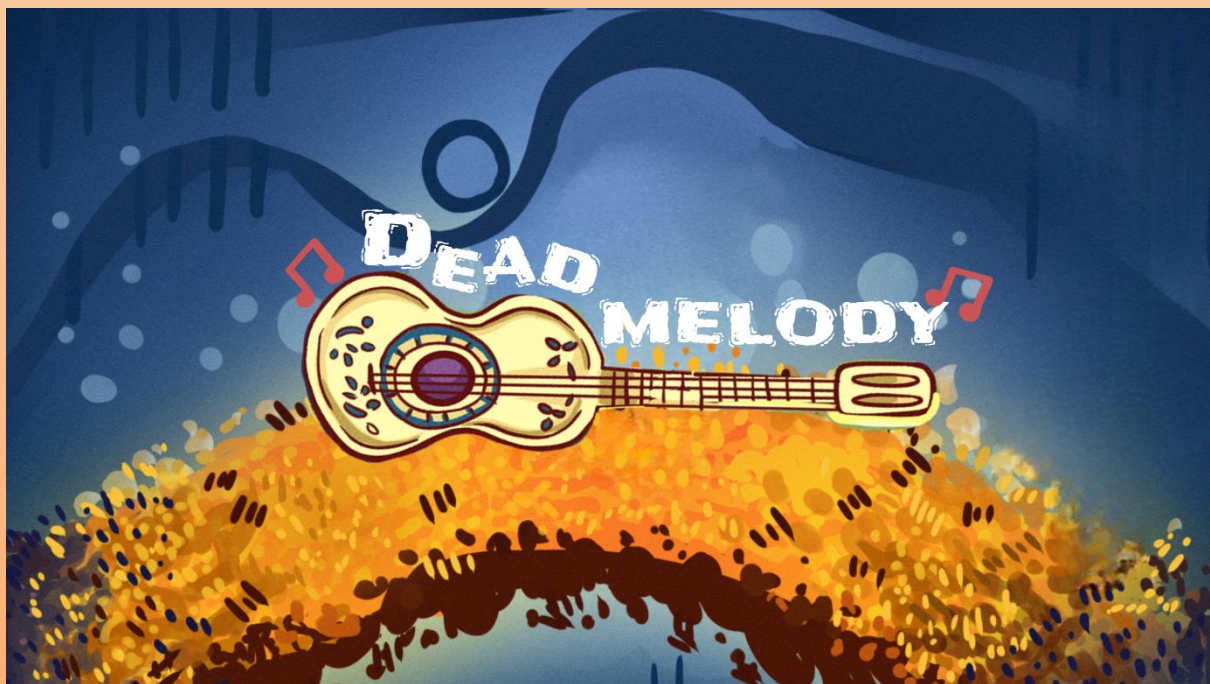


# DEAD MELODY



Lourenço Rosário – 22402755

Rafael Canhoto – 22401658

Sabrina Silva – 22402895

# Table of Contents

Change Log .....	4
Overview .....	5
Synopsis .....	5
Genre .....	5
Platform .....	5
Story .....	5
Themes .....	6
Gameplay .....	7
Design Pillars .....	7
Mechanics .....	8
Player Mechanics .....	8
Guitar Mechanic .....	10
Obstacles Mechanics (People) .....	12
Crouch Obstacle .....	12
Box/Head Obstacle .....	14
Push and Pull Obstacle .....	16
Level Design .....	18
Overview diagram by zones .....	18
Level 1 .....	19
Puzzle 1 .....	19
Puzzle 2 .....	19
Puzzle 3 .....	19
Puzzle 4 .....	20
Puzzle 5 .....	20
Puzzle 6 (Extra) .....	20
Level 2 .....	21
Puzzle 1 .....	21
Puzzle 2 .....	21
Puzzle 3 .....	21
Tileset .....	22
Menus .....	24
UI .....	27
Cutscenes .....	29

<b>Playtest</b> .....	31
<b>1° Playtest</b> .....	31
<b>Player 1</b> .....	31
<b>Player 2</b> .....	32
<b>Player 3</b> .....	33
<b>Player 4</b> .....	34
<b>Player 5</b> .....	35
<b>Player 6</b> .....	36
<b>Player 7</b> .....	37
<b>Graphs of the Questions:</b> .....	38
<b>General overview of the 1° Playtest</b> .....	39
<b>2° Playtest</b> .....	40
<b>Player 1</b> .....	40
<b>Player 2</b> .....	41
<b>Player 3</b> .....	42
<b>Player 4</b> .....	43
<b>Player 5</b> .....	44
<b>Player 6</b> .....	45
<b>Player 7</b> .....	46
<b>Graphs of the Questions:</b> .....	47
<b>General overview of the 2° Playtest</b> .....	48
<b>3° Playtest</b> .....	49
<b>Player 1</b> .....	49
<b>Player 2</b> .....	50
<b>Player 3</b> .....	51
<b>Player 4</b> .....	52
<b>Player 5</b> .....	53
<b>Player 6</b> .....	54
<b>Player 7</b> .....	55
<b>Player 8</b> .....	56
<b>Player 9</b> .....	57
<b>Player 10</b> .....	58
<b>Player 11</b> .....	59
<b>Player 12</b> .....	60
<b>Player 13</b> .....	61
<b>Graphs of the Questions:</b> .....	62

<b>Notes:</b> .....	64
<b>Player 1:</b> .....	64
<b>Player 2:</b> .....	64
<b>Player 3:</b> .....	64
<b>Player 4:</b> .....	64
<b>Player 5:</b> .....	64
<b>Player 6:</b> .....	65
<b>Player 7:</b> .....	65
<b>Player 8:</b> .....	65
<b>Player 9:</b> .....	65
<b>Player 13:</b> .....	65
<b>General overview of the 3<sup>o</sup> Playtest</b> .....	66
<b>Asset list</b> .....	67
<b>Art</b> .....	67
<b>Programmer</b> .....	68
<b>CameraFollow.cs</b> .....	68
<b>ButtonManager.cs</b> .....	69
<b>LastLevel.cs</b> .....	70
<b>Level5.cs</b> .....	71
<b>Mud.cs</b> .....	72
<b>NextLevel.cs</b> .....	73
<b>NextScene.cs</b> .....	74
<b>NextText.cs</b> .....	75
<b>OpenMenu.cs</b> .....	76
<b>RestartLevel.cs</b> .....	77
<b>TextTimer.cs</b> .....	78
<b>Obstacle.cs</b> .....	79
<b>CloudAnimParallax.cs</b> .....	80
<b>ParticleSystem.cs</b> .....	81
<b>PlayerGuitarTrigger.cs</b> .....	82
<b>TheGuitarAbility.cs</b> .....	83
<b>SoundManager.cs</b> .....	84

## Change Log

When	Who	What
12/03/2025	Rafael Canhoto	Create GDD
13/03/2025	Rafael Canhoto	Added player and enemy diagrams. Added Level 1 breakdown.
13/03/2025	Sabrina Silva	Added art asset list.
14/03/2025	Sabrina Silva	Added themes and pillars.
14/03/2025	Lourenço Rosário	Added programmer asset list
15/03/2025	Rafael Canhoto	Updated player and enemy diagrams. Added guitar diagram. Updated game breakdown. Added synopsis, genre, platform and history.
16/03/2025	Rafael Canhoto	Updated history, player and guitar mechanics Update game breakdown
16/03/2025	Lourenço Rosário	Updated programmer asset list
17/03/2025	Rafael Canhoto	Updated themes and design pillars
18/03/2025	Rafael Canhoto	Updated Level Design Section
21/03/2025	Rafael Canhoto	Updated Level Design Section and add Playtest forms
21/03/2025	Sabrina Silva	Updated playtest section
29/03/2025- 31/03/2025	Rafael Canhoto	Update Design Pillars, Mechanics and Level Design
04/05/2025	Lourenço Rosário	Update Programming section
04/05/2025	Rafael Canhoto	Update Obstacles images
21/05/2025	Rafael Canhoto	Update synopsis, story, themes and game pillars
28/05/2025	Lourenço Rosário	Update Programming section and Mechanic section
29/05/2025	Rafael Canhoto	Update and finalize the GDD

# **Overview**

## **Synopsis**

Step into the shoes of Miguel from the Rivera family, a passionate young musician in a world where music is forbidden in his household.

Even though his family has banned music, Miguel can't ignore the rhythm in his soul, he plays his guitar to bring joy, inspire and bring people together.

To help him follow his dream, you'll take on musical challenges and master enchanting songs, using the power of music to touch hearts and change the world.

## **Genre**

2D side scroller, puzzle, musical.

## **Platform**

Windows PC, macOS.

## **Story**

The Rivera family is known as the finest shoemaker in Santa Cecilia. Every member of the family, from a young age, learns the craft, starting with small tasks like shining shoes.

Miguel, however, has a secret: he dreams of creating music, not shoes. But in the Rivera household, music is strictly forbidden due to a painful past involving his great-great-grandmother and a musician who abandoned the family.

One day, while shining a mariachi's shoes, Miguel shares his dream and family history. Impressed, the mariachi challenges him to play the guitar. Just as Miguel is about to perform, his grandmother and other relatives catch him. She scolds both Miguel and the mariachi, Miguel escapes, taking the mariachi's guitar with him.

After a long chase through the town, Miguel ends up in the cemetery. He visits the tomb of Ernesto De La Cruz, a legendary musician whom Miguel believes to be his great-great-grandfather, the only family member who might understand his passion for music. As Miguel strums the guitar, something magical happens, he crosses into the Land of the Dead, without realizing it.

When he notices he's no longer in the World of the Living, Miguel panics and tries to escape the cemetery unnoticed. Instead, he runs into his deceased ancestors. At first, they try to help him return home, but Miguel soon realizes they share the same "anti-music" beliefs as the rest of the family. Determined to follow his dreams, he runs away again.

During his escape, Miguel meets a stranger named Hector, a charming trickster who claims to know De La Cruz. Hector agrees to help Miguel reach the famous musician, on one condition: Miguel must help Hector be remembered in the World of the Living, so he doesn't fade away forever.

As they journey together, Miguel and Hector finally reach Ernesto De La Cruz. But the truth is revealed: De La Cruz was a fraud who stole Hector songs and betrayed him. It turns out that Hector, not Ernesto, is Miguel's real great-great-grandfather.

Meanwhile, Miguel's deceased relatives, still in search of him, uncover the truth and join forces to confront De La Cruz during a music competition in the Land of the Dead.

In the end, Miguel and his family defeated De La Cruz. The family returns to their respective worlds, Miguel to the World of the Living, now free to pursue his musical dreams, and his ancestors to the Land of the Dead, united once again with a "new" member in the family, Hector.

With the truth revealed, the Rivera family finds healing and happiness. Miguel follows his heart, music is welcomed into the family, and Hector is remembered, rejoining his loved ones in spirit and song.

## Themes

- **Family:** The game highlights the importance of family. The story reminds us to honor loved ones.
- **Death:** Inspired by Death Day in Mexico, death doesn't mean the end of life or connections, remembering the family and the loved ones who have already passed away but keeping the memories alive.
- **Music and Dreams:** Miguel is passionate about music and dreams about being a musician but to do so he confronts his Abuelita and great-great-grandmother Victoria.
- **Forgiveness:** Throughout the story, there are moments of forgiveness and understanding between families showing that it is possible to solve broken bonds.

# Gameplay

## Design Pillars

- **Family:** Through the narrative we will show how important family is in this game.
- **Puzzle:** We will have several obstacles along our path that will react differently when they hear the music of our guitar.
- **Music:** Through our guitar and the songs we play, we will make the world happier and solve the challenges it presents.



## Mechanics

### Player Mechanics

Miguel (player) is designed to be small and agile, he can move in the horizontal axis, jump and play his guitar.

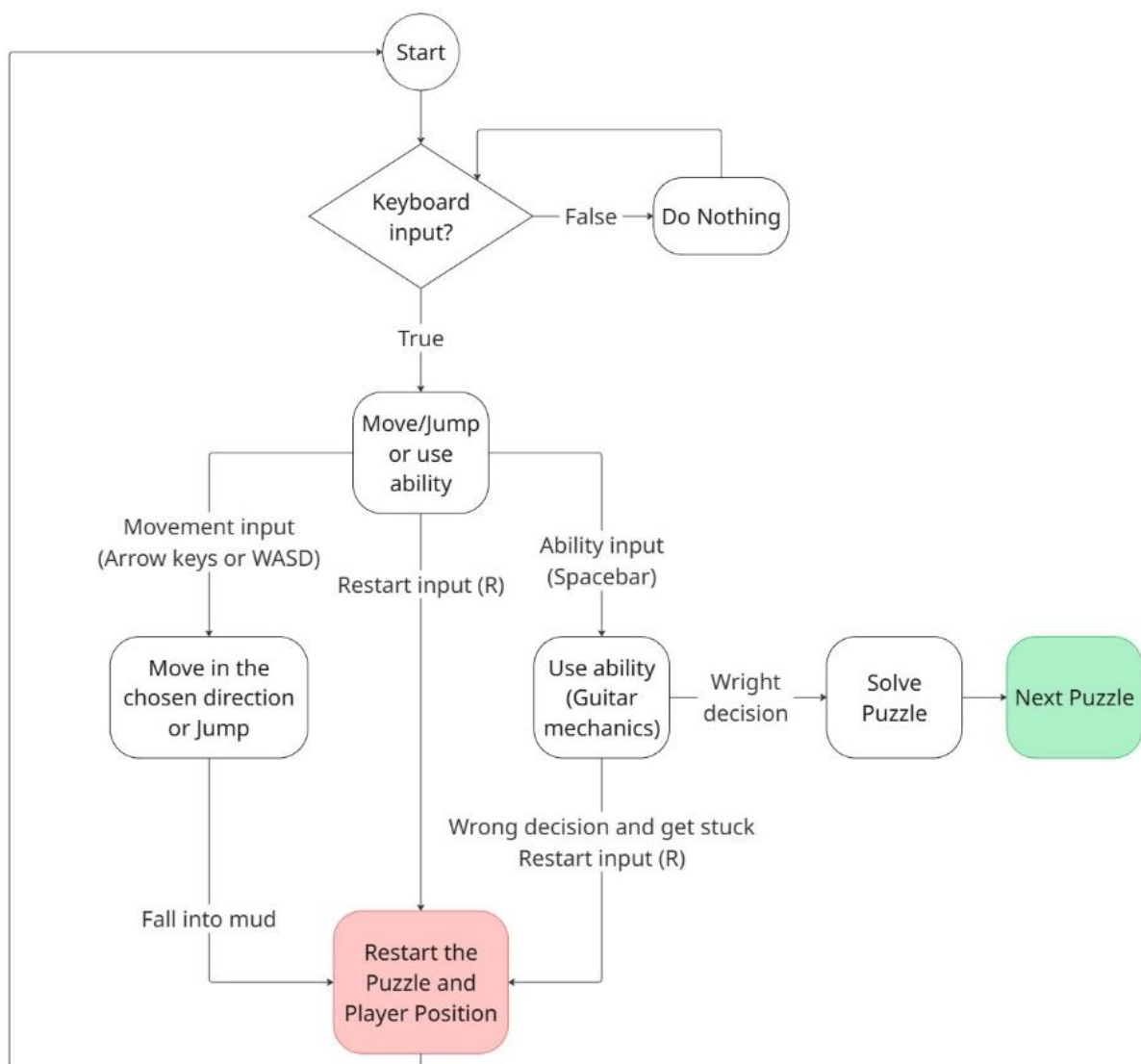
He is the only character of this size because he is a child.

Movement (Left and Right Arrows or AD)	Movement velocity: 125
Jump (Up Arrow or W)	Jump velocity: 125
Jump variation	Max duration: 014; Gravity scale: 1
Play guitar	Spacebar or Left mouse button
Character size	24x24

#### Sprite Sheet of the player



## Flowchart



[Link to open Flowchart](#)

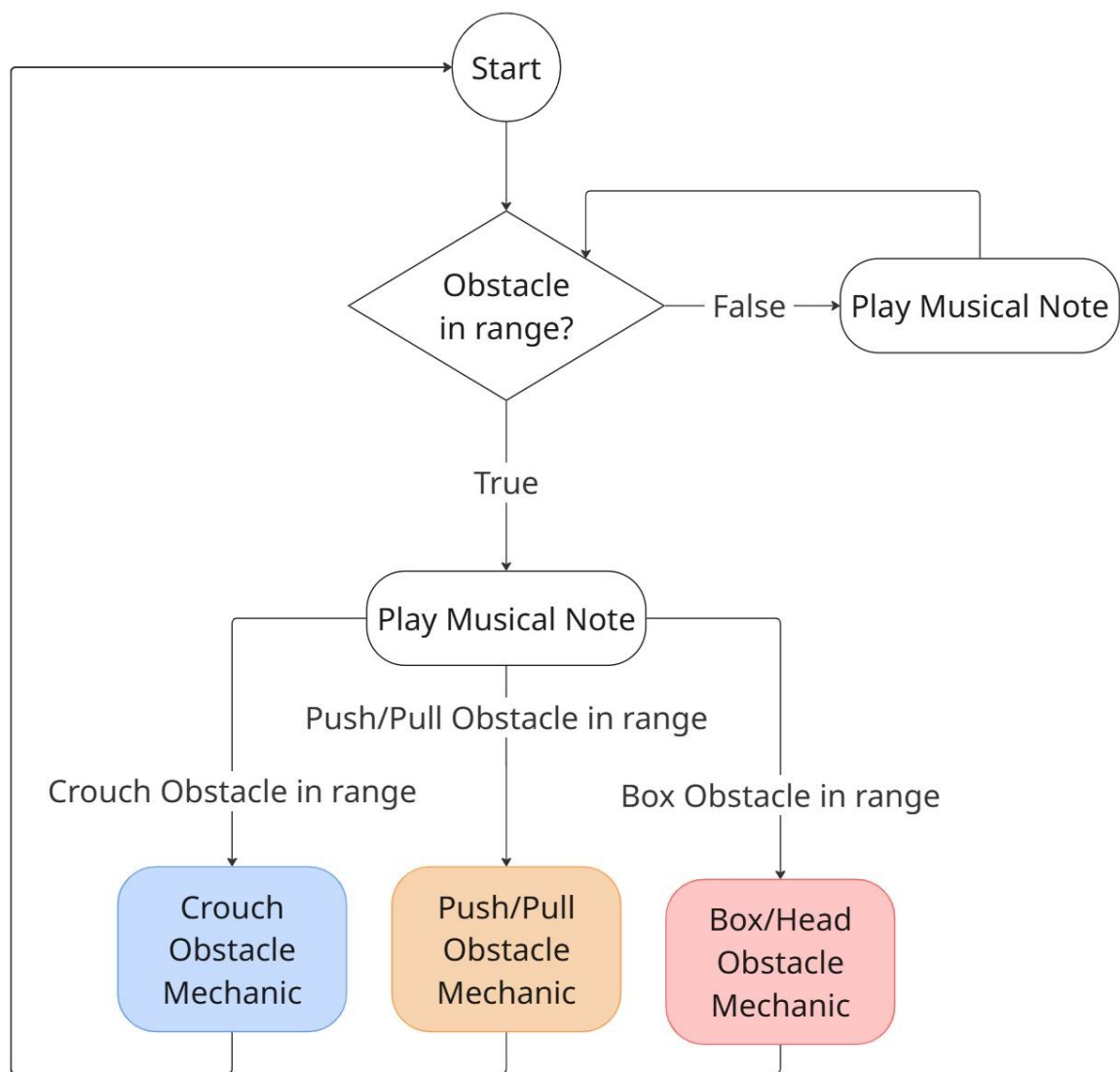
## Guitar Mechanic

Miguel’s instrument. He uses it to charm people along the way (Obstacles) and to make the world a happier place.

Ability Range	45
Particle System Range	45

### Sprite sheet of the mechanic



Flowchart[Link to open Flowchart](#)

## Obstacles Mechanics (People)

Obstacles appear in different forms and shapes to difficult Miguel 's life during his journey. Below you will find all the obstacles and what each one does.

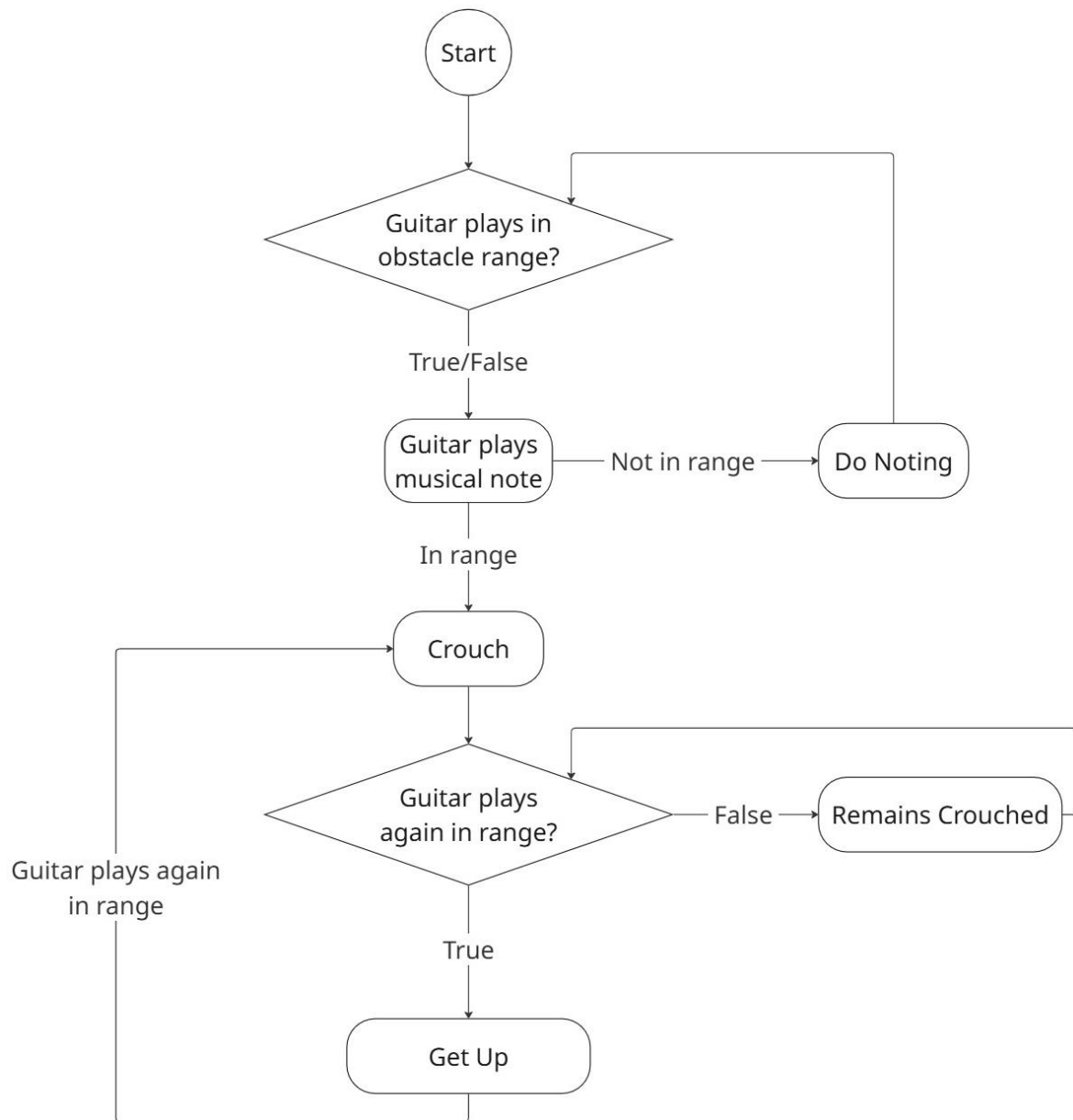
### Crouch Obstacle

The Crouch Obstacle is used by Miguel as a platform to reach higher places. This is achieved by Miguel playing the guitar within range and the object lowering itself to a point where Miguel can climb them and if Miguel plays the guitar within range again, they get up.

Old woman size	32x26
Skeleton size	32x32

Sprite sheet of the Crouch obstacle



Flowchart[Link to open Flowchart](#)

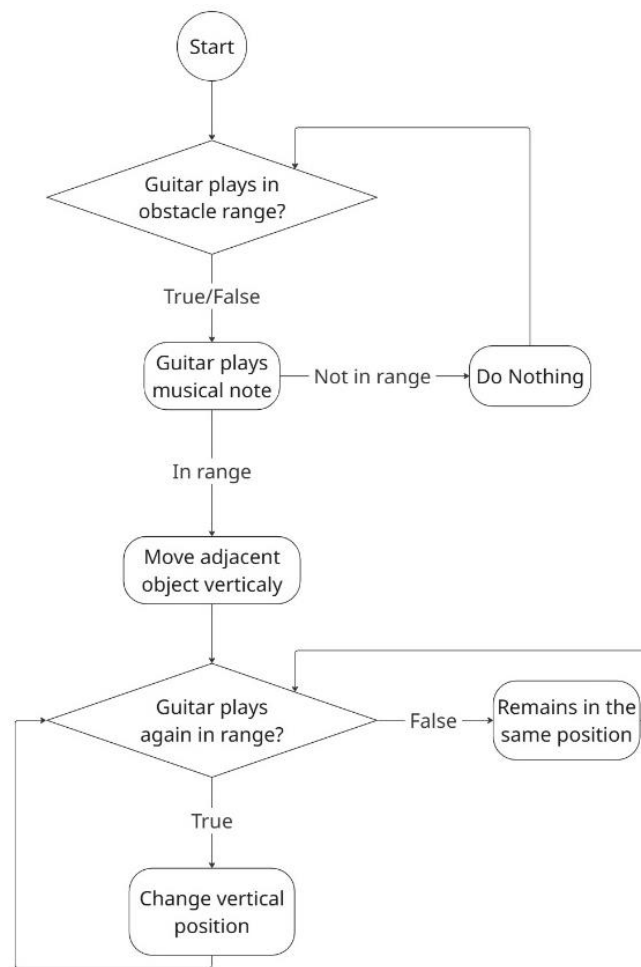
## Box/Head Obstacle

The Box/Head Obstacle is used by Miguel as a platform to reach higher places and a little bit further. This is achieved by Miguel playing the guitar within range and the object moves the box up or detaches the head down to a point where Miguel can climb the head and get the box down or the head to his start place if Miguel plays the guitar within range again.

Mariachi size	32x
Skeleton size	32x32

### Sprite sheet of the Box/Head Obstacle



Flowchart[Link to open Flowchart](#)



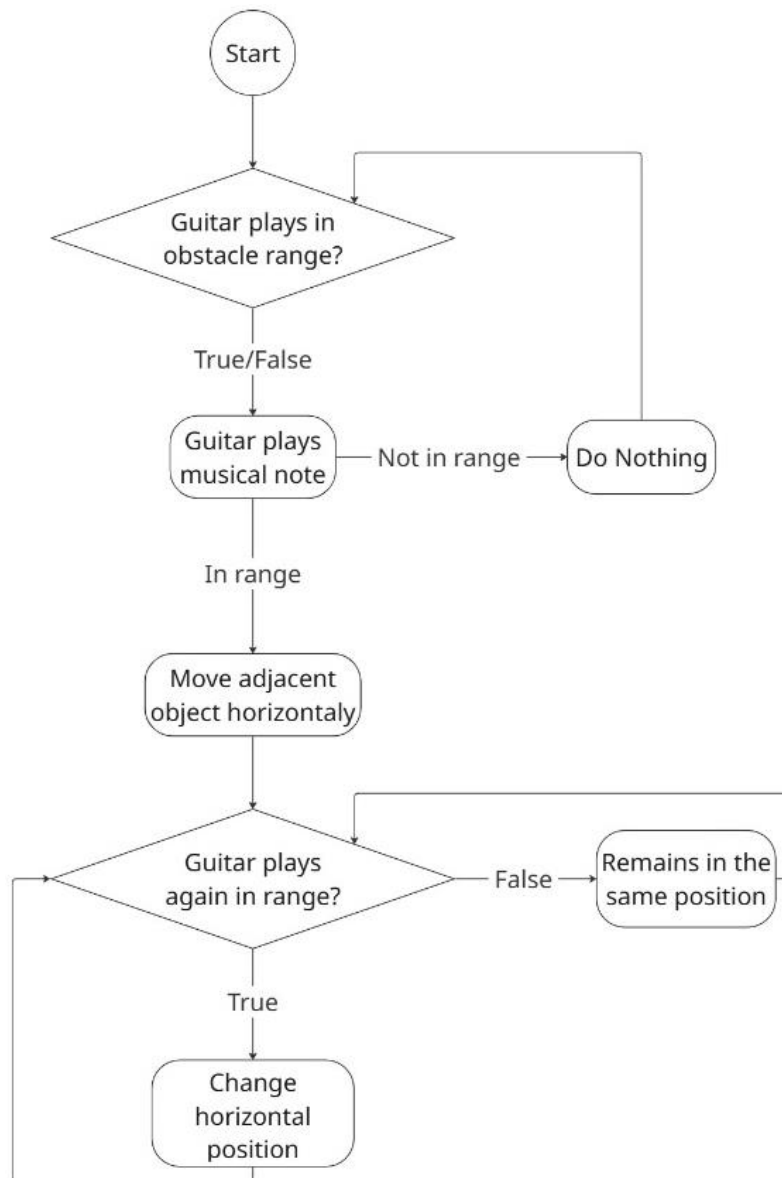
## Push and Pull Obstacle

The Push and Pull Obstacle is used by Miguel as a platform to reach further places. This is achieved by Miguel playing the guitar within range and the Obstacle kick the box away in front of him and he can pull the box if Miguel plays the guitar within range again.

Woman size	32x32
Skeleton size	32x32

### Sprite sheet of the Push and Pull Obstacle

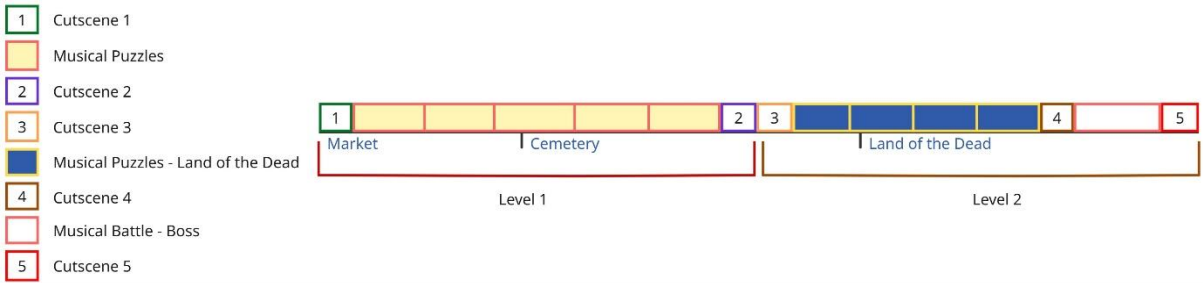


Flowchart[Link to open Flowchart](#)

# Level Design

## Overview diagram by zones

This is our idea of the *Dead Melody* breakdown. We thought of breaking each level into smaller puzzles to make them easier for players to follow.



[Link to open the Game breakdown](#)

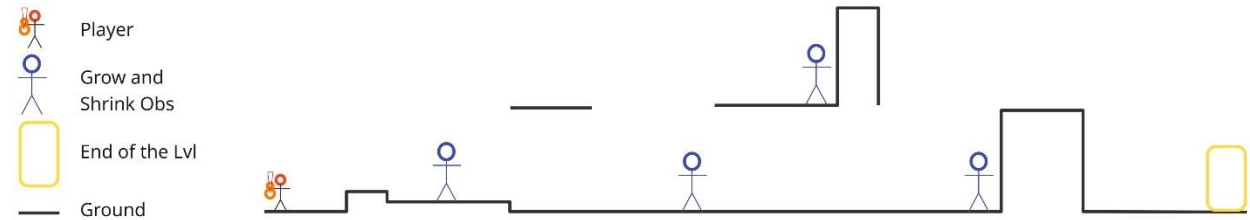
## Level 1

This is the introductory level of the game, easy puzzles and low difficulty curve.

### Puzzle 1

In this puzzle we introduce crouch obstacles.

#### Symbology:

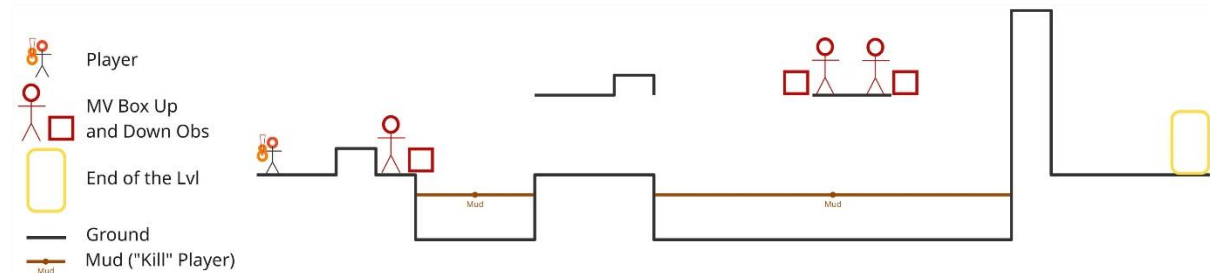


[Link to open the Puzzle](#)

### Puzzle 2

In this level we introduce the Box/Head Obstacle.

#### Symbology:

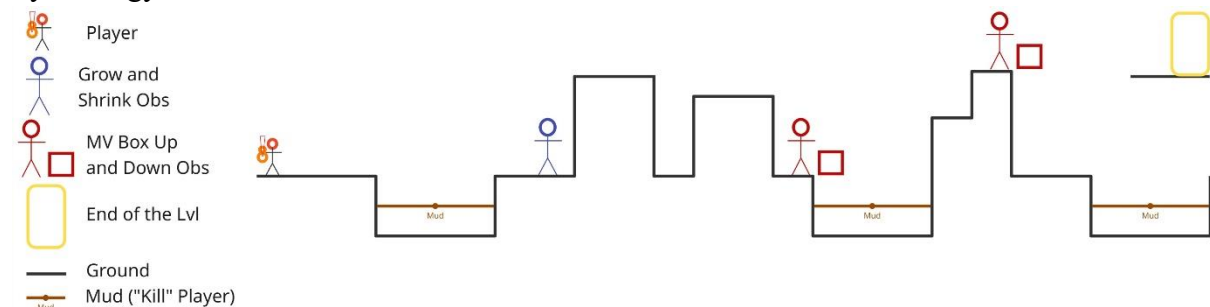


[Link to open the Puzzle](#)

### Puzzle 3

From this puzzle we start to mix these two obstacles and make more complex puzzles.

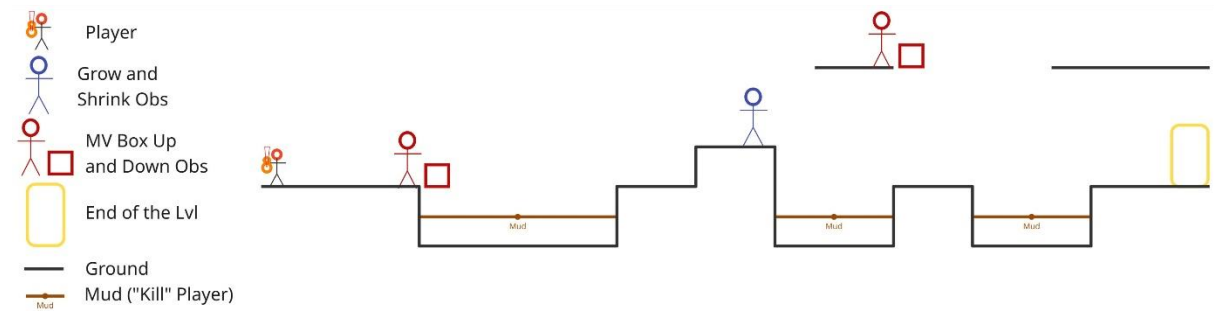
#### Symbology:



[Link to open the Puzzle](#)

## Puzzle 4

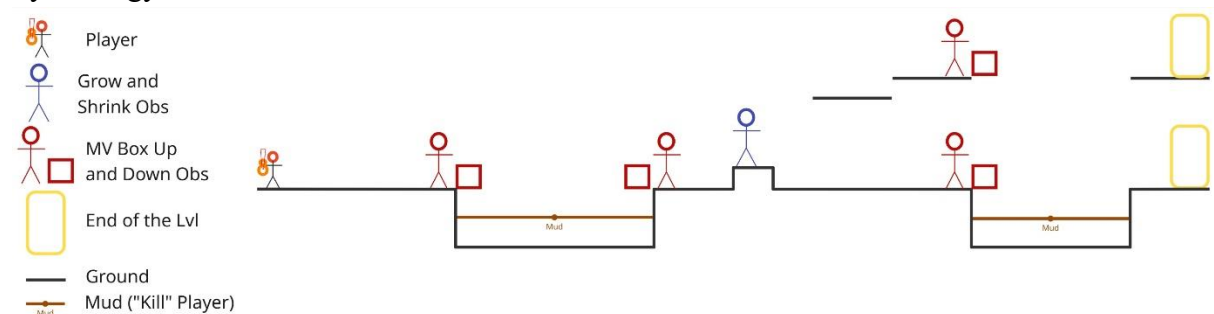
### Symbology:



[Link to open the Puzzle](#)

## Puzzle 5

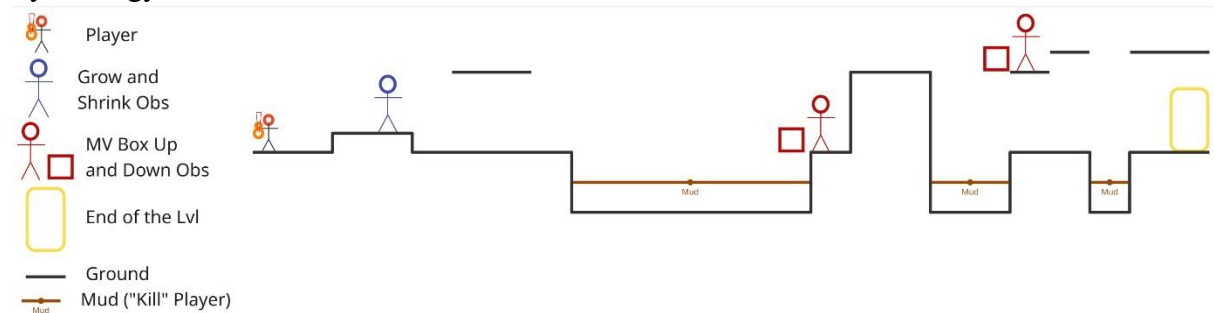
### Symbology:



[Link to open the Puzzle](#)

## Puzzle 6 (Extra)

### Symbology:



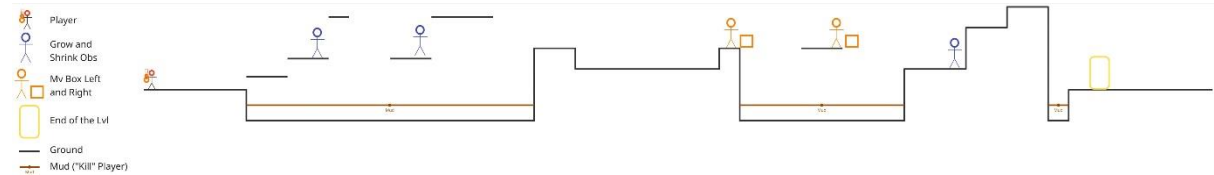
[Link to open the Puzzle](#)

## Level 2

In this level we use the same logic of the first level but with bigger puzzles.

### Puzzle 1

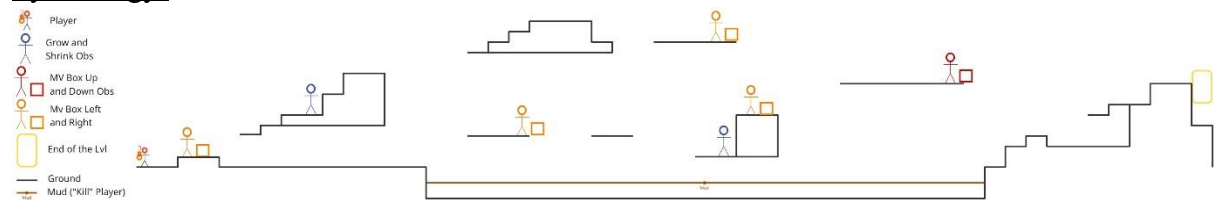
#### Symbology:



[Link to open the Puzzle](#)

### Puzzle 2

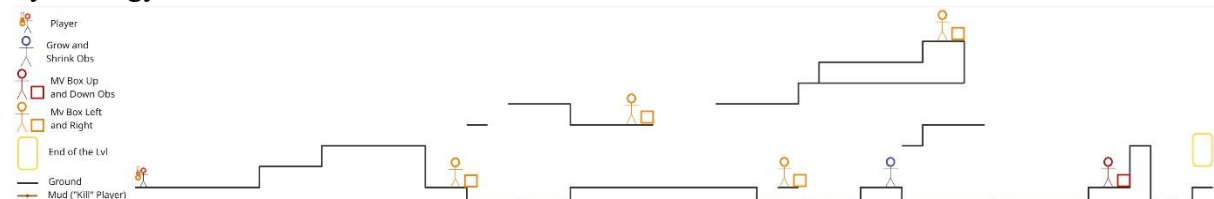
#### Symbology:



[Link to open the Puzzle](#)

### Puzzle 3

#### Symbology:



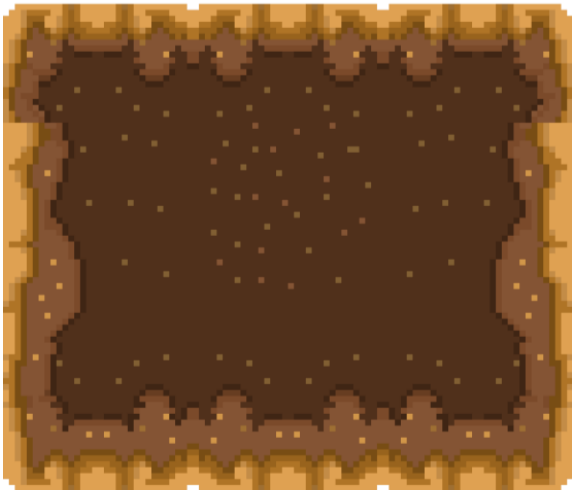
[Link to open the Puzzle](#)

# Tileset

The tileset will feature a cartoon style closely matching that of the film, allowing players to immediately recognize and connect with the visual aesthetic.

Foreground and mud tileset size	16x16
Background tileset	32x32

## Foreground and Mud



This image displays a comprehensive set of 2D pixel art assets for a game, organized into two main sections. The top section features various terrain and building components, including two large rectangular blocks with brown roofs and tan walls, a large blue rectangular block with a brown roof, and several smaller decorative elements like trees, flowers, and fences. The bottom section shows a variety of furniture and decorative items, such as a large pinkish-brown archway, a small white cross, a large pinkish-brown building with a white cross, and several smaller decorative elements like vases, flowers, and fences.

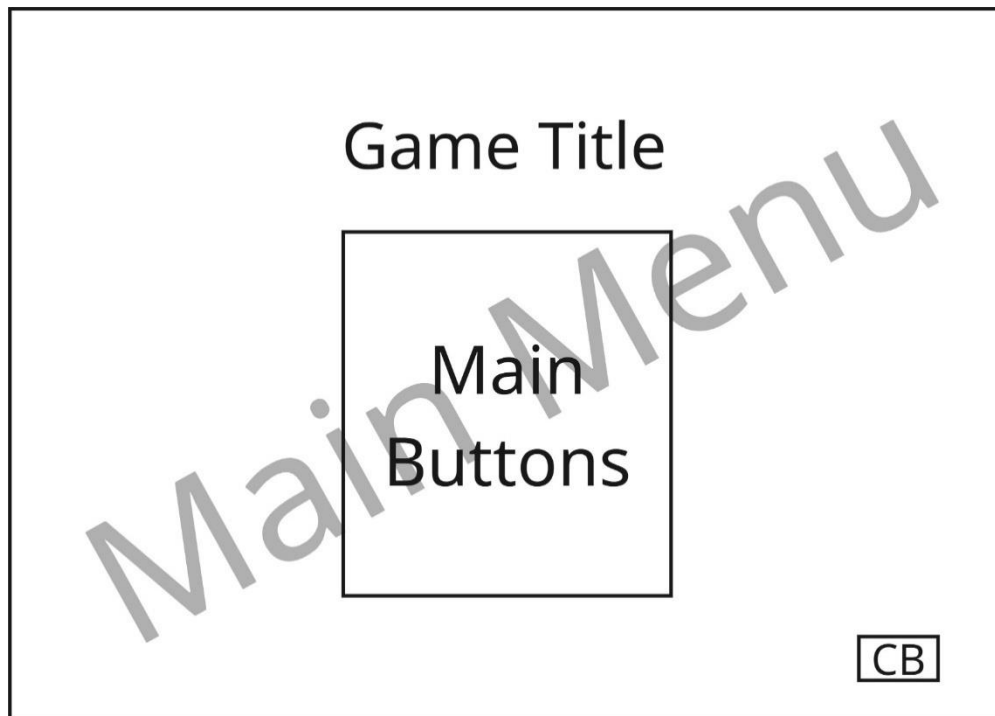


# Menus

For this game, we thought of a simple menu that wouldn't be too complicated to understand.

## Main Menu

Main Menu/Screen size	1920x1080
Game Title size	200
Main Buttons size	500x100 / 450x100 / 400x100
“Cheat” Butto (CB) size	200x100

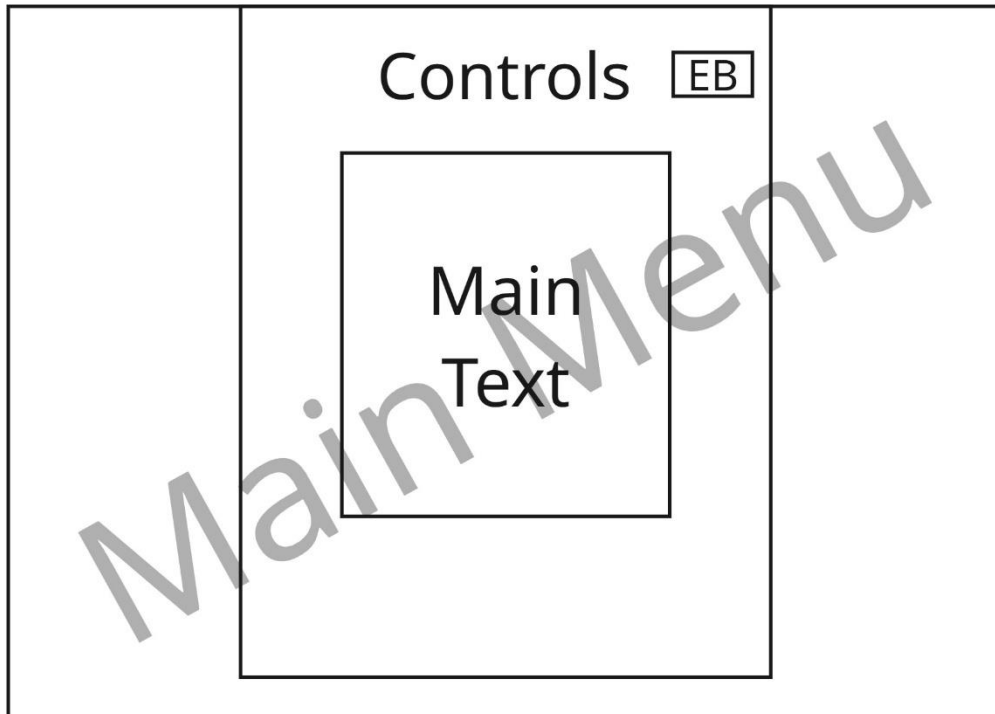


## Result

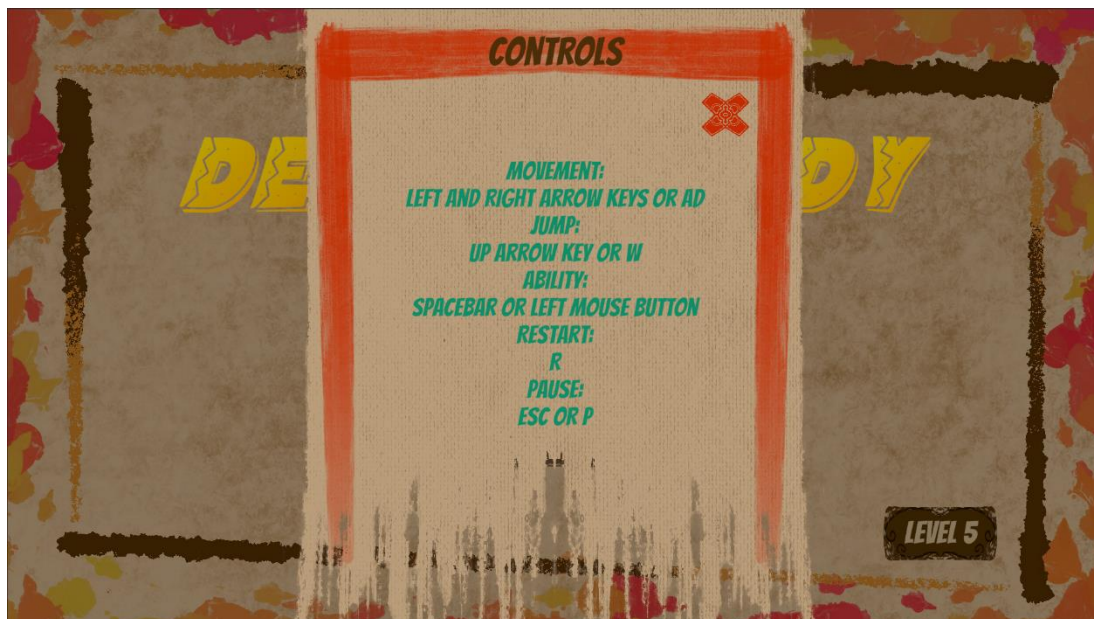


### Controls Menu

Main Menu size	1920x1080
Controls Menu size	1320x1080
Title size	70
Main Text size	45
Exit Button size (EB)	90x90

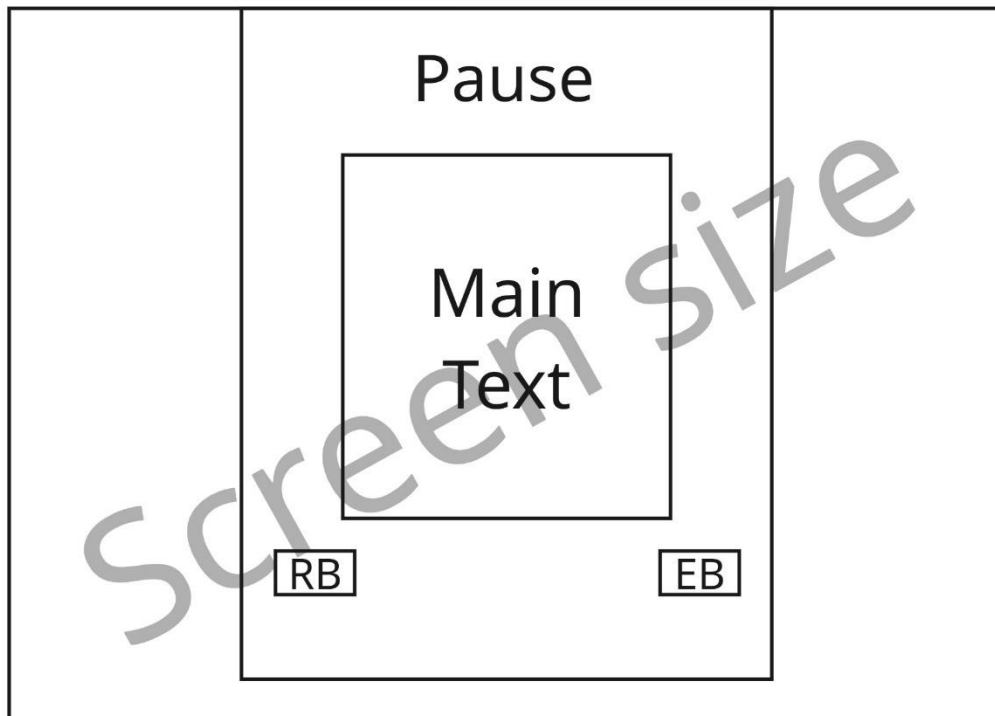


### Result

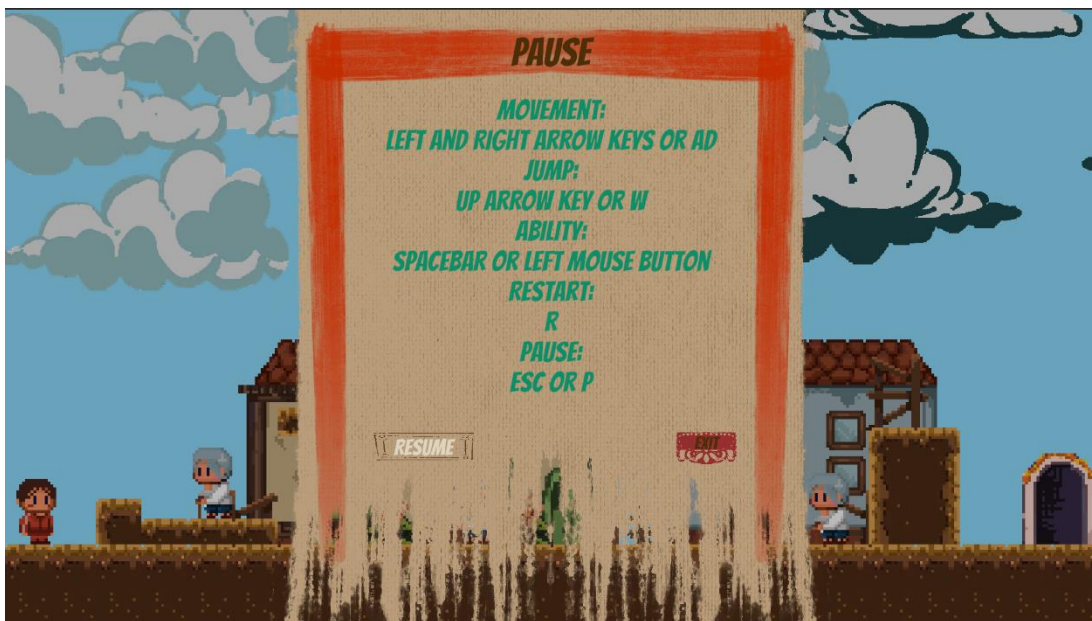


### Pause Menu

Screen size	1920x1080
Pause Menu size	1350x1080
Title size	70
Text size	50
Resume Button size (RB)	210x60
Exit Button size (EB)	110x60



### Result

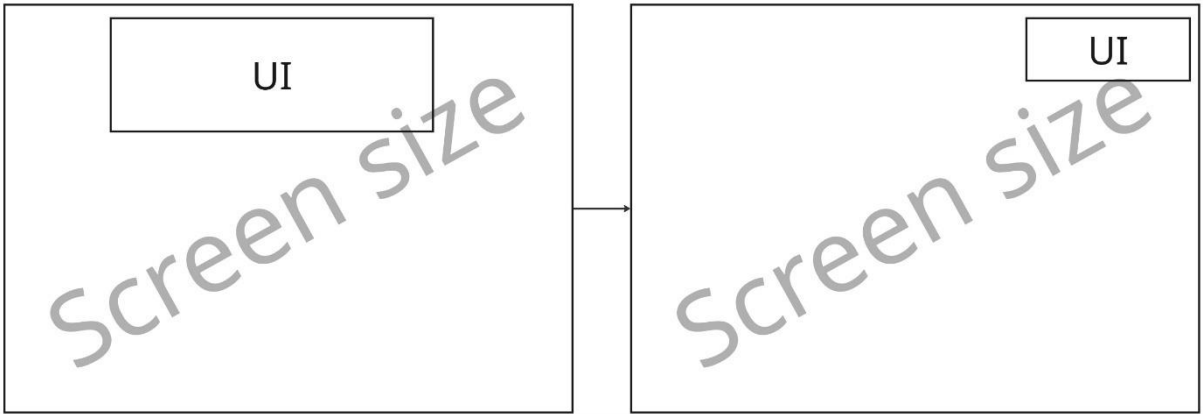


# UI

To enhance player immersion, the game will feature minimal UI. Only a text box with the controls will appear, which will decrease in size after a short time. The entire narrative will unfold through in-game cutscenes, allowing the story to be experienced organically and without distraction.

## Ui in game

Screen size	1920x1080
Time	7 seconds
Text box size (before/after)	1215x270 / 700x150
Text size (before/after)	43.4 / 25





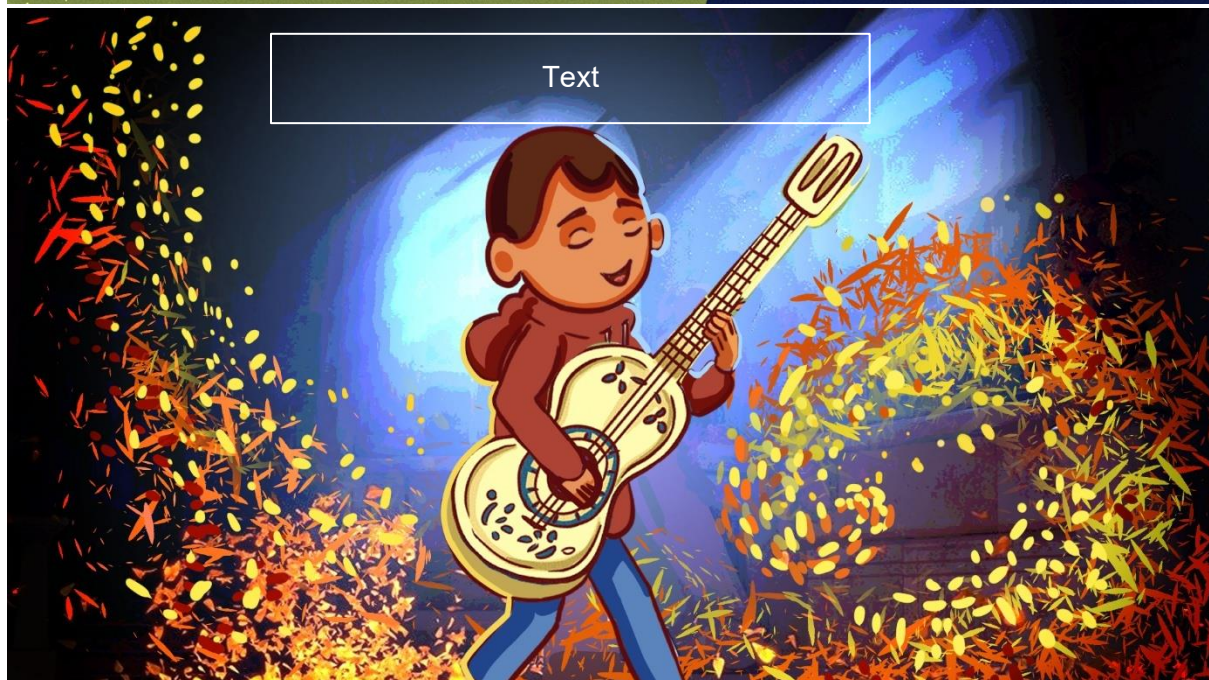
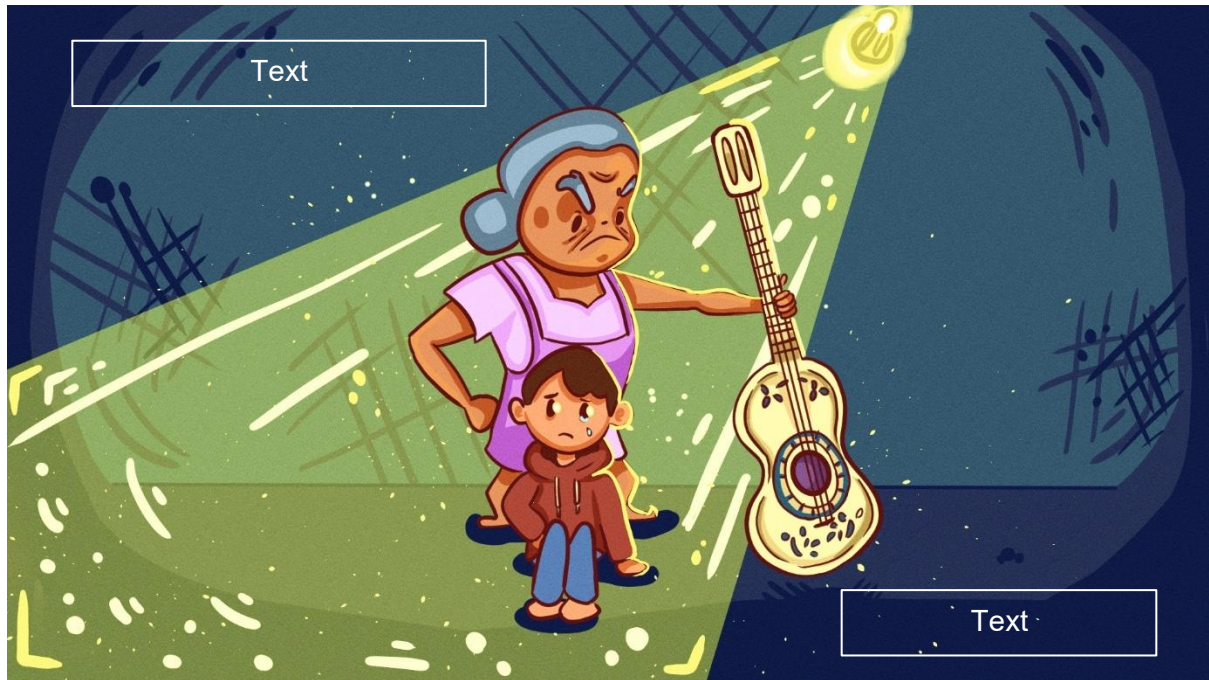
## Result



## Cutscenes

The story will be told through carefully crafted splash art cutscenes that illustrate important moments. Players will see them at the beginning and end of the game, and again between the first and second levels to bridge the narrative.

Cutscene size	1920x1080
Text size	36







# **Playtest**

## **1° Playtest**

### **Player 1**

#### **Profile Questions:**

**What is your age?**

22-27

**How many hours do you play a week?**

7h+

**What type of games do you play?**

RPG, Action, Adventure, Platform, Puzzle, Horror/Suspense.

#### **Game Questions:**

**Do you like the mechanics?**

Yes

**If you like it, tell me why?**

I like parkour and reaction mechanics.

**If you didn't like it tell me why?**

**Any additional feedback?**

Being able to stand on the platforms.

**Notes:**

None



## **Player 2**

### **Profile Questions:**

**What is your age?**

16-21

**How many hours do you play a week?**

4h-5h

**What type of games do you play?**

RPG

### **Game Questions:**

**Do you like the mechanics?**

Sim

**If you like it, tell me why?**

Tem que apertar o botão da música para abrir as portas e isso faz com que tenha uma dificuldade em manter o ritmo entre os pulos e a abertura das portas.

**If you didn't like it tell me why?**

**Any additional feedback?**

Coyote-time muito generoso, falta de feedback nas animações, o jogador fica preso na porta se ele correr em sua direção, pode usar o Koyote-time mesmo depois de ficar preso na porta.

**Notes:**

None

## **Player 3**

### **Profile Questions:**

**What is your age?**

22-27

**How many hours do you play a week?**

7h+

**What type of games do you play?**

RPG, Action, MMO

### **Game Questions:**

**Do you like the mechanics?**

Fun mechanic.

**If you like it, tell me why?**

The mechanic feels fluid and easy to grasp.

**If you didn't like it tell me why?**

**Any additional feedback?**

The audio is a bit loud, getting stuck to the wall sounds like a fun idea to implement and could use some jumps designed around it. Fun prototype GJ.

**Notes:**

None

## **Player 4**

### **Profile Questions:**

**What is your age?**

16-21

**How many hours do you play a week?**

5h-6h

**What type of games do you play?**

RPG, Action, Puzzle.

### **Game Questions:**

**Do you like the mechanics?**

Yes

**If you like it, tell me why?**

Challenging platforming and tight windows for inputs make it fun to go through, even when failing.

**If you didn't like it tell me why?**

**Any additional feedback?**

The audio is a bit loud, getting stuck to the wall sounds like a fun idea to implement and could use some jumps designed around it. Fun prototype GJ.

**Notes:**

None

## **Player 5**

### **Profile Questions:**

**What is your age?**

16-21

**How many hours do you play a week?**

6h-7h

**What type of games do you play?**

RPG, Action, Puzzle.

### **Game Questions:**

**Do you like the mechanics?**

Yes

**If you like it, tell me why?**

Challenging platforming and tight windows for inputs make it fun to go through, even when failing.

**If you didn't like it tell me why?**

**Any additional feedback?**

**Notes:**

None

## Player 6

### Profile Questions:

**What is your age?**

16-21

**How many hours do you play a week?**

0h-1h

**What type of games do you play?**

RPG, Action, Platform, Puzzle, Horro/Suspense

### Game Questions:

**Do you like the mechanics?**

Yes

**If you like it, tell me why?**

I think it's a fun mechanic, needing a bit of skill to be able to make the jump while still opening the door is a really cool idea and I think it can be improved upon.

**If you didn't like it tell me why?**

**Any additional feedback?**

Please make the guitar strum something actually listenable, pick a chord, any chord as long as it sounds good, this one isn't that good and might get annoying.

**Notes:**

None

**Player 7****Profile Questions:**

**What is your age?**

34-39

**How many hours do you play a week?**

1h-2h

**What type of games do you play?**

RPG, Action, Adventure, Platform, Puzzle, Sports, Horror/Suspense, Sims.

**Game Questions:**

**Do you like mechanics?**

Yes

**If you like it, tell me why?**

**If you didn't like it tell me why?**

**Any additional feedback?**

**Notes:**

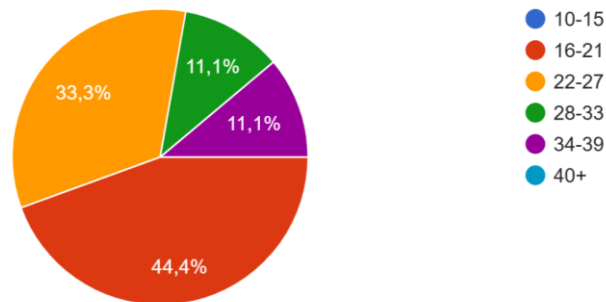
None

## Graphs of the Questions:

\*This graphs have two more people because we send the wrong link to the second playtest players

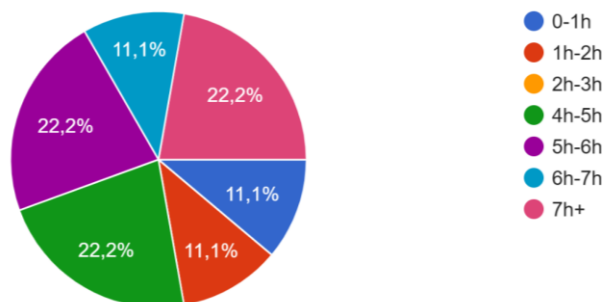
What is your age?

9 respostas



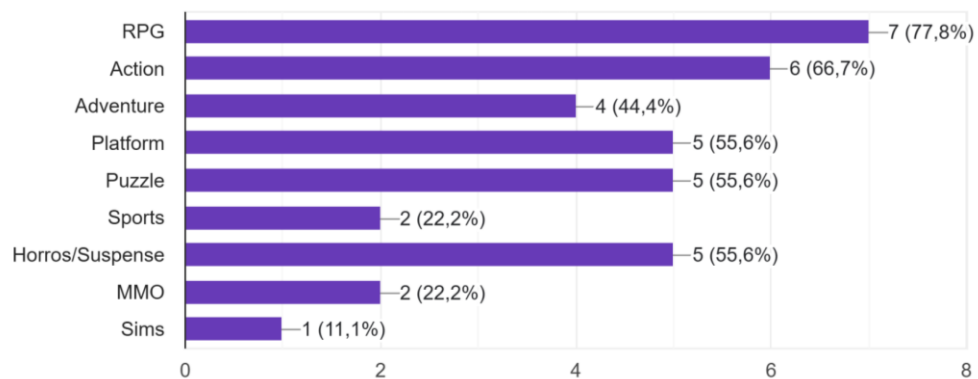
How many hours do you play per week?

9 respostas



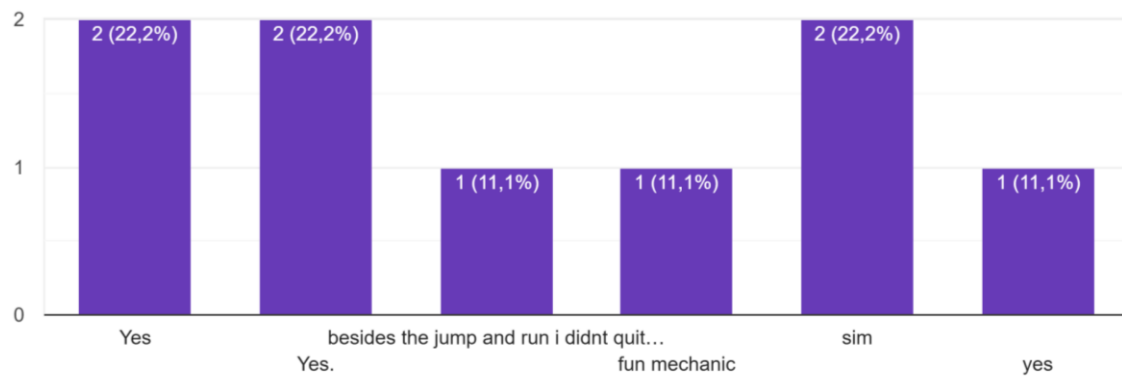
what type of games do you play?

9 respostas



Do you like the mechanics?

9 respostas



## General overview of the 1º Playtest

- The game has nothing to do with the movie
- The idea of the mechanics is very difficult to use and confusing
- We have to work better on our ideas around the mechanics



## 2º Playtest

### Player 1

#### Profile Questions:

What is your age?

28-33

How many hours do you play a week?

4h-5h

What type of games do you play?

Action, Sports.

#### Game Questions:

How hard was it to understand the mechanic?

Besides the jump and run I didn't quite get the rest.

Can you relate this game mechanic with the movie "Coco"?

Yes

Any additional feedback?

I liked the part where you can decrease and increase the obstacles.

Notes:

Fix colliders on green obstacles when they are in small and large sizes.

## **Player 2**

### **Profile Questions:**

**What is your age?**

22-27

**How many hours do you play a week?**

5h-6h

**What type of games do you play?**

Adventure, Platform, Horror/Suspense, MMO.

### **Game Questions:**

**How hard was it to understand the mechanic?**

4

**Can you relate this game mechanic with the movie "Coco"?**

3

**Any additional feedback?**

Why do objects lose the collider when they're small? Is it supposed to work that way?  
I don't think it makes much sense when that happens.

**Notes:**

It shouldn't be possible to climb over obstacles when they are big.

**Player 3****Profile Questions:**

**What is your age?**

21-26

**How many hours do you play a week?**

1h-2h

**What type of games do you play?**

Mobile games

**Game Questions:**

**How hard was it to understand the mechanic?**

4

**Can you relate this game mechanic with the movie "Coco"?**

3

**Any additional feedback?**

I think there could have been more references to the movie, but it was fun.

**Notes:**

None

**Player 4****Profile Questions:**

**What is your age?**

21-26

**How many hours do you play a week?**

5h+

**What type of games do you play?**

Videogames.

**Game Questions:**

**How hard was it to understand the mechanic?**

2

**Can you relate this game mechanic with the movie "Coco"?**

3

**Any additional feedback?**

More references to the film.

**Notes:**

None

## **Player 5**

### **Profile Questions:**

**What is your age?**

21-26

**How many hours do you play a week?**

2h-3h

**What type of games do you play?**

Videogames.

### **Game Questions:**

**How hard was it to understand the mechanic?**

3

**Can you relate this game mechanic with the movie "Coco"?**

2

**Any additional feedback?**

It's cute and I think the color related to emotions is interesting. However, to relate more to COCO, because in this first phase I couldn't relate it, I suggested either putting a guitar on the main character or the setting being very similar to Mexico/the setting of the film.

**Notes:**

None

**Player 6****Profile Questions:**

**What is your age?**

15-20

**How many hours do you play a week?**

1h-2h

**What type of games do you play?**

Mobile games.

**Game Questions:**

**How hard was it to understand the mechanic?**

2

**Can you relate this game mechanic with the movie "Coco"?**

2

**Any additional feedback?**

No feedback.

**Notes:**

None

**Player 7****Profile Questions:**

**What is your age?**

21-26

**How many hours do you play a week?**

0h-1h

**What type of games do you play?**

Board games (cards, hanabi, virus...)

**Game Questions:**

**How hard was it to understand the mechanic?**

2

**Can you relate this game mechanic with the movie "Coco"?**

2

**Any additional feedback?**

No feedback.

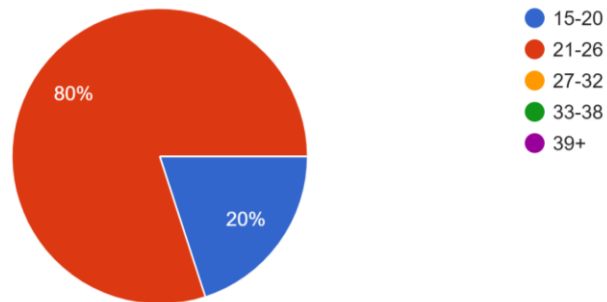
**Notes:**

None

## Graphs of the Questions:

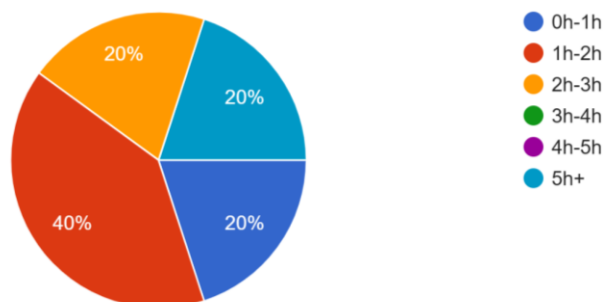
What is your age?

5 respostas



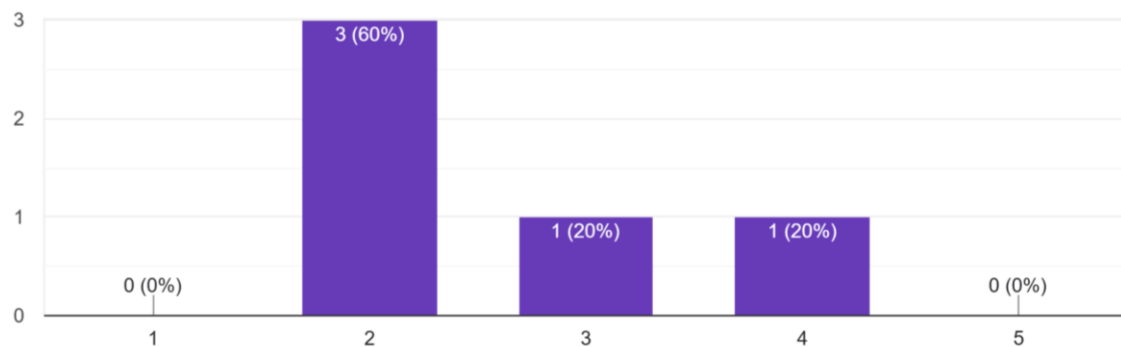
How many hours do you play per week?

5 respostas



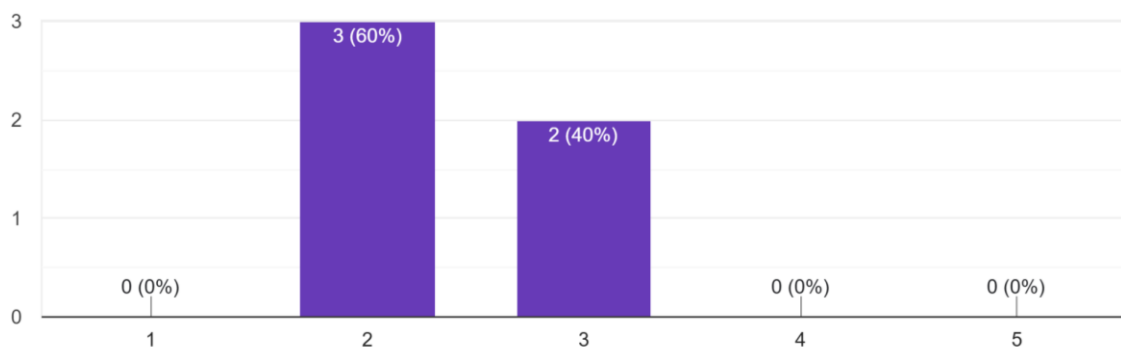
How hard it was to understand the mechanic?

5 respostas



Do you can relate thiss game mechanic with the movie "Coco"?

5 respostas





## **General overview of the 2° Playtest**

- This time we got it right with the mechanic
- We need to Adjust the puzzles because they don't offer riddles to solve, it's just a game to run and avoid obstacles

## 3° Playtest

### Player 1

#### Profile Questions:

**What is your age?**

21-26

**What type of games do you play?**

RPG/JRPG, Action, Puzzle, Board Games

**How many hours do you play per week?**

1h-2h

**It was easy to understand what the mechanic does to each obstacle and the puzzle objective?**

- **Mechanic**  
Easy
- **Objective**  
Very Easy

**Puzzle difficulty:**

- **Puzzle 1**  
Very Easy
- **Puzzle 2**  
Very Easy
- **Puzzle 3**  
Easy
- **Puzzle 4**  
Easy
- **Puzzle 5**  
Easy

**The order of the levels seems right or do you change any of them?**

Seems right

**Any other feedback?**

The placement of the red enemies does not seem relevant as they are not the ones affected by the ability.

## Player 2

### Profile Questions:

**What is your age?**

21-26

**What type of games do you play?**

RPG/JRPG, Action, Adventure, Puzzle, Simulator, Sport, Board Games

**How many hours do you play per week?**

1h-2h

**It was easy to understand what the mechanic does to each obstacle and the puzzle objective?**

- **Mechanic**  
Normal
- **Objective**  
Easy

**Puzzle difficulty:**

- **Puzzle 1**  
Easy
- **Puzzle 2**  
Easy
- **Puzzle 3**  
Easy
- **Puzzle 4**  
Easy
- **Puzzle 5**  
Normal

**The order of the levels seems right or do you change any of them?**

Ok

**Any other feedback?**

Divertido. Precisa de um timer para speedrunning :)

## Player 3

### Profile Questions:

**What is your age?**

39+

**What type of games do you play?**

RPG/JRPG, Action, Adventure, Puzzle, Sport, Board Games

**How many hours do you play per week?**

1h-2h

**It was easy to understand what the mechanic does to each obstacle and the puzzle objective?**

- **Mechanic**  
Normal
- **Objective**  
Normal

**Puzzle difficulty:**

- **Puzzle 1**  
Easy
- **Puzzle 2**  
Normal
- **Puzzle 3**  
Hard
- **Puzzle 4**  
Hard
- **Puzzle 5**  
Normal

**The order of the levels seems right or do you change any of them?**

Poderia ir aumentando a dificuldade até ao fim

**Any other feedback?**

Não

## Player 4

### Profile Questions:

**What is your age?**

15-20

**What type of games do you play?**

RPG/JRPG, Action, Adventure, Puzzle, Horror, Simulator, Sport

**How many hours do you play per week?**

5h+

**It was easy to understand what the mechanic does to each obstacle and the puzzle objective?**

- **Mechanic**

Easy

- **Objective**

Easy

**Puzzle difficulty:**

- **Puzzle 1**

Normal

- **Puzzle 2**

Hard

- **Puzzle 3**

Normal

- **Puzzle 4**

Hard

- **Puzzle 5**

Normal

**The order of the levels seems right or do you change any of them?**

Acho que trocava o 4º pelo 5º

**Any other feedback?**

None

## Player 5

### Profile Questions:

**What is your age?**

21-26

**What type of games do you play?**

RPG/JRPG, Action, Adventure, Puzzle, Simulator, Board Games

**How many hours do you play per week?**

1h-2h

**It was easy to understand what the mechanic does to each obstacle and the puzzle objective?**

- **Mechanic**

Normal

- **Objective**

Very Easy

**Puzzle difficulty:**

- **Puzzle 1**

Very Easy

- **Puzzle 2**

Normal

- **Puzzle 3**

Easy

- **Puzzle 4**

Normal

- **Puzzle 5**

Normal

**The order of the levels seems right or do you change any of them?**

No. The correct order should be 1-3-2-4-5

**Any other feedback?**

No, thanks!

## Player 6

### Profile Questions:

**What is your age?**

21-26

**What type of games do you play?**

Action, Adventure, Puzzle, Board Games

**How many hours do you play per week?**

4h-5h

**It was easy to understand what the mechanic does to each obstacle and the puzzle objective?**

- **Mechanic**  
Easy
- **Objective**  
Very Easy

**Puzzle difficulty:**

- **Puzzle 1**  
Very Easy
- **Puzzle 2**  
Easy
- **Puzzle 3**  
Very Easy
- **Puzzle 4**  
Very Easy
- **Puzzle 5**  
Very Easy

**The order of the levels seems right or do you change any of them?**

Yes

**Any other feedback?**

The position of each of the keys for the controls could be tinkered for a more normal position of the hand. It's very easy to get stuck in the points of the squares. At first, it's a bit confusing to understand how each of the npcs function, since for the blue one you need to interact directly with them but with the red ones you need to interact with their boxes.

## Player 7

### Profile Questions:

**What is your age?**

21-26

**What type of games do you play?**

RPG/JRPG, Action, Adventure

**How many hours do you play per week?**

5h+

**It was easy to understand what the mechanic does to each obstacle and the puzzle objective?**

- **Mechanic**

Very Easy

- **Objective**

Hard

**Puzzle difficulty:**

- **Puzzle 1**

Very Easy

- **Puzzle 2**

Very Easy

- **Puzzle 3**

Very Easy

- **Puzzle 4**

Easy

- **Puzzle 5**

Hard

**The order of the levels seems right or do you change any of them?**

Yes

**Any other feedback?**

Inimigos vermelhos tendo habilidade, e porta dupla com fases diferentes.



## Player 8

### Profile Questions:

**What is your age?**

21-26

**What type of games do you play?**

Puzzle, Board Games

**How many hours do you play per week?**

3h-4h

**It was easy to understand what the mechanic does to each obstacle and the puzzle objective?**

- **Mechanic**

Normal

- **Objective**

Very Easy

**Puzzle difficulty:**

- **Puzzle 1**

Hard

- **Puzzle 2**

Very Easy

- **Puzzle 3**

Normal

- **Puzzle 4**

Easy

- **Puzzle 5**

Easy

**The order of the levels seems right or do you change any of them?**

Yes

**Any other feedback?**

None

## Player 9

### Profile Questions:

**What is your age?**

21-26

**What type of games do you play?**

Adventure, Puzzle, Board Games

**How many hours do you play per week?**

2h-3h

**It was easy to understand what the mechanic does to each obstacle and the puzzle objective?**

- **Mechanic**  
Easy
- **Objective**  
Easy

**Puzzle difficulty:**

- **Puzzle 1**  
Normal
- **Puzzle 2**  
Normal
- **Puzzle 3**  
Normal
- **Puzzle 4**  
Normal
- **Puzzle 5**  
Normal

**The order of the levels seems right or do you change any of them?**

Yes. I think it's okay

**Any other feedback?**

No

## Player 10

### Profile Questions:

**What is your age?**

15-20

**What type of games do you play?**

Simulator, Sports

**How many hours do you play per week?**

4h-5h

**It was easy to understand what the mechanic does to each obstacle and the puzzle objective?**

- **Mechanic**  
Normal
- **Objective**  
Easy

**Puzzle difficulty:**

- **Puzzle 1**  
Easy
- **Puzzle 2**  
Easy
- **Puzzle 3**  
Normal
- **Puzzle 4**  
Normal
- **Puzzle 5**  
Normal

**The order of the levels seems right or do you change any of them?**

Looks alright

**Any other feedback?**

The player gets stuck on the corner of the obstacles.

## Player 11

### Profile Questions:

**What is your age?**

27-32

**What type of games do you play?**

Board Games

**How many hours do you play per week?**

0h-1h

**It was easy to understand what the mechanic does to each obstacle and the puzzle objective?**

- **Mechanic**  
Normal
- **Objective**  
Easy

**Puzzle difficulty:**

- **Puzzle 1**  
Easy
- **Puzzle 2**  
Hard
- **Puzzle 3**  
Hard
- **Puzzle 4**  
Hard
- **Puzzle 5**  
Hard

**The order of the levels seems right or do you change any of them?**

Didn't manage to go until the end

**Any other feedback?**

None

## Player 12

### Profile Questions:

**What is your age?**

21-26

**What type of games do you play?**

Sport

**How many hours do you play per week?**

5h+

**It was easy to understand what the mechanic does to each obstacle and the puzzle objective?**

- **Mechanic**  
Normal
- **Objective**  
Normal

**Puzzle difficulty:**

- **Puzzle 1**  
Easy
- **Puzzle 2**  
Easy
- **Puzzle 3**  
Normal
- **Puzzle 4**  
Hard
- **Puzzle 5**  
Easy

**The order of the levels seems right or do you change any of them?**

Puzzle 4 last

**Any other feedback?**

None

## Player 13

### Profile Questions:

**What is your age?**

15-20

**What type of games do you play?**

Action, Puzzle, Sport

**How many hours do you play per week?**

0h-1h

**It was easy to understand what the mechanic does to each obstacle and the puzzle objective?**

- **Mechanic**

Easy

- **Objective**

Very Easy

**Puzzle difficulty:**

- **Puzzle 1**

Normal

- **Puzzle 2**

Hard

- **Puzzle 3**

Easy

- **Puzzle 4**

Normal

- **Puzzle 5**

Normal

**The order of the levels seems right or do you change any of them?**

Yes

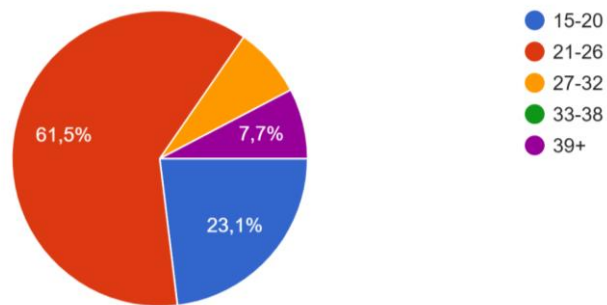
**Any other feedback?**

None

## Graphs of the Questions:

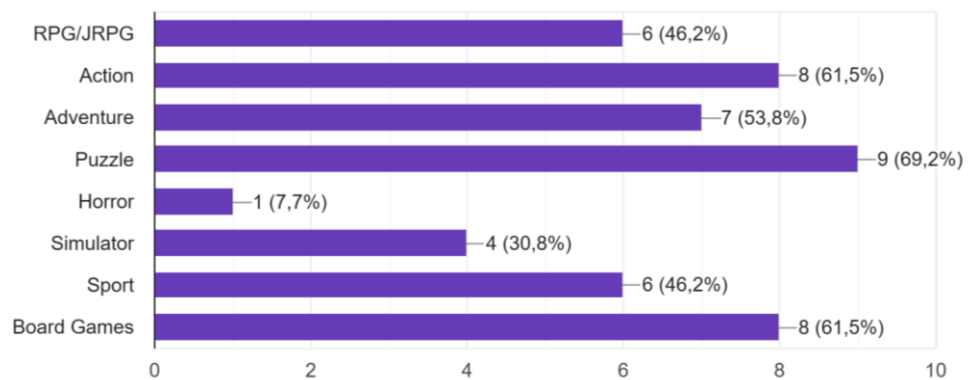
What is your age?

13 respostas



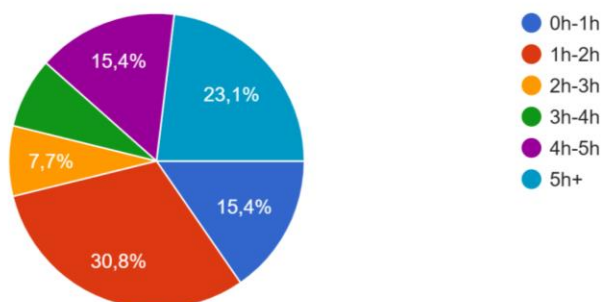
What type of games do you play?

13 respostas

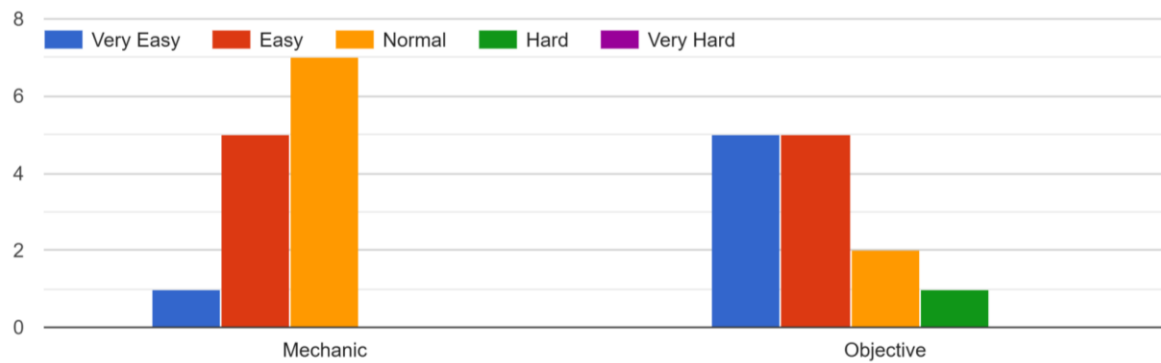


How may hours do you play per week?

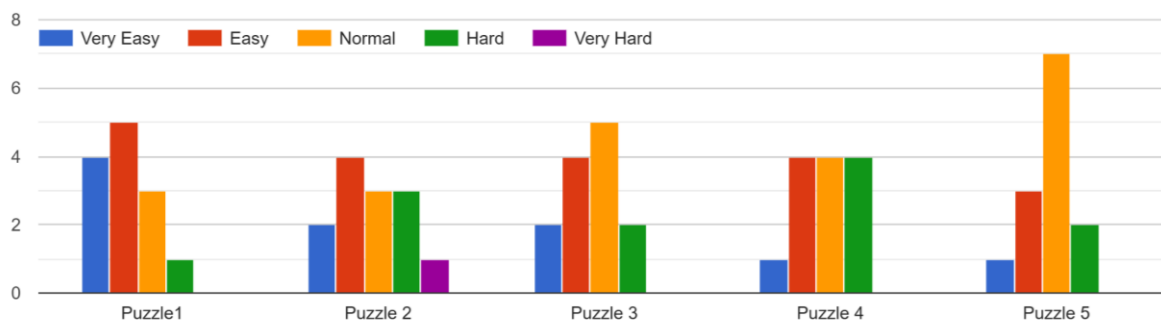
13 respostas



It was easy to understood what the mechanic do to each obstacle and the puzzle objective?



Puzzle difficulty





## Notes:

### Player 1:

- Start to trying to understand the mechanic of the blue obstacles and what they do
- Avoid the mud instinctively
- 2° puzzle, 2° jump it's a little hard
- 4° puzzle, forgot he can run so he couldn't do the first jump
- Change the jump a bit

### Player 2:

- Test if the obstacles can kill him or not
- Like the first player avoid the mud
- Get stuck between boxes doesn't kill

### Player 3:

- Confuse about the mechanic of the blue obstacle because of the animation bug
- Understand better the red obstacles mechanic
- Got a little frustrated because he lost a few times at the beginning of the 2° puzzle
- Had help in the 2° puzzle
- Complete the 4° puzzle at the first attempt
- Understand that the particles are the radius of his ability

### Player 4:

- Struggle completing the 2° puzzle
- Always running, cause of the struggling
- 2° puzzle doesn't saw that he has 2 exits
- Think that one of them was a trap

### Player 5:

- Strats to try to understand what he can do with the obstacles and if they kill him
- The red obstacles are useless without the boxes
- Stay more time in 4° puzzle because he forgot he can run
- Change the jump a bit
- Realized that he dies in the mud
- Adjust the distance between the Blue obstacles and the walls

**Player 6:**

- Difficult in understanding the mechanic of the blue obstacle because he's continuous pressing the "S" key
- Struggle with the controls
- Change the jump a bit
- Avoid the mud
- Change the radius of the ability for the player or put it in all the obstacles

**Player 7:**

- Easy understand the blue obstacles mechanic but had a little bit of struggle in the red obstacles
- 4° Puzzle understand at first that he had to run to make the jump
- 5° Puzzle choose first the way down path

**Player 8:**

- Star to try testing if he can use the blue obstacles
- took some time to understand the blue obstacles mechanic
- Got frustrated in the 2° puzzle because couldn't complete it
- Doesn't understand that the mud kill

**Player 9:**

- Easy understand the mechanics
- Easy understand the he has to run to make the jump in the 4° puzzle
- In the last puzzle he was in doubt in which door was he exit
- Understand that the mud kill

**Player 13:**

- Difficult in understand the he can use the blue obstacles as platforms
- Died in the mud once and understand that the mud kill
- Struggle with the controls
- Difficult completing the 2° puzzle
- 3° puzzle understand that he has to use the ability to pass by the box
- 4° puzzle understand that he has to use the box and run to make the jump

## General overview of the 3° Playtest

- In General the testers like the way the game was going
- We maybe have to adjust the jump of the player, because some players said that this adds a bit more challenge in the puzzles
- We have to solve the problem with the animation in the blue obstacles and the boxes that sometimes don't stop and ruin the puzzle
- If we introduce more than one exit in the same puzzle, create different puzzles for each door instead of both leading to the same place
- Although they are not the definitive controls, we have to choose the controls better, because the testers who do not usually play videogames had some difficulties using them
- Last but not the least, make a better progression between puzzles

# **Asset list**

## **Art**

### **Cemetery Day**

- Graves.
- Tombstones.
- Candles.
- Flowers.
- Humans.

### **Market**

- Bandstand.
- Humans.

### **Cemetery Night**

- Graves.
- Tombstones.
- Candles.
- Flowers.
- Dead people.

### **Characters**

- Miguel with guitar.
- Dog and spirits (if we have time).
- Hector (if we have time).

## Programmer

### CameraFollow.cs

This script is responsible for making the camera follow a target (usually the player) smoothly in a 2D Unity game. It also makes sure the camera stays within a certain area, so it doesn't show parts of the level that shouldn't be visible.

- First, we have a reference to the **target**, which is the Transform the camera will follow. Usually, this is the player's transform.
- The **Look-Ahead** section lets the camera look a bit ahead in the direction the player is moving, which makes the movement feel more dynamic. It uses the player's velocity to decide how far ahead to look.
- **Vertical Follow** just controls how smooth the camera is when following the player up and down. It's handled separately from horizontal movement for more control.
- **Camera Bounds** are the limits the camera shouldn't go past. These are set in the editor and prevent the camera from going outside the level.

In **Start()**, the script grabs the Rigidbody2D component from the target so it can read the velocity later.

In **LateUpdate()**, it first checks if the target exists, and then updates the camera position.

- It calculates the **look ahead** value depending on how fast the target is moving on the x-axis.
- The y-position of the camera smoothly follows the player's y-position.
- It then combines all that into a new position for the camera.
- After that, it calculates the visible area of the camera using the camera's orthographic size and aspect ratio.
- Finally, it clamps the new camera position so that it stays inside the defined min/max bounds, and then applies that position to the camera.

#### Summary:

Overall, this script makes the camera movement feel smooth and responsive, while also keeping the view inside the level limits.

## ButtonManager.cs

This script handles the main menu's UI buttons in a Unity game—things like starting the game, opening the controls menu, going back, or quitting. It's attached to a UI manager object in the scene and connects to the button events through the Unity Inspector.

- There's a reference to the controls menu GameObject, which is used to show or hide the controls screen. It's serialized so you can assign it easily in the editor.
- It also collects an array of Animator components from the controls menu's child objects, which it uses for button animations or transitions.

### Start()

- When the game starts, it grabs all Animator components under the controls menu and saves them.
- Then it immediately hides the controls menu so it's off by default.

### StartButton()

- Called when the player clicks "Start Game." It loads the next scene (scene 1), which usually starts gameplay.

### ControlsMenu()

- Called when the player opens the controls menu. It pauses the game by setting `Time.timeScale` to 0, resets any "Pressed" animation triggers on buttons (to avoid animation glitches), and shows the controls menu.

### BackButton()

- When the player wants to exit the controls menu, this resumes the game (`Time.timeScale = 1`) and hides the controls menu.

### QuitButton()

- If the player clicks "Quit," it logs a debug message (for testing) and exits the application. Note that `Application.Quit()` works in builds but does nothing in the editor.

### Summary:

Overall, this is a simple and clean way to handle main menu buttons, keeping different UI behaviors neatly separated and managing smooth menu transitions.

## LastLevel.cs

This script manages moving the player to the final level when they reach a certain trigger area. Attach it to a GameObject with a 2D trigger collider.

### OnTriggerEnter2D()

- When something enters the trigger, it checks if it has a **PlayerMovement** component (to confirm it's the player).
- If it is, it starts a coroutine to load the last level.

### LoadLastLevel()

- Waits a very short delay (0.05 seconds) before switching scenes. This small pause can help the transition feel smoother or allow a sound/animation to finish first.
- Then it loads scene 8, presumably the last level.

### Summary:

In short, this script cleanly detects when the player finishes the game and handles transitioning to the final level, using Unity's physics and coroutines.

## Level5.cs

This script lets you skip directly to a specific level (scene 6). It's often hooked up to a button or another trigger.

### SkipLevels()

- When called, it loads scene 6 using `SceneManager.LoadScene(6)`.  
Useful for debugging, cheat codes, or level selection.

### Summary:

It's a lightweight script with a single purpose: jump straight to a level quickly.



## Mud.cs

This script handles what happens when the player collides with mud, a hazard in the game. It restarts the level after a short delay, plays a sound, and triggers death effects.

### Variables:

- **delay**: How long to wait before restarting (adjustable in Inspector).
- **soundManager**: Reference to the SoundManager for playing sounds.
- **collided**: Tracks if the player has already hit the mud.
- **deathParticlesCheck**: A flag likely used by other scripts to trigger death particles.

### Start()

- Finds the SoundManager at the start.

### Update()

- If the player collided:
  - Counts down the delay.
  - Once it hits zero, reloads the current scene.

### OnCollisionEnter2D()

- When the mud detects a collision (probably with the player):
  - Play a sound effect.
  - Sets **deathParticlesCheck** to true to trigger death particles.
  - Marks **collided** as true.

### Summary:

In summary, this script controls death and restart behavior when the player touches mud, including sound and particle effects and a pause before resetting.

## NextLevel.cs

This script handles moving to the next level after the player reaches a trigger (usually at the level's end).

### OnTriggerEnter2D()

- When something enters the trigger, it checks if it has **PlayerMovement** to confirm it's the player.
- If yes, it starts the coroutine to load the next level.

### LoadNextLevel()

- Waits 0.05 seconds (to let animations/sounds finish), then loads the next scene by adding 1 to the current scene index.

### Summary:

This is a minimal and clean way to handle level progression, ensuring smooth transitions between levels.

## NextScene.cs

This script automatically moves to the next scene shortly after the current one starts, useful for cutscenes, splash screens, or auto-progressing levels.

### Start()

- As soon as the scene begins, it starts a coroutine called `LoadNextLevel()`.

### LoadNextLevel()

- Waits 2 seconds, then checks if the current scene index is 13:
  - If yes, it loads scene 0 (usually the Main Menu).
  - Otherwise, it loads the next scene by incrementing the build index.

### Summary:

In short, this is an automatic scene controller for timed transitions that also handles returning to the menu at the game's end.

## NextText.cs

This script controls when a piece of UI text appears after a delay, useful for tutorials or cutscenes.

### Variables:

- **delay**: How many seconds to wait before showing the text (adjustable in Inspector).
- **nextText**: The GameObject (UI element) to show after the delay.

### Start()

- Hides the nextText object initially.

### Update()

- Counts down the delay every frame.
- When the timer hits zero, activates the nextText object.

### Summary:

This script helps pace out dialogue or instructions automatically without player input.

## OpenMenu.cs

This script manages the pause menu, listening for the pause input (usually Escape), showing the menu, pausing the game, and letting the player resume or exit.

### Variables:

- **pauseMenu**: Reference to the pause menu UI GameObject.
- **animators**: Animator components under the pause menu, for resetting animations.

### Start()

- Finds all child animators under pauseMenu.
- Hides the pause menu on start.

### Update()

- Checks each frame for the Cancel input (Escape or pause button).
- When pressed:
  - Logs a debug message.
  - Pauses the game (**Time.timeScale = 0**).
  - Resets any "Pressed" animation triggers.
  - Shows the pause menu.

### Resume()

- Called when the player hits "Resume":
  - Logs a message.
  - Resumes gameplay (**Time.timeScale = 1**).
  - Hides the pause menu.

### Exit()

- Loads scene 0 (Main Menu) when the player selects exit.

### Summary:

This script provides a straightforward and responsive pause system.

## **RestartLevel.cs**

This script lets the player restart the current level by pressing a specific input button.

### **Update()**

- Every frame, checks if the “Restart” input was pressed.
- If yes:
  - Logs “Restart Level” for debugging.
  - Reloads the current scene.

### **Summary:**

It’s a simple but important feature for quick retries during gameplay or testing.

## TextTimer.cs

This script waits a set time before destroying the GameObject it's on and showing new UI (like a message).

### Variables:

- **delay**: How long to wait before triggering (adjustable in Inspector).
- **nextText**: UI element to show after the delay.

### Start()

- Hides the **nextText** object initially.

### Update()

- Counts down the delay each frame.
- When zero:
  - Clamps the timer.
  - Destroys this GameObject.
  - Activates **nextText**.

### Summary:

It's useful for temporary prompts, fading effects, or timing sequences.

## Obstacle.cs

This script doesn't do anything on its own—it acts as a marker for GameObjects considered obstacles.

- Even though it has empty `Start()` and `Update()`, its main role is to let other scripts check if an object is an obstacle by looking for this component.
- It's a lightweight, code-based way to tag objects instead of using Unity's tags or layers, which can make your code easier to organize and extend.



## CloudAnimParallax.cs

This script creates a smooth, looping parallax effect for clouds in a 2D scene. The clouds continuously scroll across the screen and seamlessly restart their animation when they move offscreen.

### Variables:

- **cloudsAnimator**: Reference to the Animator controlling the cloud animations. It can be set in the Inspector or automatically grabbed if left empty.
- **startPos**: Records the cloud's initial position to reset to later.
- **width**: The width of the cloud sprite, used to detect when it has fully scrolled offscreen.

### Start()

- If no Animator is assigned, it grabs the one attached to the same GameObject.
- Saves the initial position of the clouds.
- Measures the cloud sprite's width using the SpriteRenderer's bounds.

### Update()

- Each frame, checks if the cloud has moved beyond the left boundary (start position minus width).
- If it has fully scrolled offscreen:
  - Resets the cloud's position back to the start.
  - Restarts the animation from the beginning.

### Summary:

This script makes a seamless, infinite cloud parallax effect by repositioning and restarting animations as clouds leave the screen. It's ideal for dynamic backgrounds in 2D games needing a flowing sky.

## ParticleSystem.cs

This script handles the player's death effect by triggering a particle system and disabling the player's visuals and movement when the death condition is met.

### Variables:

- **deathCheck**: Reference to a **Mud** script that detects if the player has died.
- **playerMovement**: The player's movement component, which is disabled upon death
- **deathParticleEnable**: The particle system played when the player dies.
- **spriteRenderer**: The player's sprite renderer, hidden during death.

### Start()

- Tries to find the **Mud** script in the scene.
- Gets references to the ParticleSystem and PlayerMovement components.

### Update()

- If **deathCheck** is missing, it keeps searching for the **Mud** script.
- Once **deathCheck** is found and its **deathParticlesCheck** flag is true (meaning the player died):
  - Plays the death particle effect.
  - Hides the player sprite.
  - Disables player movement to prevent further input.

### Summary:

This script listens for a death event and reacts by playing a death effect, hiding the player, and stopping movement, making the death visually obvious and disabling control until the game resets.

## PlayerGuitarTrigger.cs

This script detects when the player presses the A, S, or D keys and tracks combinations of those keys being pressed together, updating a combo value accordingly. It's designed for rhythm or music-based mechanics where pressing keys triggers different gameplay effects.

### Variables:

- **rippleEffectPS**: A particle system intended for ripple effects (currently unused).
- **comboValue**: An integer representing the current combo or key press combination.
- **keyA**, **keyS**, **keyD**: Booleans tracking whether each key is currently pressed.

### Start()

- Checks if the ripple particle system is assigned; logs a warning if it's missing.

### Update()

- Monitors key presses/releases for A, S, and D:
  - Sets each key's boolean true on key down and false on key up.
  - Updates **comboValue** based on which keys are pressed:
    - Single keys: 1 (A), 2 (S), 3 (D)
    - Multiple keys: 4 (A+S+D), 5 (A+S), 6 (A+D), 7 (S+D)
  - Logs the current combo value for debugging.

### RippleEffect()

- Plays the ripple particle effect (not currently called).

### Notes:

- The logic has an issue: single key checks overwrite the combo before combined key checks run, which can cause incorrect combo values.
- The combined key checks should happen first or have a clearer priority order.
- The ripple effect isn't tied to combos yet but could be triggered there.

### Summary:

This script tracks key presses and combos for A, S, and D, updating a combo state usable for gameplay events, perfect for rhythm-based inputs.

## TheGuitarAbility.cs

This script lets the player use a guitar ability that affects nearby obstacles.

### Variables:

- **abilityCheckTransform** & **abilityRadius**: Define the center and range of the ability, detecting obstacles within a circle.

### Update()

- Every frame:
  - Finds all colliders inside the ability radius.
  - Checks if each has an **Obstacle** component and an Animator.
  - Keeps a dictionary of tracked obstacles and their “lifted” state (**UpMovement** bool in Animator).
  - Removes obstacles that leave the radius.
- When the player presses the ability button (**Fire1**):
  - Plays a guitar animation and a particle effect (**soundWaveArea**).
  - Toggles the “lifted” state of each tracked obstacle by setting animator booleans **UpMovement** and **DownMovement**.

### OnDrawGizmos()

- Draws a blue wire sphere in the editor to visualize the ability’s range.

### Summary:

This script manages detecting and toggling obstacles within range when the player activates a guitar ability. It’s great for rhythmic interaction with the environment.

## SoundManager.cs

This script manages audio playback, including music, sound effects, and key input sounds.

### Audio Sources:

- **musicSource**: Array of music tracks.
- **SFXSource**: Array of sound effects.
- **KeyInput**: Array of sounds for key presses.

### Indexes:

- **musicIndex**: Current music track index.
- **sfxIndex**: Current sound effect index.

### Pitch Settings:

- Controls dynamic pitch shifting for the music:
  - **pitchChangeSpeed**: How quickly pitch changes over time.
  - **minPitch** and **maxPitch**: Limits for pitch.

### Start()

- Initializes audio sources and plays the selected music track.

### Update()

- Smoothly adjusts the pitch of the playing music between min and max.
- Restarts music if it stops playing.
- Plays a key input sound when space is pressed.

### Methods:

- **SFXPlay(int sfxIndex)**: Plays a sound effect at the given index.
- **KeyInputPlay(int keyInputIndex)**: Plays a key input sound when the ability button is pressed.
- **PlayMusic(int index)**: Plays the music track at the specified index.
- **PitchChanger(int index)**: Ensures pitch stays within limits and plays the selected track.

### Summary:

The SoundManager handles background music and sound effects with dynamic pitch modulation, as well as plays sounds based on player input, maintaining smooth control over audio throughout the game.