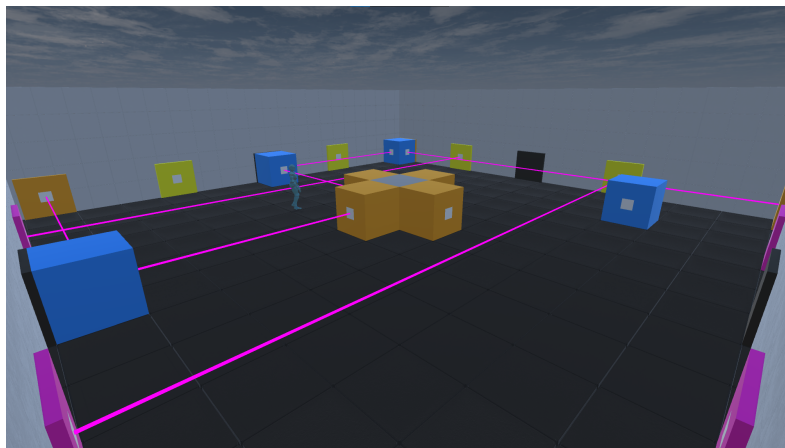


Figure 8.5: Level 3.3 (puzzle scene)

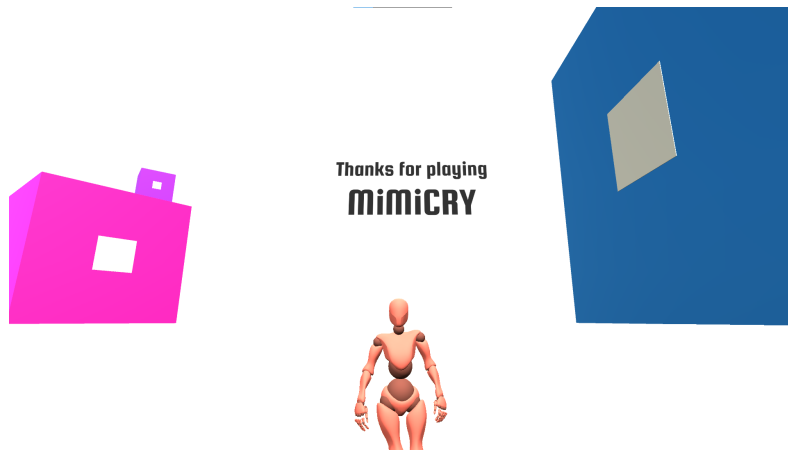


Figure 8.6: Ending scene

CONCLUSIONS

I will try to talk point by point about what I think I found the final result of the game and how it could be improved (if it were the case).

I think Mimicry as a whole has successfully managed to tell the story I wanted to tell, but along the way I've realized that some things could be improved for future development, for example, the puzzles. I don't want to be misunderstood, I think the puzzle mechanics are quite satisfying and the overall design is quite satisfactory, but I do think the pace at which they are solved is too slow. Let me explain: in puzzle games you should always have a trap in which you trust that everyone must fall (it will be the challenge of that puzzle) and you must look for the player to have a sense of accomplishment so that all that time invested has been worth it... in Mimicry that feeling of "eureka" exists, but sometimes the amount of movements you have to make to get to it is somewhat tedious and can make that feeling of satisfaction dissipate over time while you place the cubes to solve it; so I think that in the future I have to look for more agile mechanics so that the player can try and rectify much faster in a puzzle.

Another aspect that I would like to have improved is the visual aspect, although this does not concern me so much since this final degree work was never intended to be a work of art or modeling, but I do believe that with the help of an artist the game would have been much more visually pleasing and, consequently, it would be much easier to attract the attention of a player who is looking to play something new.

The last aspect I wasn't entirely happy with is the mentor cutscenes. As I mentioned before, the mentor should have been a character (at least in the first levels) that would help the player with the main mechanics of the puzzles, but due to lack of time it was

decided to change the AI for that NPC for a cutscene that would show a kind of mini-tutorial so that the player could get an idea of how to play. The problem I have seen with the cutscenes (although they work perfectly as a tutorial) is that there is the possibility that, if you are not very attentive, you can miss something important, whereas if the mentor had been an NPC in the room he would have reacted to your actions in such a way that the player would have the opportunity to get distracted without too many consequences.

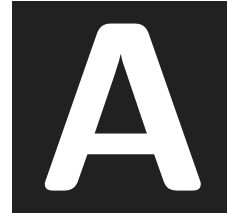
Those would be the aspects that I would like to improve with time and experience, but I've also learned many things in the development of the game. For example, I've learned how to program shaders in Unity (until now I had only done it in Open GL), I've learned a lot about how to handle sound in a videogame, I've learned how Unity's URP (Universal Render Pipeline) works to use image postprocessing, and so on.

In general I have to admit that I'm happy with the result of the game, because I was really looking forward to dedicate a lot of time to a project that would allow me to apply a narrative to the metalanguage that videogames handle, and that's exactly what I think I have achieved... although I have to admit that during the development I was a bit afraid that the way to do it was a bit pretentious on my part (after all, it is a quite experimental and abstract game in terms of the way to tell certain facts).

BIBLIOGRAPHY

Here are some of the books and articles I've named in this document, as well as some technical documentation links:

- [1] Federico Peinado and Michael Santorum González. Juego emergente: ¿Nuevas formas de contar historias en videojuegos? (Madrid. Universidad Complutense de Madrid. 2012).
- [2] Ian Bogost. Persuasive Games: The Expressive Power of Videogames. (Massachusetts. Massachusetts Institute of Technology. 2007).
- [3] Francisco Ignacio Revuelta Rodríguez and Jorge Guerra Antequera. ¿Qué aprendo con videojuegos? Una perspectiva de meta-aprendizaje del videojugador. (Extremadura. Universidad de Extremadura. 2015).
- [4] Roger Caillois. Man play and games. (Chicago. University of Illinois. 1958). Chapter 2.
- [5] Joseph Campbell. The Hero with a Thousand Faces. (1949).
- [6] Unity Technologies. Cinemachine documentation. <https://docs.unity3d.com/Packages/com.unity.cinemachine@2.3/manual/index.html>.
- [7] Unity Technologies. Charactercontroller documentation. <https://docs.unity3d.com/es/2019.4/Manual/class-CharacterController.html>.
- [8] Unity Technologies. Linerenderer documentation. <https://docs.unity3d.com/Manual/class-LineRenderer.html>.
- [9] Unity Technologies. Physics.raycast documentation. <https://docs.unity3d.com/ScriptReference/Physics.Raycast.html>.



REFERENCES

Video games I've talked about:

The Witness: <https://store.steampowered.com/app/210970/TheWitness/?l=spanish>

Portal: <https://store.steampowered.com/app/400/Portal/?l=spanish>

Archaica: The path of light: <https://store.steampowered.com/app/550590/ArchaicaThePathofLight/>

Free assets used for the game:

Old nightstand: <https://assetstore.unity.com/packages/3d/props/furniture/old-nightstand-100266>

Walls and floor: <https://assetstore.unity.com/packages/3d/environments/shooting-gallery-environment-pack-105306>

Exam projector screen: <https://www.turbosquid.com/es/3d-model/movie-screen>

Fishbowl: <https://free3d.com/3d-model/freshwater-aquarium-v1-721361.html>

Characters: <https://www.mixamo.com/?page=1query=bottype=Character>

Skyboxes: <https://assetstore.unity.com/packages/2d/textures-materials/sky/skybox-series-free-103633>

Free sounds and music used for the game:

Transition sound: <https://freesound.org/people/Rvgerxini/sounds/475691/>

Main menu music: <https://freesound.org/people/Erokia/sounds/183881/>

Broken screen sound: <https://www.soundjay.com/tv-static-sound-effect.html>

Wind sounds: <https://mixkit.co/free-sound-effects/wind/>

Puzzle theme and ending theme: <https://www.fesliyanstudios.com/royalty-free-music/downloads->

[c/peaceful-and-relaxing-music/22](#)

ANNEX

First sketches of puzzles (in the image shown you can see how one of the mechanics being considered was the cubes traveling on automatic conveyor belts and pass through portals or rotate at certain key points):

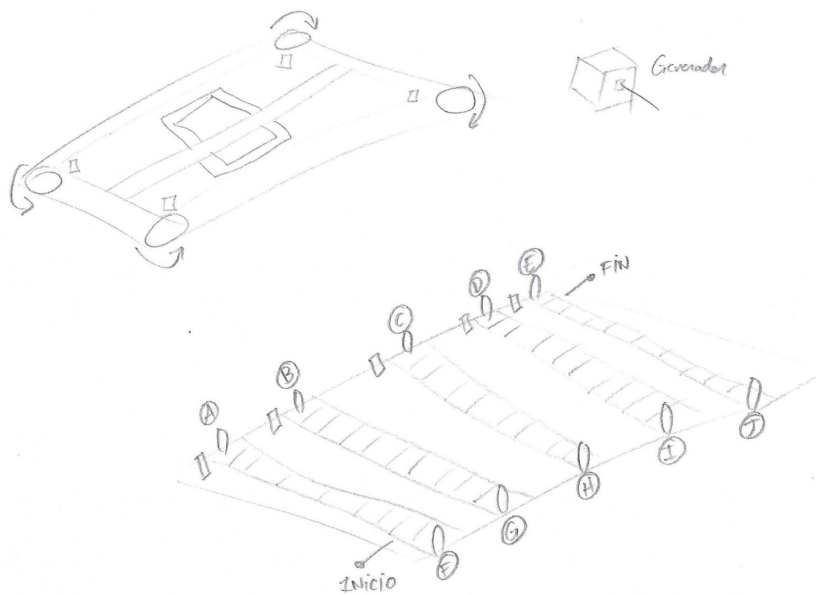


Figure B.1: Discarded puzzle mechanics

Some sketches of the puzzles found in the final version of the game:

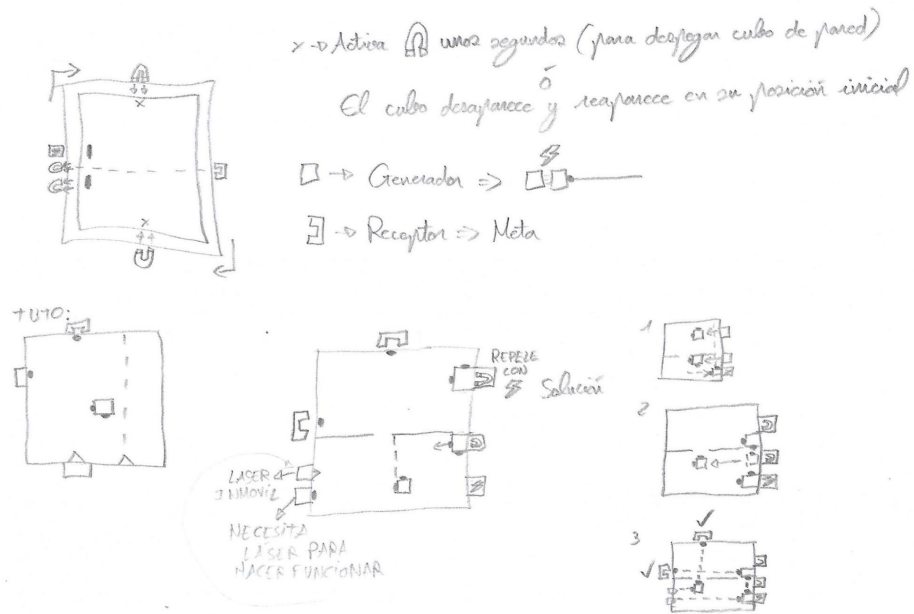


Figure B.2: Concept of some puzzle mechanics + level 2.3

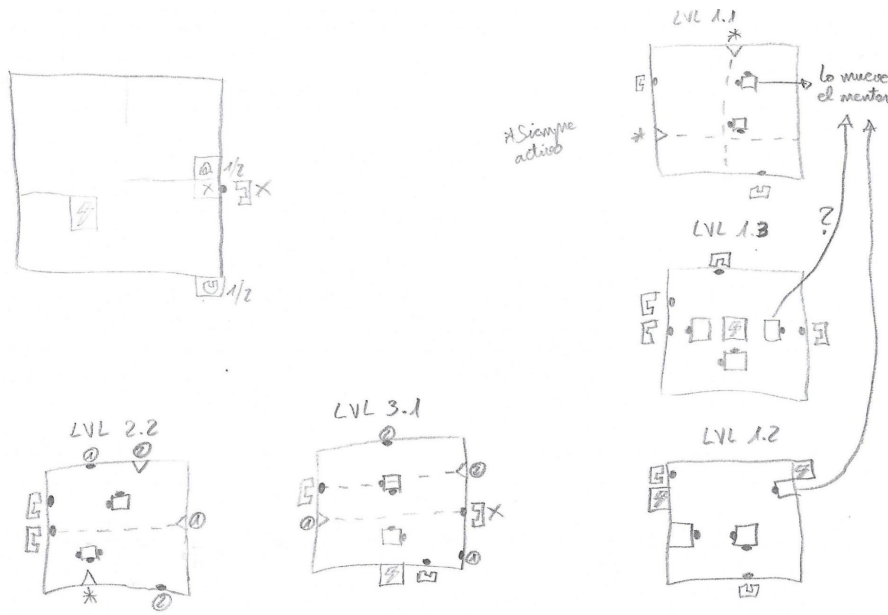


Figure B.3: Some level sketches

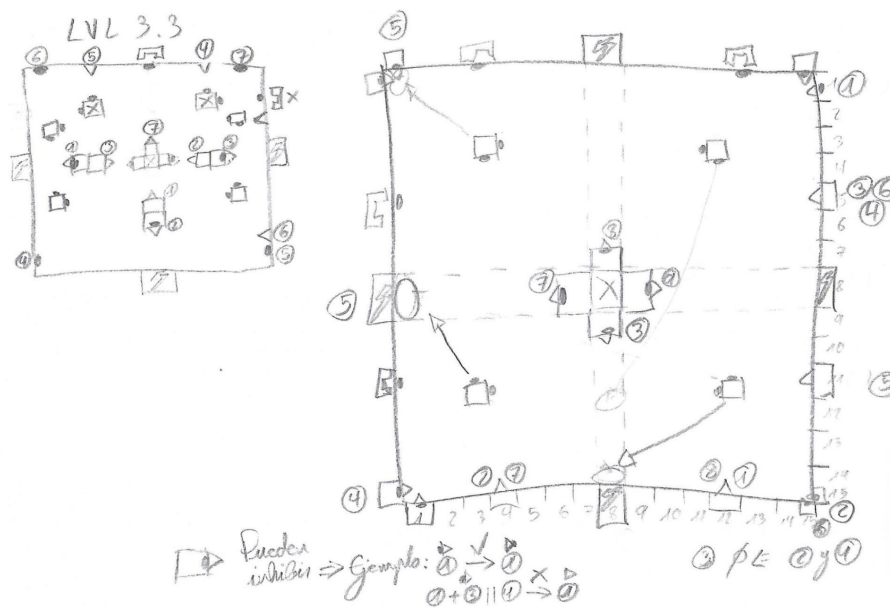


Figure B.4: Final puzzle sketch

