



**Universitat Jaume I**

**Bachelor's Degree in Videogame Design and Development**

**Final Project Report**

**Design and development of an interactive adventure in  
Unity 3D**

**Author: Ángel Jara Plaza**

**Tutor: Francisco Ramos Romero**



## **Abstract**

This document presents the project memory for the planning, design, development and implementation of an interactive third-person adventure game in 3D.

The objective of the game is to find different clues that will drive the player to continue the story. These clues are spread over the setting, in order to travel along the entire scene.

The game is linear; this means that the story cannot be continued until the current riddle is not solved.

The project involves the develop of two characters, in a realistic style, in ZBrush tool, and the creation of a complete futuristic setting, where the characters are caught.

There are other pick-up objects that are not necessary to continue the game, but they are an important part of the story, since the story cannot be completely understood without them.

The player can move the characters, interact with some objects in the scene, add them to the Inventory, speak with the other character and change the character they control.

The main goal of this project is the creation of a good game experience, mixing narrative, art and programming.

## **Keywords**

Adventure, exploration, puzzle, 3D, futuristic.

# Index

1. Introduction
2. Justification
  - 2.1. VJ1227: Motores de Juegos
  - 2.2. VJ1218: Narrativa Hipermedia y Análisis de Videojuegos
  - 2.3. VJ1216: Diseño 3D
  - 2.4. VJ1222: Diseño Conceptual de Videojuegos
  - 2.5. Other related subjects
3. Tasks and Schedule
4. Tools
  - 4.1. Unity
  - 4.2. GitHub
  - 4.3. ZBrush
  - 4.4. Autodesk Maya
  - 4.5. Autodesk 3ds Max
  - 4.6. Marvelous Designer
  - 4.7. Adobe Illustrator
  - 4.8. Fungus
5. Game Design Document (GDD)
  - 5.1. Game Overview
  - 5.2. Game Flow
  - 5.3. Story
    - 5.3.1. Introduction
    - 5.3.2. Start of the adventure
    - 5.3.3. Outcome
  - 5.4. Scenes
  - 5.5. Characters
    - 5.5.1. Julie
    - 5.5.2. Brown
    - 5.5.3. Captain Stone
  - 5.6. Game world
6. Project development
  - 6.1. Player Mechanics
  - 6.2. Camera Mechanics
  - 6.3. Object Properties
  - 6.4. Game Managers
  - 6.5. Fungus
7. Art
  - 7.1. 2D Art

- 7.1.1. In-game interface
- 7.1.2. Menu design
- 7.1.3. Renders
- 7.1.4. HUD
- 7.2. 3D Art
  - 7.2.1. Characters
    - 7.2.1.1. Brown’s model
    - 7.2.1.2. Julie’s model
    - 7.2.1.3. Animation
- 7.3. Environment
- 7.4. Props and objects
- 8. Results
- 9. Conclusions
  - 9.1. Achieved Goals
  - 9.2. Real Workflow
  - 9.3. Personal conclusions
- 10. Bibliography
- 11. Annexes

- Table 1: Related competences in VJ1227..... 9
- Table 2: Learning outcome of VJ1227 ..... 9
- Table 3: Related competences in VJ1218..... 9
- Table 4: Learning outcome in VJ1218 ..... 10
- Table 5: Related competences in VJ1216..... 10
- Table 6: Learning outcome in VJ1216 ..... 10
- Table 7: Related competences in VJ1222..... 10
- Table 8: Learning outcome in VJ1222 ..... 11
- Table 9: Initial schedule..... 13
- Table 10: Final schedule..... 60

Figure 1: Tools used in this project .....	16
Figure 2: Scene scheme.....	21
Figure 3: First sketches of the building's distribution (2D and 3D) .....	23
Figure 4: Building space distribution.....	23
Figure 5: Character Scripts .....	25
Figure 6: Manager Scripts .....	26
Figure 7: Fungus flowcharts: Simple (top) and compound (bottom).....	27
Figure 8: Inventory (Left) and Options (Right) menu buttons .....	28
Figure 9: Locked (Left) and unlocked (Right) pick/interact buttons. ....	29
Figure 10: Locked (Left) and unlocked (Right) speak buttons.....	29
Figure 11: Change character button. ....	29
Figure 12: Empty Item Menu (top) and Item Menu when an object is picked (bottom) .....	30
Figure 13: Options Menu without options selected (left) and Options Menu with Sound option selected (right). ....	31
Figure 14: Brown's PC in the scene. ....	31
Figure 15: Brown's PC render in 3ds Max. ....	32
Figure 16: Interface with hidden menu.....	32
Figure 17: Interface with Items menu and Options menu opened.....	33
Figure 18: Main characters final face.....	33
Figure 19: First changes in the model .....	34
Figure 20: Face after and before the geometry subdivision .....	35
Figure 21: shirt model and subdivision level up.....	36
Figure 22: First hair approach and clothes model.....	37
Figure 23: Face changes to improve realism.....	37
Figure 24: Face with base colour, detailed colour and hair and final face .....	38
Figure 25: Painted model .....	39
Figure 26: Julie's model first steps .....	40
Figure 27: Adding detail, hair and clothes .....	41
Figure 28: Hair model, base colour and detailed colour .....	42
Figure 29: Julie's face with base colour, adding details and final face.....	43
Figure 30: First steps in Mixamo .....	44
Figure 31: Animation selection and modification .....	45
Figure 32: Male (top) and female (bottom) base animations.....	46
Figure 33: Animations selected and animation controller in Unity .....	47
Figure 34: Controlling animations with scripts in Unity.....	47
Figure 35: Animated character in the game.....	48
Figure 36: First model of the building .....	48
Figure 37: Adding details.....	49
Figure 38: Roof distribution (compound roof in dark colour).....	50
Figure 39: Detailed environment model with simple textures .....	50
Figure 40: Smart materials in Substance Share.....	51
Figure 41; Spaceship and intercom models. Directly imported to Unity.....	52
Figure 42: Bed and door models. Modified and imported. ....	52
Figure 43: Table modeled and imported to Unity. Added materials to the metallic part in the engine.....	53

Figure 44: In-game capture 1 .....	54
Figure 45: In-game capture 2 .....	54
Figure 46: In-game capture 3 .....	55
Figure 47: In-game capture 4 .....	55
Figure 48: In-game capture 5 .....	56
Figure 49: In-game capture 6 .....	56
Figure 50: In-game capture 7 .....	57
Figure 51: In-game capture 8 .....	57
Figure 52: In-game capture 9 .....	58

# 1. Introduction

This project has been created with three objectives: The first one is the creation of an interesting story, with good characters and ambient narrative. The second objective is the creation of a realistic scenario, with detailed characters and objects. The last one is the creation of a scene set in Unity3D.

The main idea is important to the story, but some stories with a great premise fail because of the develop.

Can seem difficult that a game with a good story and good art fail, but if the game has not good mechanics, or good implementation, it can be a disaster. So, what part of the game is the most important? There is not a correct question for that, but the developer should assess what part will be the reference on their game. If the project is a fighting game, the developer usually choses mechanics over story; if the game is an interactive adventure, the creators focus on the story and art as a rule.

The game story can be told in many ways. Some games develop the story of a traditional way, as it can be told in books or films. This is not a bad way of telling a story, but some capacities of videogames are wasted. Other games accomplish a more complete story, adding narrative effects to the ambient, objects and places, creating a game Lore. To develop a complete game Lore is very difficult, since the game designer must be careful about the objects they put in the scene, what are they trying to tell and how they magnify the story.

To achieve the main objectives, it is necessary to subdivide them in more simple tasks:

- Designing a game map with different zones where the characters inhabit.
- Modelling two realistic characters in ZBrush.
- Creating assets and putting them into the scene, in order to enhance the story.
- Creating a visual and interactive story, that can be completely understood only if the player pay attention to the scenario and items.
- Creating an Artificial Intelligence that follows the character controlled by the player and allows them to dialogue with it.
- Animating and implementing the couple of characters, with walking, running, and idle animations.
- Create a good camera movement that allows the player to pay attention to the scene details.



## 2. Justification

In videogames, a lot of different professionals are required to develop a single activity. This project has been proposed as a summary of all these professions.

This section will focus the relationship between this project and the degree of design and game development.

### 2.1. VJ1227: Motores de Juegos

This is probably the most important subject for the development of the project. In this subject is made the first 3D videogame in the degree, and it must be done in Unity 3D engine.

<b>Related competences:</b>
E12: "Capacidad para evaluar, usar y extender motores de juegos".

*Table 1: Related competences in VJ1227*

<b>Learning outcome:</b>
E12, G09: "Evaluar las características técnicas de los motores de juegos como tecnología para la creación de videojuegos".
E12, IR07, IR14: "Usar motores de juegos para la creación de videojuegos".

*Table 2: Learning outcome of VJ1227*

### 2.2. VJ1218: Narrativa Hipermedia y Análisis de Videojuegos

If VJ1227 is the main subject for the development of the videogame, VJ1218 is the main subject for the design of this project. "Narrativa Hipermedia y Análisis de Videojuegos" is essential to understand how the game design should be done.

<b>Related competences:</b>
E11: "Capacidad para generar y analizar recursos expresivos y narrativos aplicados a discursos audiovisuales, con especial énfasis en el multimedia, hipermedia y videojuegos".

*Table 3: Related competences in VJ1218*

<b>Learning outcome:</b>
E11: "Generar guiones audiovisuales".
E11, G01: "Utilizar las nuevas tecnologías audiovisuales emergentes para la elaboración de discursos audiovisuales en el terreno de la publicidad, del diseño gráfico, la infografía y los productos multimedia".

*Table 4: Learning outcome in VJ1218*

### 2.3. VJ1216: Diseño 3D

This subject is the first contact with a 3D design tool (3ds Max). The first 3D model in the degree is made in this subject too. Is very important to learn how to design the model in a proper way, and to understand all the process, step by step.

<b>Related competences:</b>
E04: "Capacidad para el diseño y la creación de elementos gráficos".
IR07: "Conocimiento, diseño y utilización de forma eficiente los tipos y estructuras de datos más adecuados a la resolución de un problema".
IR13: "Conocimiento y aplicación de las herramientas necesarias para el almacenamiento, procesamiento y acceso a los Sistemas de Información, incluidos los basados en web".

*Table 5: Related competences in VJ1216*

<b>Learning outcome:</b>
E04: "Analizar las características técnicas de las herramientas de diseño 3D".
E04: "Usar aplicaciones de modelado tridimensional para videojuegos".

*Table 6: Learning outcome in VJ1216*

### 2.4. VJ1222: Diseño Conceptual de Videojuegos

The ability of being singular is very hard to find, but if it is reached, is an incredible identity brand. In this subject are taught some abilities that can be used for this purpose.

<b>Related competences:</b>
E15: "Capacidad para crear y analizar juegos en sus elementos fundamentales y desarrollar la comprensión de cuáles son las claves que determinan su funcionamiento y desarrollo".
G05: "Capacidad de gestión de la información".

*Table 7: Related competences in VJ1222*

<b>Learning outcome:</b>
E15: "Definir reglas que armonicen con las posibilidades tecnológicas y aporten fluidez, así como dinámicas del juego como elementos protagonistas en la construcción del diseño".
E15, G05: "Conocer y controlar todos los elementos y objetos que participan en la configuración de un juego y la capacidad de establecer un equilibrio armónico entre los mismos".
E15, G10: "Diseñar escenarios y entornos equilibrados para el desarrollo del juego".
E15, G10: "Diseñar sistemas de equilibrio entre las mecánicas de juego, los objetivos a cumplir y los posibles conflictos dentro y fuera del juego".
E15, G10, G05: "Desarrollar de forma precisa, ordenada y clara toda la información relativa al diseño de un proyecto de videojuego".
G04: "Redactar un texto en inglés técnico sobre videojuegos".

*Table 8: Learning outcome in VJ1222*

## 2.5. Other related subjects

- VJ1203: Programación I: First contact with a program language. Basic knowledge of programming.
- VJ1208: Programación II: C# learning.
- VJ1223: Arte del Videojuego: HUD design approach.
- VJ1236: Técnicas de Producción y Realización Sonora: How to create sound atmospheres.

## 3. Tasks and schedule

### 3.1. Tasks

- Task 1: Story design. This part should be the first one, because all designs depend of what kind of story is pretended to talk. In this task is included the game design.
- Task 2: Designing and modelling the building. First, it is necessary to make a 2D design. When this part is completed, is the moment to create the 3D model. This task is made in Adobe Illustrator (2D) and Autodesk Maya (3D).
- Task 3: Brown and Julie model creation. When the character personality is designed, is easier to know how they should be. It is important to give them some distinctive features (in the design and in the model phases), in order to improve the reality and the player's empathy. All the character models are made in Pixologic ZBrush.
- Task 4: Animating the characters. Animation is essential in a 3D model. The model is the clay figure, but the animation is when life is given to it. All character animations are made in Adobe Mixamo.
- Task 5: Finding or model the appropriate objects to add them into the game. Now, characters and scenario are completed, but the third part of the project design is not. It is necessary to add the objects to the scene. Some of them have been modelled in Autodesk 3ds Max and Autodesk Maya, and others have been downloaded from free 3D model webpages, like "Turbosquid", "CGTrader" and "Free3D".
- Task 6: HUD creation. When the player picks an object, it disappears from the scene, but it should be transferred to the Inventory. This inventory is represented in the HUD, showing an object image and a brief description of it. The HUD is made in Adobe Illustrator.
- Task 7: Unity environment implementation. When designs and models are completed, it is time to add them to the Unity project. These models need behaviour patterns, that stipulates how they act in different circumstances.
- Task 8: Unity characters and animation implementation. Adding characters and their animations to the Unity project can be harder than adding simple objects. There are more properties to add and more possible errors to fix. Even the animations are made, it is necessary to create an animator controller to modify the behaviour of them. Furthermore, other scripts are necessary to control the character actions (moving, picking objects...).
- Task 9: Adding dialogues to the project. When the characters talk to each other, a dialogue box appears in the screen. Other text box appears when the character interacts with an object. Sometimes, another text box is shown when the character is

thinking or figuring something. All this narrative has been programmed to appear in a definite moment. The tool used to develop this task is Fungus, a free tool available in Unity Asset Store.

- Task 10: Scene changing. The game is compound by eight main scenes. A scene change is due to the player has managed to advance the story. Only when the player takes the right decisions, the scene will change.
- Task 11: Debugging and playtesting. Even the game seems finished, it is very possible that some bugs remain hidden, and the developer cannot fix them in they do not realize of their existence. For this reason, debugging and testing is so important. It is even better if different people play the game, since there are more possibilities to find those bugs.
- Task 12: Memory. The game is finished, but there is another important part of the project development. This memory document details all the creation process, since the first idea to the final results.

### 3.2. Schedule

	FEB	MAR	APR	MAY	JUN	JUL	OVERALL
TASK 1	25	5					30
TASK 2	5	20					25
TASK 3	5	25	10				40
TASK 4			10				10
TASK 5	5	15	5				25
TASK 6				10			10
TASK 7			20	10			30
TASK 8			15	25			40
TASK 9				10	10		20
TASK 10					10		10
TASK 11			5	10	5		20
TASK 12					40		40
OVERALL	40	65	65	65	65	0	300

*Table 9: Initial schedule*

## 4. Tools

The tools used during this project are going to be named and described in this section.

### 4.1. Unity

Unity is the main tool for the development of the project. This is the game engine that will be used to implement the art, script and code created in other tools, in its version 5.6.

Unity is a great option to create any 2D or 3D game. This game engine has been considered the best option due to the experience gained in other subjects in this degree, more specifically, “VJ1227: Videogame engines”, as a first contact experience, and “VJ1239: Basis for the Design of Didactical Videogames”.

For writing the code the tool used is MonoDevelop, the default coding program in Unity.

### 4.2. Github

GitHub is a web-based Git or version control repository and Internet hosting service mostly used for code. This tool will be used to control, save in the cloud, and revert, if necessary, the changes in the project.

### 4.3. ZBrush

ZBrush is a digital sculpting tool that combines 3D/2.5D modelling, texturing and painting. This program is used for creating high-resolution models for use in movies, games and animations.

In this project, ZBrush will be used for modelling and painting the main characters.

This tool has many features. Some of the used in this project are:

- 3D brushes: Each brush offers unique attributes as well as allowing general control over hardness, intensity and size.
- Polypaint: This feature allows the user to paint on an object’s surface without the need to first assign a texture map by adding colour directly to the polygons.
- GoZ: Allows the user to transfer in a simple way the model to Maya, 3ds Max, and other platforms. Its plugin can be installed in these tools to make the connexion between programs too much simple.

#### **4.4. Autodesk Maya**

Maya is an animation and modelling program used to create three-dimensional full-motion effects. This tool will be used to model the building. Also, Maya is going to be used to add materials and textures to other objects in the scene.

#### **4.5. Autodesk 3ds Max**

3ds Max is a professional 3D computer graphics program for making 3D animations, models, games and images. In this project, this tool will be used, along with Maya, to design, model and edit the objects that take part in the game. This is the software used in the subject "VJ1216: 3D Design".

#### **4.6. Marvelous Designer**

Marvelous Designer allows creating 3D virtual clothing with its own cutting-edge design software. In this project, this tool will be used to design and model clothes for the characters.

#### **4.7. Adobe Illustrator**

Adobe Illustrator is a computer program that allows users to compose and edit vector graphics images interactively on a computer. The HUD will be designed and created with this software.

#### **4.8. Fungus**

This tool can be found on Unity Asset Store. It is a free tool that allows the user to add dialogues, conversations and flowcharts in a very intuitive way.



Figure 1: Tools used in this project



## 5. Game Design Document (GDD)

### 5.1. Game Overview

Between the Lines is an exploration adventure game, with simple puzzles that the player must solve to continue the story.

Two people are in a building, they are trapped but they do not know that. They don't remember being outside, so this is the only place they know.

Everything changes when Julie, the art manager, finds a strain message written in a book. The three friends start asking questions about who are they, and why cannot they leave this building.

The player can explore the map, but if they want to advance, they must solve the puzzles. Some puzzles open new zones, blocked at the beginning of the adventure.

The scenario has a bedroom zone in the east wing, an office zone in the west, a living room and dining room in the centre and a library-museum in the north.

This game is targeted to people who like adventure and mystery games.

### 5.2. Game Flow

The player can interact with the character anytime, with different commands:

- Walk: The characters can walk in the scene, but they cannot do that if they are talking or reading.
- Run: If the player is in a hurry they can run instead of walk. This control makes the character go faster.
- Pick object: If the character is next to an object, the pick button is activated, and the player can pick this object.
- Speak: When both characters are near enough, they can speak to each other. The dialogues will be different depending who starts the conversation.
- Change character: The player will play with both characters. There are scenes in which only one character is involved. Other scenes involve both characters. In the second case, the player can change the player who is following them.

- Stop follow/start follow: If the player wants to explore without the other character, they can press the stop follow button, and the non-playable character will stay in the same place until the player presses the same button (now it will be the start follow button). In this moment, the non-playable character will start following the player again.
- Interact: There are some places in the scenario where the interact button can be pressed. If the player presses that button, some actions should be unlocked.
- Open inventory menu: The inventory menu can be opened anytime, and it will be updated when the player picks an object. This menu includes an object's image and a brief description of it.
- Open options menu: The options menu is the menu that allows to continue the game, enable or disable sound and exit the game.

The game is divided in scenes. There is not a main scene, but some scenes are larger than other.

### **5.3. Story**

#### **5.3.1. Introduction**

In the first scene, we saw Brown's daily awakening, and his road to work. When Brown is set to work, the first scene finishes. The second scene has the same outline but this time the player can control Julie. The change character mechanic is not activated in these levels.

#### **5.3.2. Start of the adventure**

Third scene starts with Julie reading a book, when she realizes someone had written that book. Julie reads this message, and she feels a new sensation, fear. The person who wrote that was trying to make them notice that they are caught, and they will be forever in this building if they do not do something. Julie calls Brown and makes him read the note. The last words in this message say: "Take me always with you, I can open doors you cannot".

The player is free now to move Julie or Brown in the scenario, but if they want to advance with the story, they must go next to the locked room with a controller. In the first two scenes, this controller says: "This door is closed". In this scene, if the player takes the book with the note, the door is unlocked and the controller has now two options: 1: "Enter the room". 2: "Keep walking". If the player picks the first option, they will pass to the next scene.

The door that the player opened in the third scene leads to an office warehouse, with a lot of papers, documents and other stuff. The fourth scene will be completed if the player finds an old document: a personal pass for a bedroom. The warehouse door will be locked until the player finds this pass, which will be hidden between a lot of useless objects, with which the player can interact, but they cannot pick them.

The fifth scene starts when the player exits the warehouse. In this moment, a door in the bedroom zone has been unlocked, due to the pass is from a person who lived there. In that moment, they realize that the possibility of going out of the building is possible, since they had never seen that person. When the player enters the room, they can see new objects, like a huge spaceship model, or a simple small robot that can speak about his creator and what he knows about him. If the player asks the right questions, the robot will answer them how to continue. The robot explains that his creator wanted to make a girl-robot for a little girl, who he appreciated a lot. She was the alpha model, but then, when his creator was designing a better Artificial Intelligence, he disappeared. The robot says that there was an old boss, who was the only one who could assign works and tasks. She knows about the boss because his creator put an AI chip on her that belonged to that boss. The boss was the best in artificial intelligence, and she was trying to create cyborgs, but she failed all her attempts. The last thing the robot says is that they need to take her chip if they want to know more about the boss, because this is the only way to open her office, and in that place, they will find answers.

When the player exits the bedroom with the robot's chip, the fifth scene ends, and the sixth scene automatically starts.

### **5.3.3. Outcome**

The sixth scene starts in the living room, with Brown and Julie talking about the people who lived there before them. If that people could get out of this building, maybe they can too.

The boss office can be opened now. When the player enters this room, the door will close, and a strange voice will start speaking:

VOICE - A... Amy? Is that you?

JULIE – Who is talking?

VOICE – What? I was here before you. What is your name?

JULIE – My name is Julie, and he is Brown. Who are you?

VOICE – Julie... That's your name now?

BROWN – Who are you?!

VOICE – Take it easy, I am Captain Stone, commander of this ship.

JULIE – Ship?

VOICE – Yes, spaceship, don't you know that?

BROWN – How are we going to get out of here if we are in the space? This can't be true!

VOICE – Why do you think you are still working here? I should kick you out the space after all things you did to me, but I need you. And you need me.

BROWN – What do you mean? I don't know you.

VOICE – You confined me in this old system! You are the only one who can do that.

JULIE – And what about your old pupil?

VOICE – Are you out of your mind? Or you simply don't remember anything? Oh, yes, sometimes I don't remember you don't remember.

BROWN – She is perfectly sane, and you are lying.

VOICE – Wow, you still love her. But I don't think it's the same kind of love.

JULIE – What are you talking about?

BROWN – I... Don't...

VOICE – She was like your little sister, but now you are in love with her. Maybe I can't delete your feelings after all.

JULIE – Are you saying that you deleted all our memories? How did you do that?

VOICE – Do you think I am going to answer that?

BROWN – Answer her question! Or I...

VOICE – Or what? You should not hurt me, if you want to stay alive. Anyways, you could do nothing. I am not in this computer, I am the system right now, I can see you anytime, and believe me, I do.

JULIE – Have you been spying us all this time?

VOICE – Spying? No, I was worried about my daughter and her friend!

JULIE – What?!

VOICE – Why do you think you are alive? There were people more capable than you, but I still have feelings, and I still want you to be alive. Now we need each other to survive, at least until we take a new planet.

BROWN – What happened to the other people? Why did they die?

VOICE – They were useless. When we lifted off, a lot of people was necessary to continue this travel, but I knew with that amount of people this travel would be impossible.

JULIE – And you killed random people with no reason?

VOICE – No! I told you, they were useless. They finished their work, and it was impossible to make them to do other tasks, so they must die.

JULIE – And you tell us all this story, you don't want to kill us, and you expect that we are still going to help you to reach other planet. Are you insane?

VOICE – I told you because you have no choice, I can delete your memories, but I think you want to keep them. Moreover, if we don't reach the planet that Brown calculate, food and water will run down.

BROWN – That calculations were for this ship?

JULIE – Brown, is she telling the truth?

BROWN – Unfortunately... yes.

JULIE – And it is not possible that you miscalculate the time we have?

VOICE – He never miscalculates.

JULIE – Well, we have no choice if we want to survive, but why we want to survive if we are caged?

VOICE – I didn't want to tell you that but, it is very possible that you were the last couple of human been in the universe.

BROWN – But, Earth...

VOICE – We left Planet Earth to survive, not because we wanted to. You think I'm a monster, but I did only what I had to do.

BROWN – We will reach the planet in a few months, what will happen then?

VOICE – Probably I will be useless then.

#### 5.4. Scenes

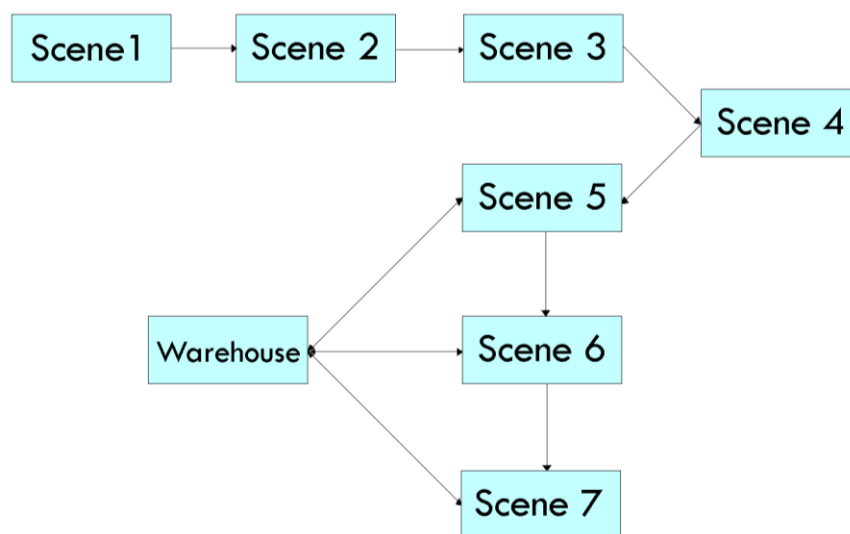


Figure 2: Scene scheme

## **5.5. Characters**

### **5.5.1. Julie**

Age: 24-28

Gender: Female

Occupation: Librarian, book-machine interpreter.

Role: Protagonist.

Background: Julie is an active woman. She loves to read and discover, that is why she gets up every morning with a desire to go working. She does not know nothing about her past, and she never asks questions about it, since she reads the book note. Then she realizes that she does not remember when she started to work reading and interpreting books to add them in the system.

### **5.5.2. Brown**

Age: 30-36

Gender: Male

Occupation: Space orbit and engine engineer.

Role: Protagonist.

Background: Brown has been doing a long time an activity that he hates. He does not really know why he do this work, but he is sure he must do that. When Julie finds the note, Brown is determined to search and find the answers to their questions.

### **5.5.3. Captain Stone**

Age: Undetermined

Gender: Female

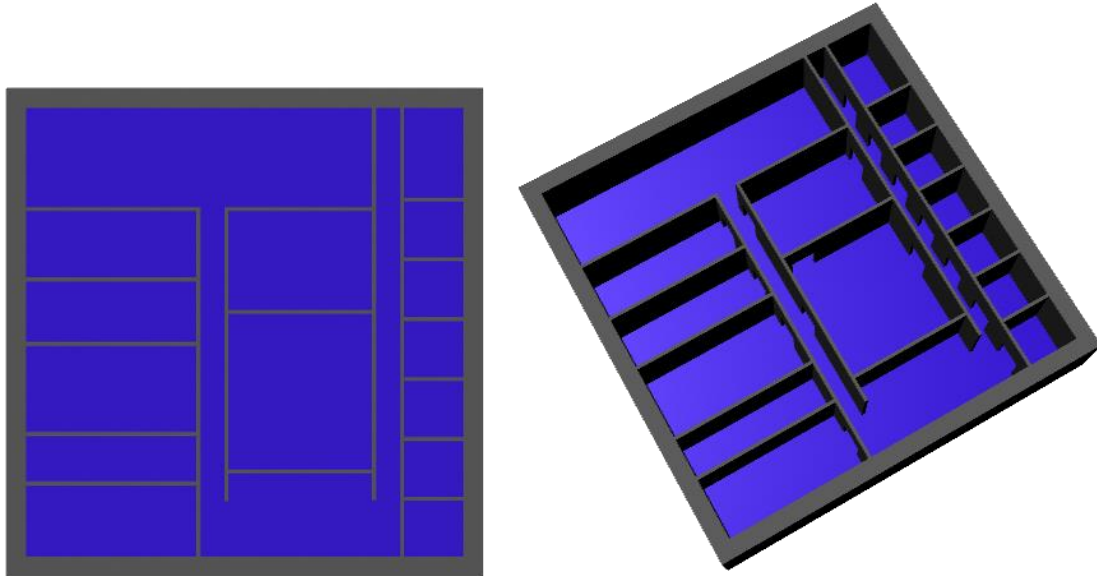
Occupation: Captain of the spaceship.

Role: Secondary.

Background: It is not clearly known when the spaceship lifted off, but is sure than the captain was still commanding it. Captain Stone is a Machiavellian person, who is always looking for the success of her main assignment: preserve the human life.

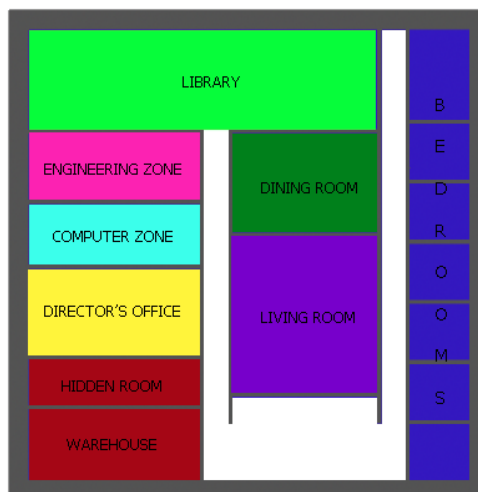
## 5.6. Game World

The game take place in a closed building, and the player will unlock new areas when they advance the story.



*Figure 3: First sketches of the building's distribution (2D and 3D)*

After the sketch, it is necessary to divide the building in areas, and these areas should be divided in different rooms.



*Figure 4: Building space distribution*

## 6. Project Development

In this section will be exposed all the implementation and mechanics of the project. This display is divided in player mechanics, where all scripts used for the character are explained; camera mechanics, where its mechanics are exposed; the object properties, section in which are defined all the objects possibilities and Game Managers.

To make all dialogues and emergent texts, is used the free tool Fungus, available on Unity Asset Store. With this tool, the character can add dialogue flowcharts, and modify them with different options. It is possible to make selection menus, that change the dialogue depending the option selected. Furthermore, this tool can instantiate, enable and disable objects after the dialogue, or if one option is selected.

### 6.1. Player Mechanics

The player can choose the character they control. In some scenes, there is only one character to play. Obviously, in that scenes, the character change is disabled. In scenes in which both characters are, the player can press the keyboard button “E”, and the controlled character will change to the other one.

The player can make the character move, pressing the moving buttons (“W” to walk, “A” and “D” to rotate, and “S” to turn). The character animations change when the character is moving. The player can make the character run too, pressing “Space” button.

When both characters are near enough, the speak option is enabled. Then the player can make them speak, pressing the “R” button. Depending of the character that the player is controlling, the dialogues will be different. The dialogues change depending the scene too.

In scenes with both characters, the NPC follows the character controlled by the player. Depending of the distance, the character will run, walk or stay calmed.

### 6.2. Camera mechanics

In third person games, the camera is very important, due to it should follow the character and look at them, and its movement must be smooth. There are components that the developer can change, like the distance between the camera and the character and its elevation.

In the warehouse scenes, the main camera is in a perspective mode, its position does not change when the player moves, it simply looks always the character.



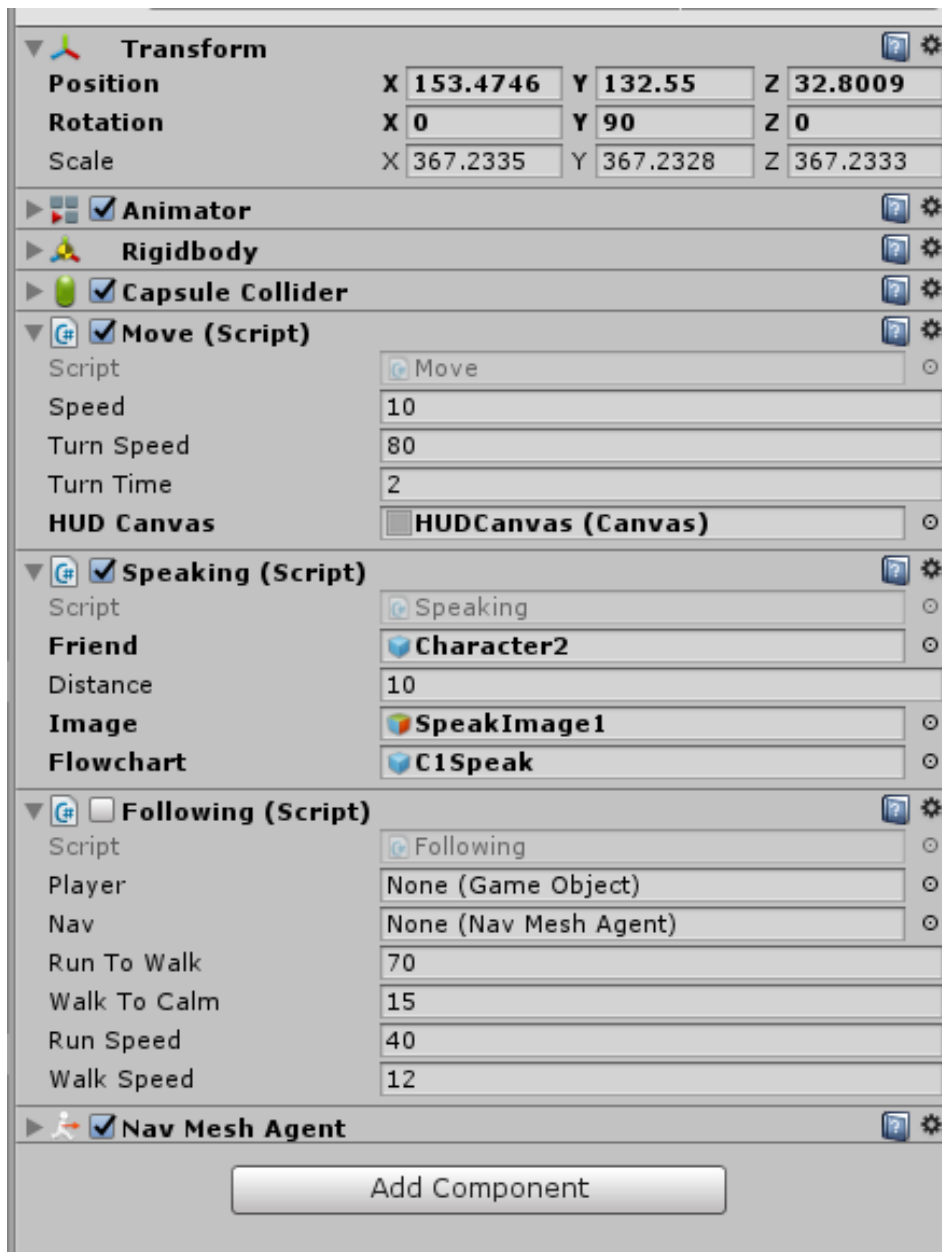


Figure 5: Character Scripts

### 6.3. Object properties

Pickable objects have two phases, the first one, when they are in the scene. At this point, the object is a 3D model, with the possibility of being picked if the character is near enough and the player presses the “F” button. If the player does that, the object disappears from the scene as a 3D object and appears in the Inventory (Item menu). In some cases, this object unlocks some environment features (no need to use it, simply carrying it that features are unlocked). Anyways, the object added to the Inventory can be seen in the Item Menu.

## 6.4. Game Managers

Some options are not attached to a simple model. There are some functions that controls more than one object behaviour.

When the player presses the “E” button, the character they are controlling changes, but these changes affect to other GameObjects, like the camera, that always follows the character controlled by the player. Objects cannot react to the NPC, so their behaviour must be changed too according the actual player character. This function is made with the idea of adding more characters to it simply dragging their GameObjects to the “Characters” list.

When the scene changes, the old scene is destroyed, and the new one is loaded. This can be a problem in this game, since the inventory must not be deleted. All items picked must stay in the same position, no matter what scene is loaded. This requirement is solved adding GameObjects to the DontDestroyOnLoad() list. This function keeps the selected GameObjects when a new scene is loaded.

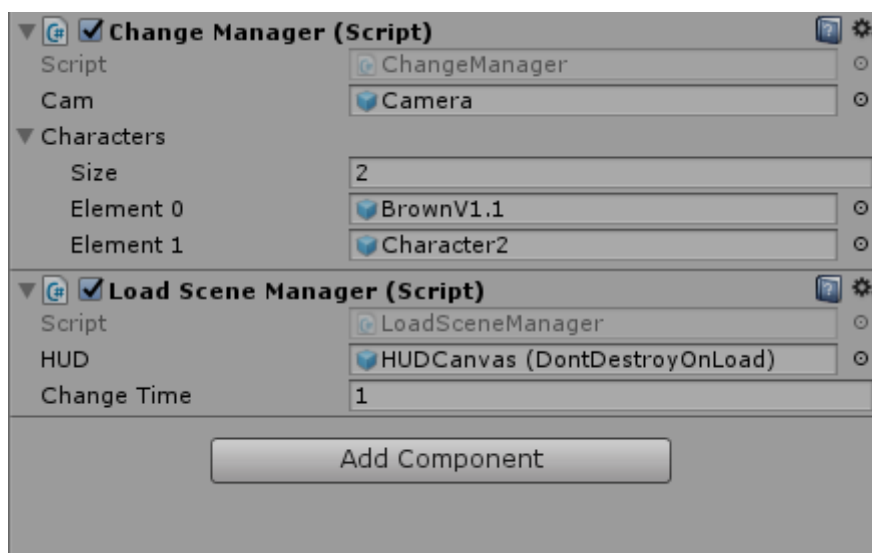


Figure 6: Manager Scripts

## 6.5. Fungus

Fungus is a tool that allows the user to create dialogues and conversation trees, using flowcharts and option menus. To use this tool, it is necessary to download it in the Unity Asset Store. When it is imported to the project, in “Tools” menu is added a Fungus option with some creation alternatives.

In this project is used the flowchart option, since its easiness and versatility.

Once the flowchart is added to the workspace, for modifying its behaviour the Flowchart window must be opened.

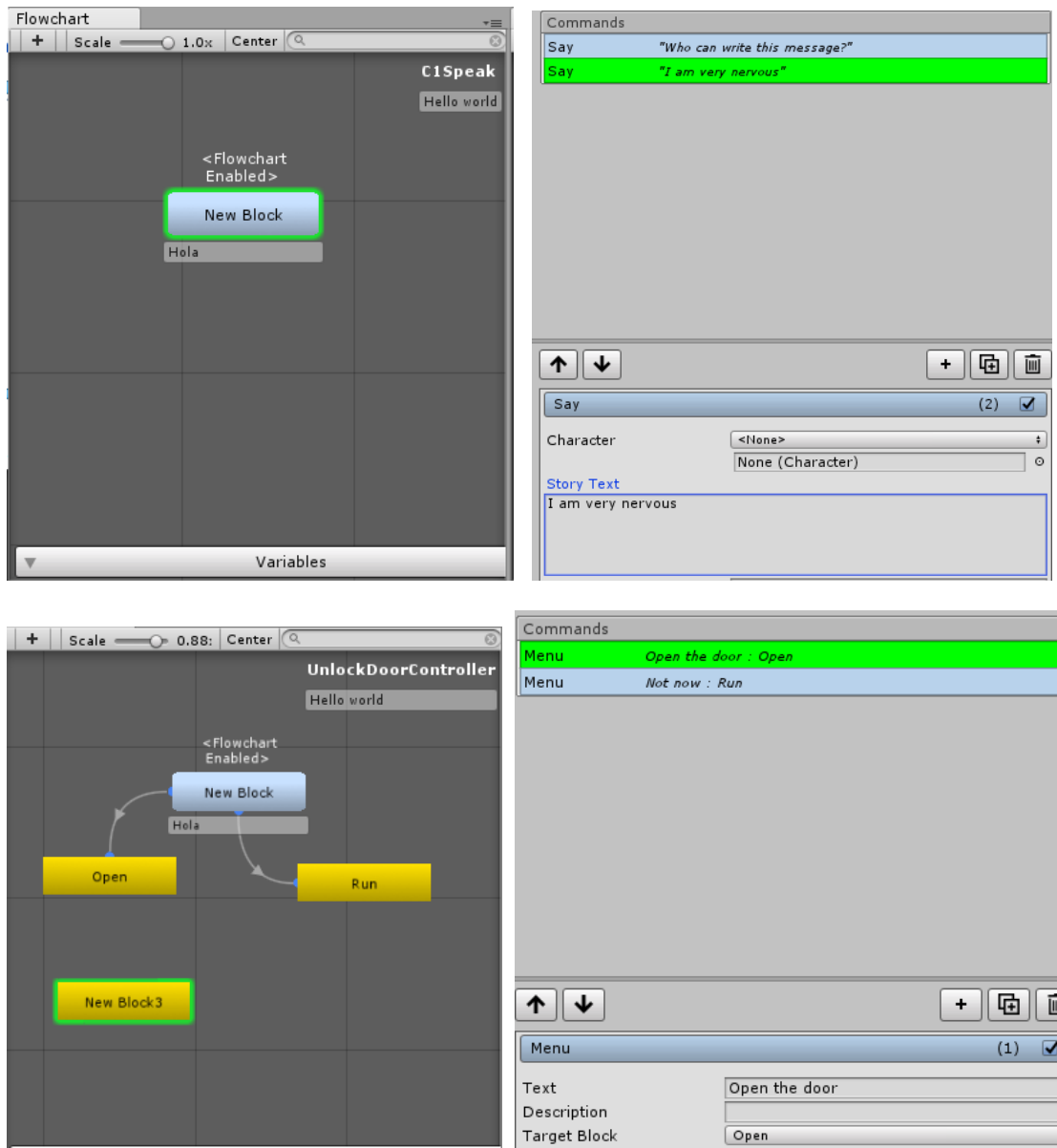


Figure 7: Fungus flowcharts: Simple (top) and compound (bottom)

## 7. Art

Graphics are an essential part of videogames. It is not necessary to be a realistic design, but the graphics body must be consequent. This means that the final outcome should make sense, and it should not be a mix of different styles, unrelated to the story or the design.

In this section, the design and creation of the art (2D and 3D) in the game is shown.

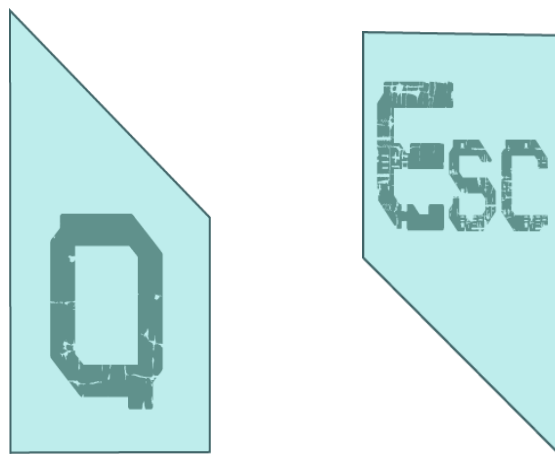
### 7.1. 2D Art

In this project, 2D is divided in three main categories: Menu design, in-game interface design and image renders. These parts work as one to create the HUD.

#### 7.1.1. In-game Interface

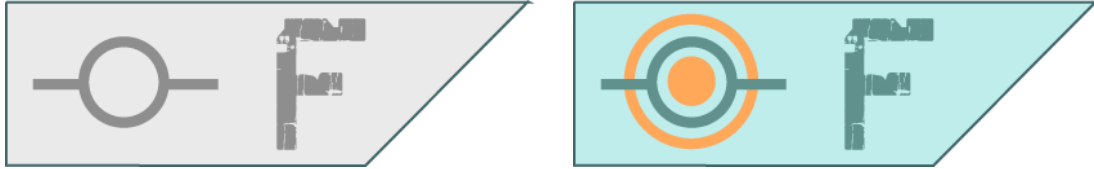
The player has multiple options and mechanics every moment, and they need to know what can they do. For example, when the character is near an interactable object, the player can press the action button “F” and the character will interact with that object. There are five buttons to press:

- Open/close inventory menu: To open this menu, the player must press the “Q” button. This command can be executed anytime.
- Open/close options menu: This menu will be opened if the player presses the “Esc” button.



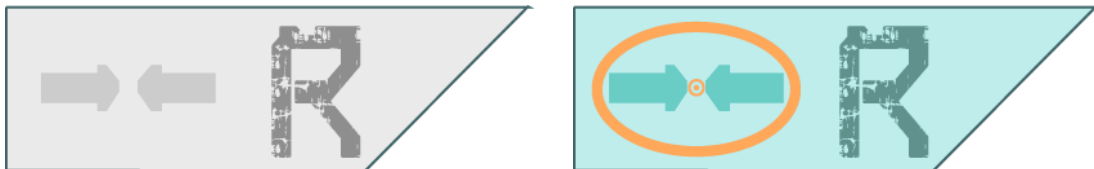
*Figure 8: Inventory (Left) and Options (Right) menu buttons*

- Pick/interact button: This button is locked unless the character is next to an object. In that case, the player can press the keyboard button “F” and the object will be added to the inventory (if it is a pickable object) or the character will interact with it (if it is an interactable object). When the button is locked, the HUD shows it in grayscale:



*Figure 9: Locked (Left) and unlocked (Right) pick/interact buttons.*

- Speak button: When the character is next to the other NPC, the player can press the keyboard button “F” if they want to talk with the NPC. When the button is locked, the HUD shows it in grayscale.



*Figure 10: Locked (Left) and unlocked (Right) speak buttons.*

- Change character button: In the “two characters” scenes, the player can change the character anytime. In other scenes, this button does not appear, because the player cannot change characters. When this button is unlocked, the player must press the keyboard “E” to change the character.



*Figure 11: Change character button.*

## 7.1.2. Menu Design

There are two menus that the player can open anytime: The inventory menu, located in the left side of the screen, that shows all the objects collected by the player; and the options menu, located in the right side of the screen, with three options (continue, enable/disable sound and exit the game):

- Inventory menu: This menu shows all the items collected by the player in all the scenes, and a brief description of each one. When the inventory is opened, it does an animation outside/inside in the left side of the screen, and keeps the open/close menu button attached to its right side. When the menu is closed, it does the opposite operation:

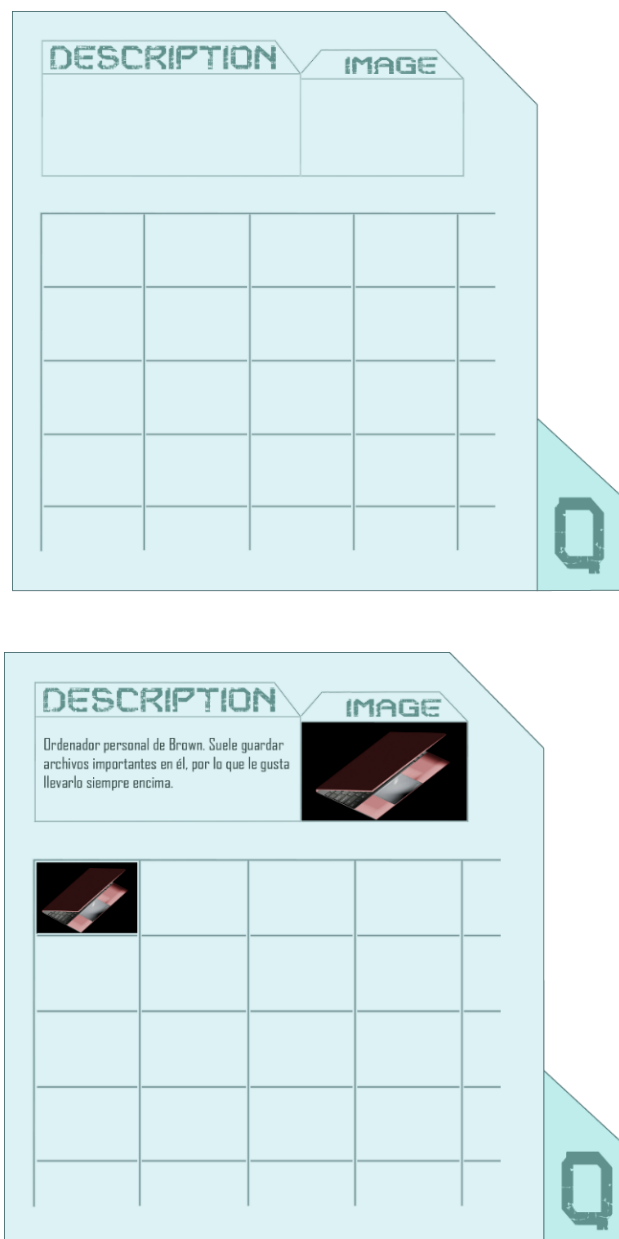


Figure 12: Empty Item Menu (top) and Item Menu when an object is picked (bottom)

- Options menu: This menu shows the “Continue”, “Sound” (enable/disable), and “Exit” options. When this menu is opened, it does an animation outside/inside in the right side of the screen, and keeps the open/close menu button attached to its left side. When the menu is closed, it does the opposite operation:

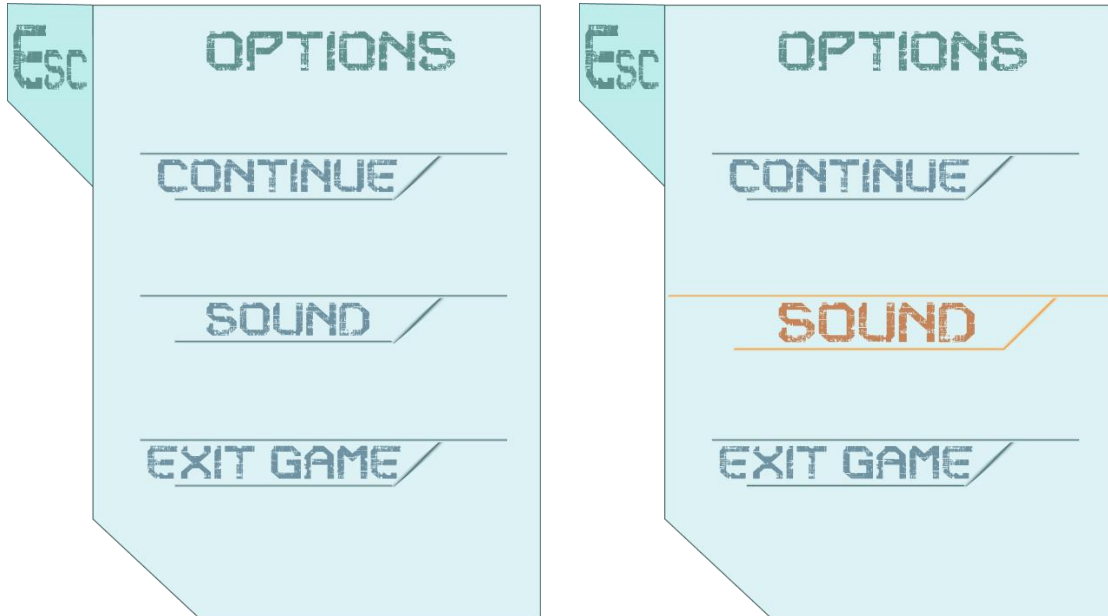


Figure 13: Options Menu without options selected (left) and Options Menu with Sound option selected (right).

### 7.1.3. Image Renders

When the player picks an item, it should be on the inventory menu, with a picture and a brief description. This picture cannot be a simple screenshot of the item in the scene. That is why the objects should be rendered in the modelling tool before its export.



Figure 14: Brown's PC in the scene.



Figure 15: Brown's PC render in 3ds Max.

#### 7.1.4. HUD

At the start of this section, it has been said that the in-game Interface, the menus and the renders work as one to create the HUD. The result of this melting is shown below.



Figure 16: Interface with hidden menu





Figure 17: Interface with Items menu and Options menu opened

## 7.2. 3D Art



Figure 18: Main characters final face

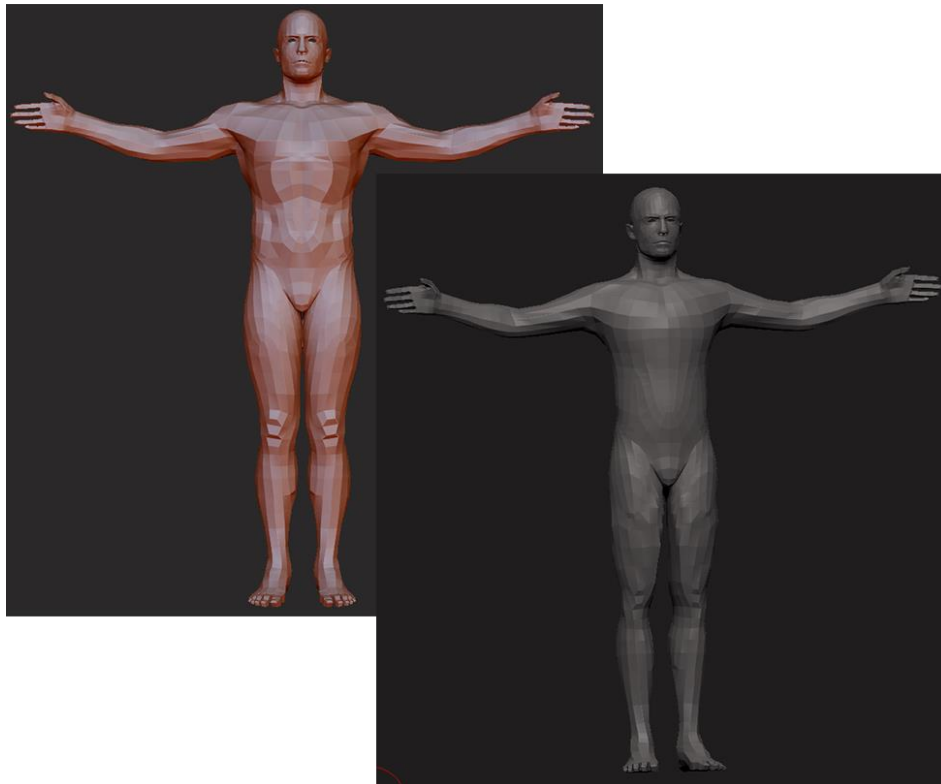
As a three-dimensional game, all the game assets are in 3D (excepting the HUD). The scene, the characters and all the objects has been modelled in 3D, in different modelling tools. For the inorganic modelling (environment and objects), the chosen tools are Autodesk 3ds Max 2017 and Autodesk Maya 2017. All the organic models (characters) has been made with ZBrush, in its versions 4R7 an 4R8.

### 7.2.1. Characters

Two characters are the protagonists of the game, and they have been created in ZBrush from simple low-poly base models. ZBrush is a program that allows the user to modify the object topology without the vertex/faces interaction used in Maya or 3ds Max. ZBrush way of modifying topology is more akin to sculpting than traditional modelling software. This program works with tools (the model in the canvas) and subtools (divisions of this model, like body, teeth or clothes). The object can be modified in many ways, depending of the selected brush. Some of them are “Move topology”, “Clay”, “Pinch” or “Smooth”. ZBrush allows the user to paint the model too. The software has an intuitive mode called “PolyPaint”, that allows to paint directly the model.

#### 7.2.1.1. *Brown's Model*

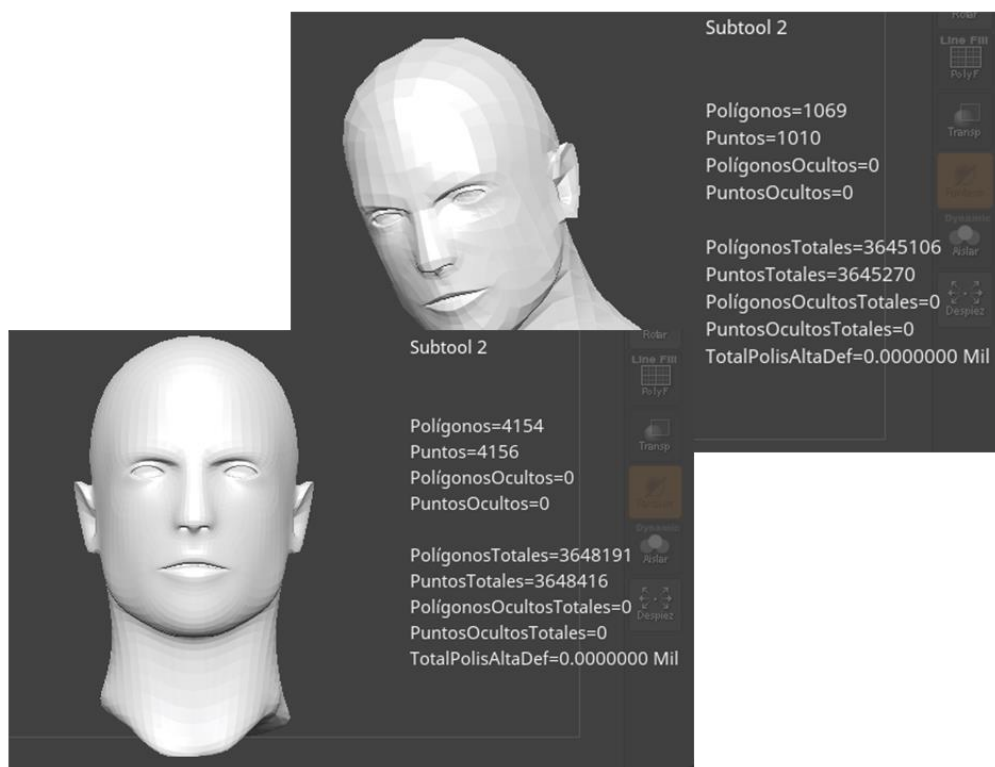
The first character is Brown. The male co-protagonist of the game.



*Figure 19: First changes in the model*

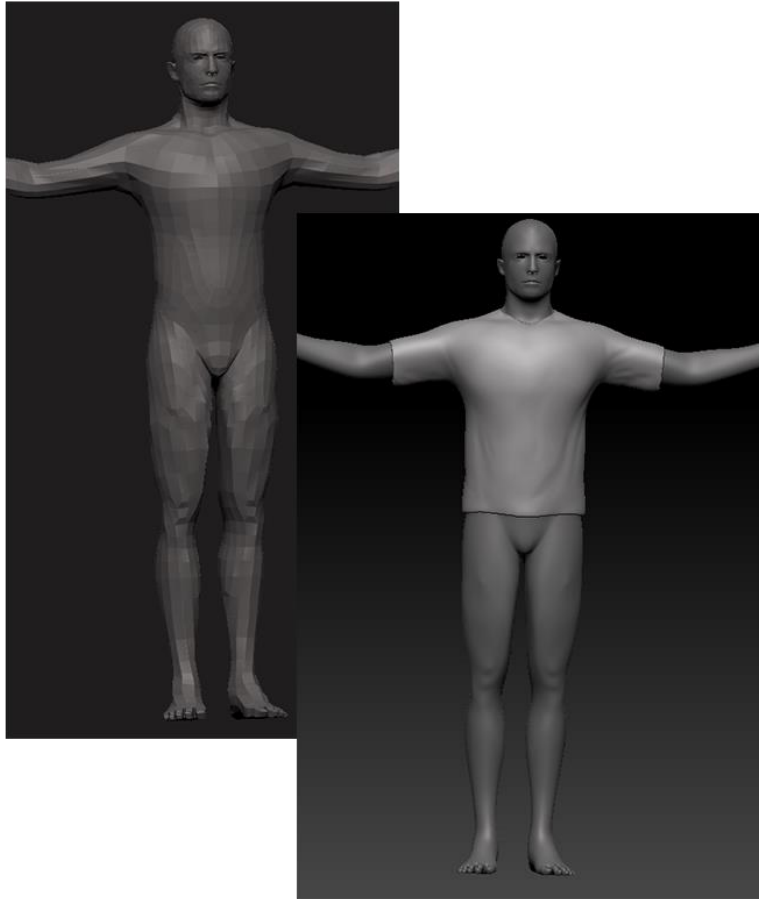
With a low poly model like this one, the first thing to do is adapt to the concept of the character. In this case, the torso volume has been reduced, as same as thigh volume and forearm muscles.

Once the basic body form is acquired, is time to make the geometry subdivision. This operation will make the polygon number grow exponentially, but this is not a problem at this point. Now the subtool is subdivided, and the polygon number had grown, as well as the level of detail. This is one of the most important things in ZBrush: If the subtool has more polygons, the user can add more detail to it. It is clearly appreciated in low subdivisions, where the capacity of add details is very restricted.



*Figure 20: Face after and before the geometry subdivision*

When the model has enough subdivisions to modify topology as desired, is the moment to start modelling details. If the character is dressed, now is the moment to create the clothes. One way to do that is extruding a selected part of the body in other subtool. This operation does not break the body subtool, it simply copies one part of the body and paste as a new subtool with a different size (stipulated by the user). This copy can be modified normally, and it is necessary to achieve a good realism level.



*Figure 21: shirt model and subdivision level up*

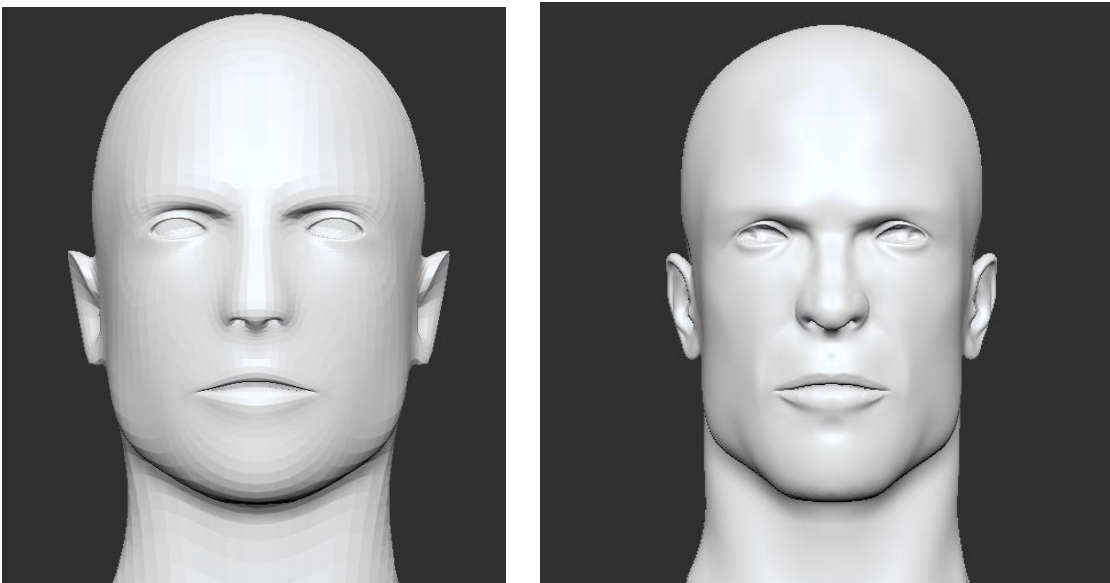
For a realistic shirt model, it is necessary to model some creases, as well as mimic the gravity and wear effects that exist in a real one. For a complete outfit model is necessary to repeat the same operation with trousers and shoes.

When the clothes are modelled, the next step is to create the hair. This can be done extruding a head selection, just like the clothes. In this case, this was not the way to make the hair. Instead of this, the hair is made from a sphere added as a subtool. This sphere is modified as any other subtool, subdividing it and modifying its topology.



*Figure 22: First hair approach and clothes model*

With this level of subdivision, the face should be modified to add more details, and some features to improve the realism of the model. It is necessary to change the face and some reference images should be helpful.



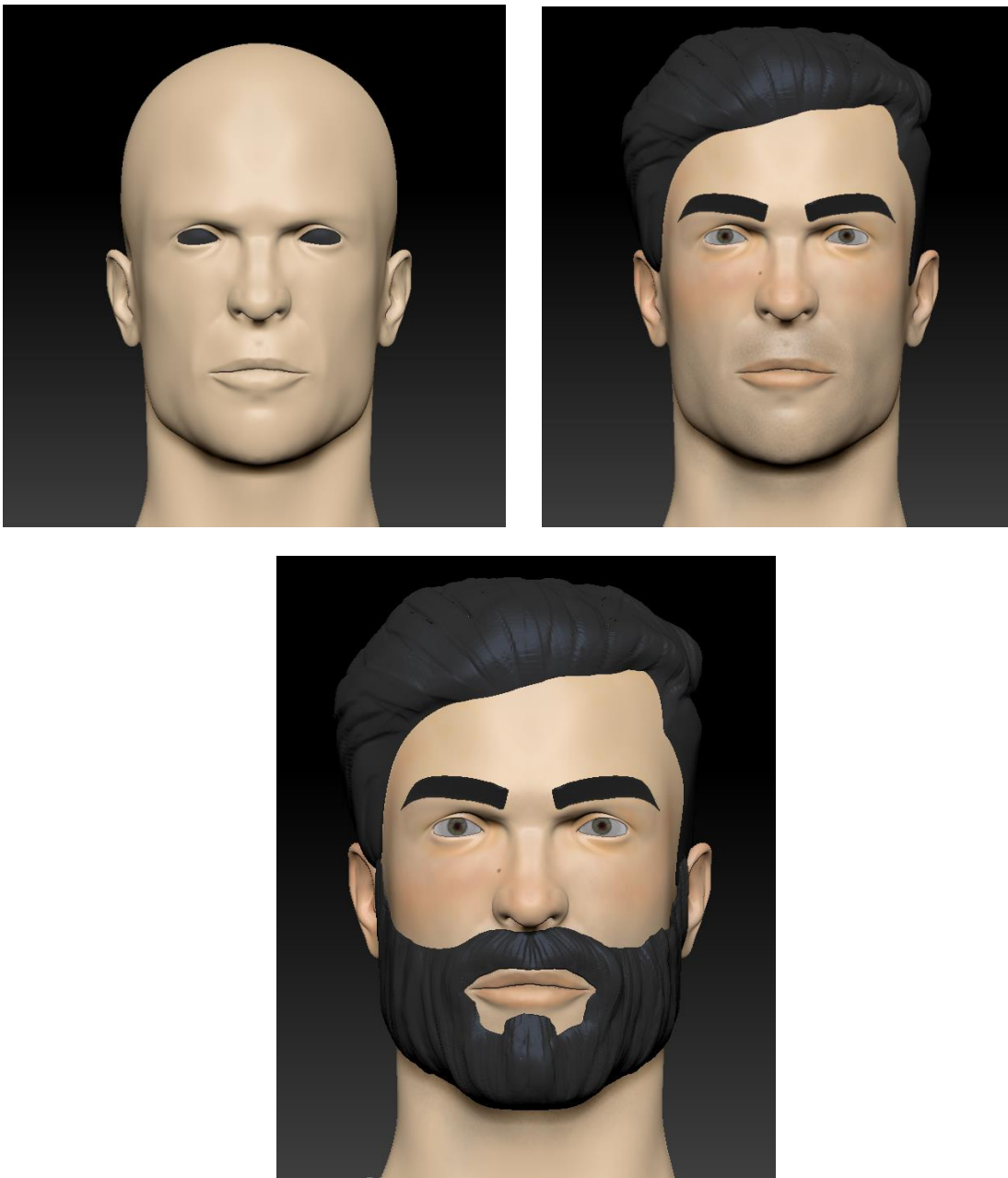
*Figure 23: Face changes to improve realism*

Now, body, clothes and face are completed, but it is necessary to add colour to the model, as well as finish the face and detail it more, adding eyes, eyebrows, and facial hair.

Painting in ZBrush is very intuitive, using its polypaint function. This feature allows the user to paint an object surface without assigning first a texture map, simply adding colour directly to the polygons.

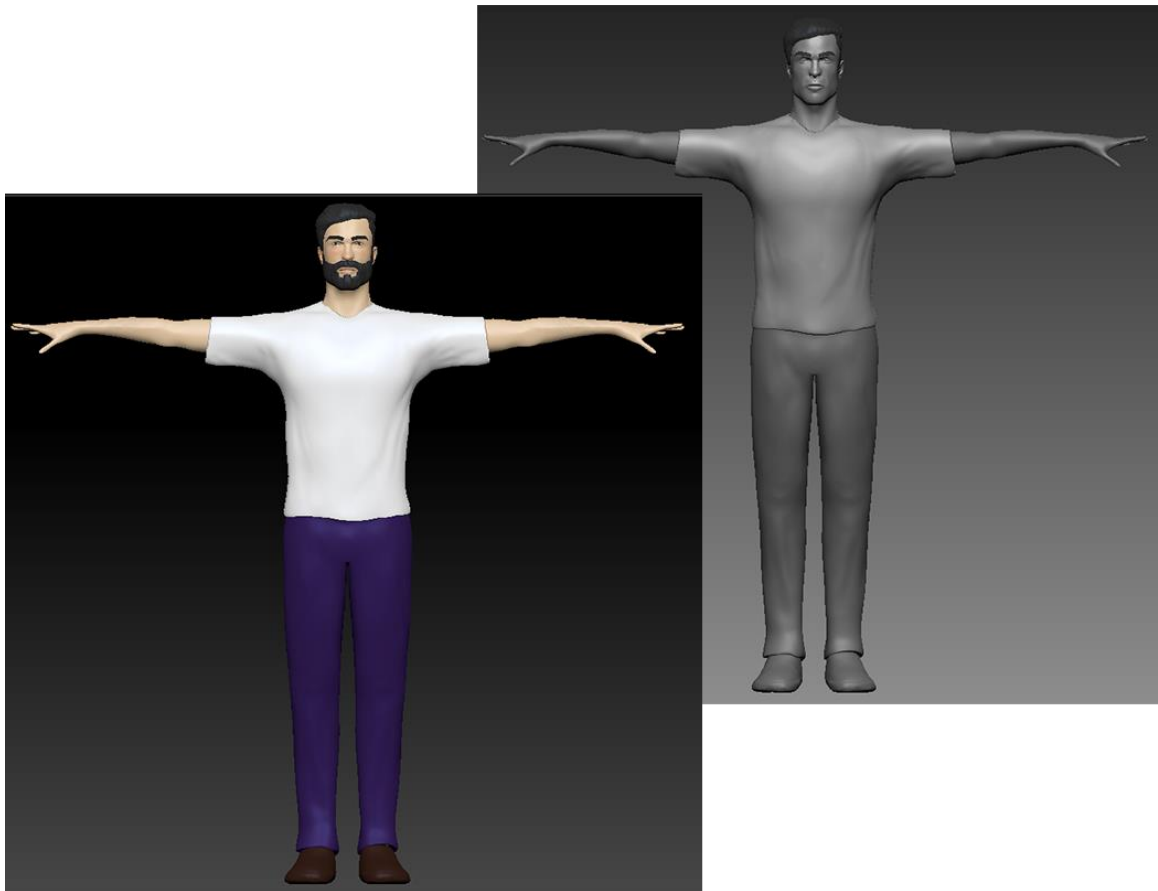
The first step when painting is to set the correct material to each subtool. If the chosen subtool is the face, the material can be a “SkinShade”, but it should not be a metallic material. When the material is assigned, it is time to fill the object with the chosen colour. To do that, the user must press the “FillColor” option in the “Color” menu.

To add colour detail, is necessary to change the stroke of the brush, and the RGB intensity of it.



*Figure 24: Face with base colour, detailed colour and hair and final face*

The face is finished but it is necessary to complete body and clothes. The steps to paint the clothes are like the steps to paint the face.



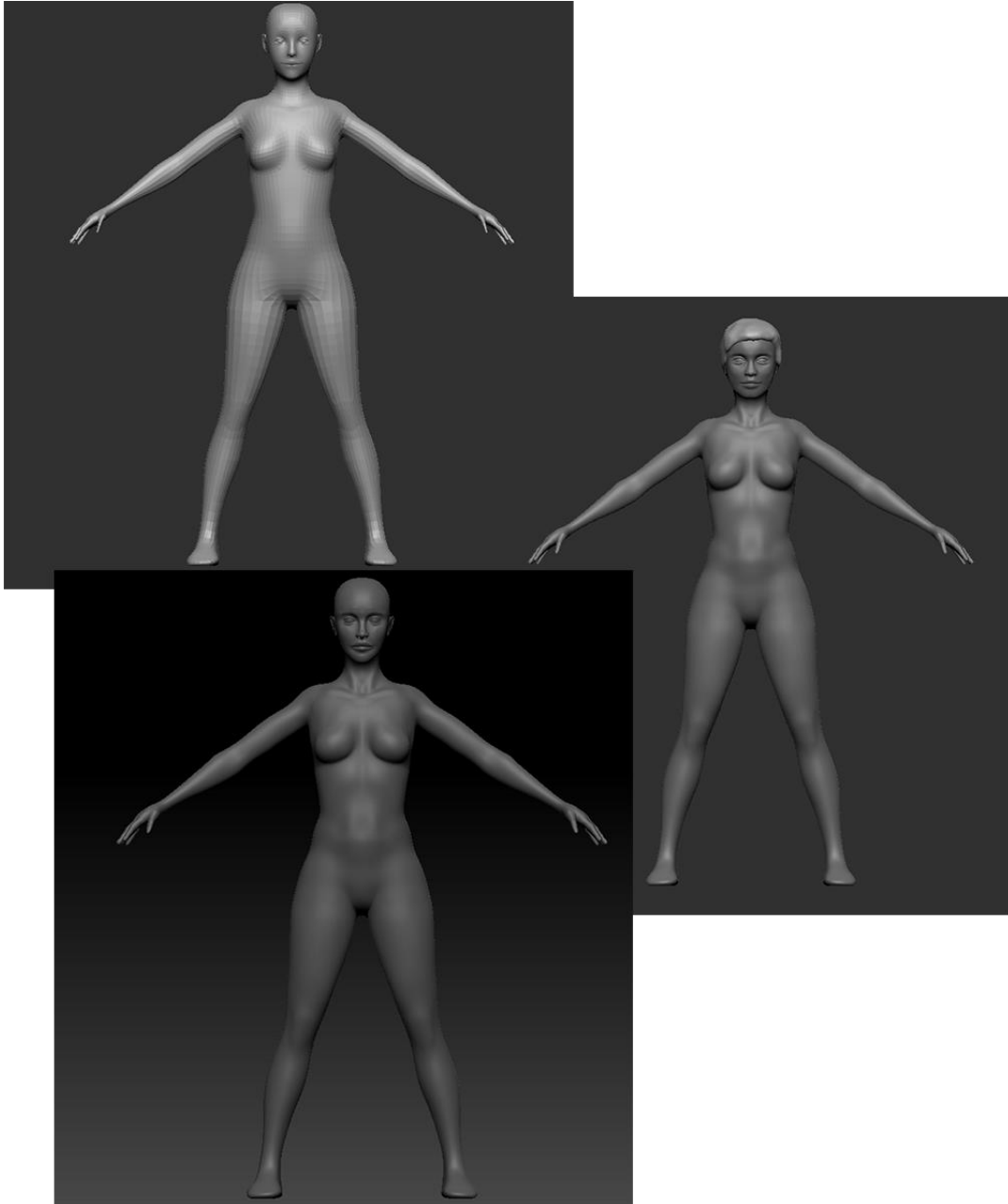
*Figure 25: Painted model*

The model is finished, but there is still a last job to do until this character can be exported to the game engine.

It is necessary to reduce the polygon number, but it is necessary to keep as many details as possible. This objective can be achieved with retopology, UV's maps and Decimation Master.

#### *7.2.1.2. Julie's Model*

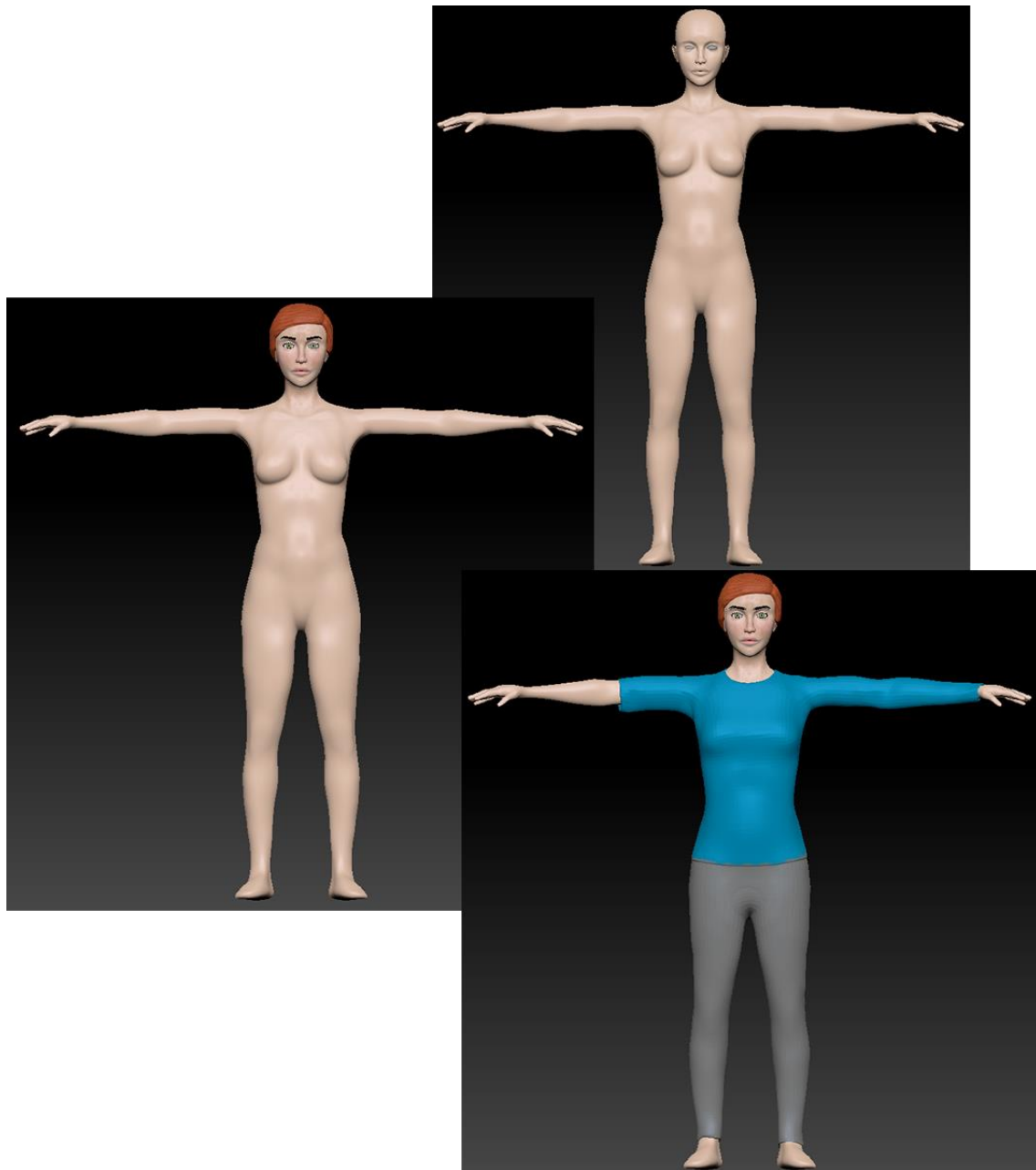
Julie was the second model made in ZBrush. She is the female co-protagonist of the game. The steps to make that character are the same as Brown's model, but with obvious differences. For example, the base model must be different, since the women's anatomy is different to men's.



*Figure 26: Julie's model first steps*

This model had an extra difficulty: the body was not on T-pose (arms perpendicular with the body and legs closed).

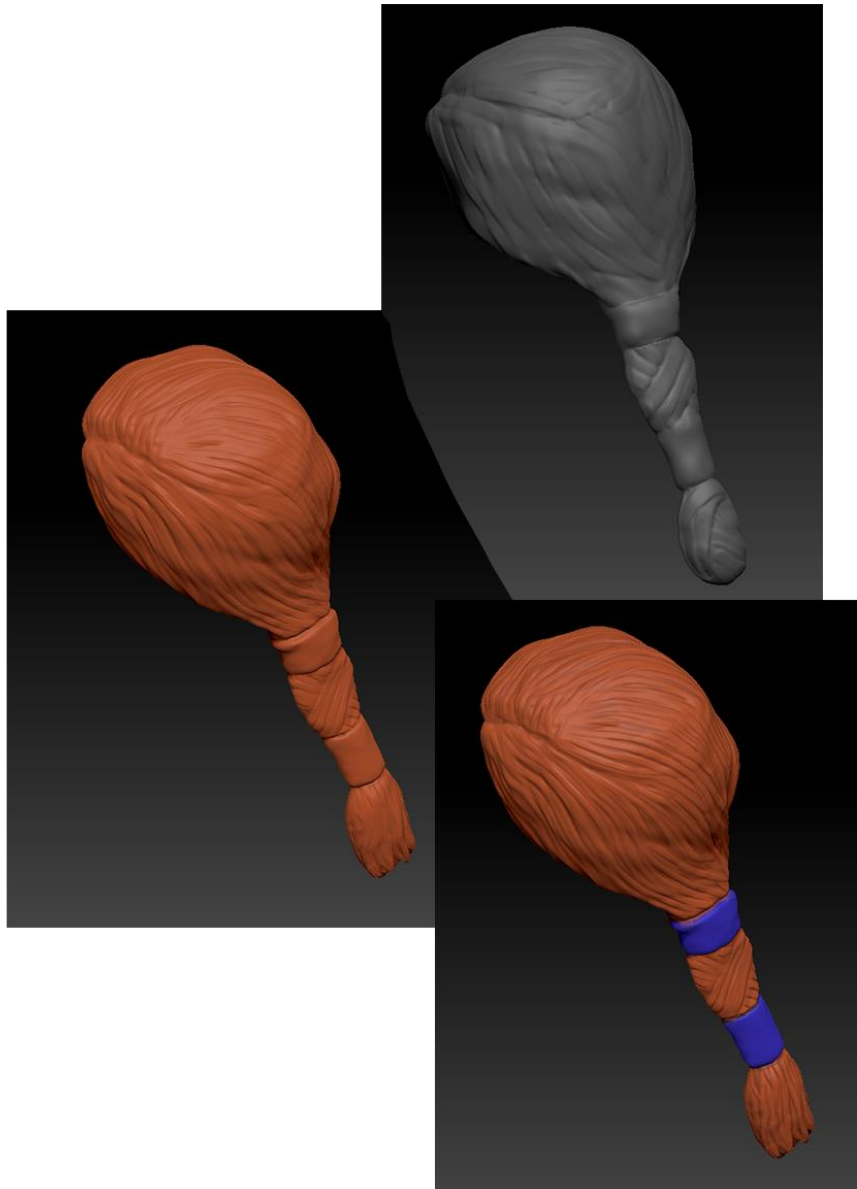




*Figure 27: Adding detail, hair and clothes*

Usually, women's hair is longer and bulkier, so it is necessary to model it with a higher level of detail than the male hair.

To give hair more realism, a darker colour has been added to the deepest parts.



*Figure 28: Hair model, base colour and detailed colour*

In Brown's model, eyelashes had not been added. In women, that part of the face should be larger, so it is necessary to model that.

Of course, colour and model detail has been added to visible body parts. It is a waste of time to add, for example, corporal hair, if clothes are going to hide it.



*Figure 29: Julie's face with base colour, adding details and final face*

### *7.2.1.3. Animation*

The chosen tool for the animation of both characters is Mixamo, a very intuitive tool in a web page that allows the user to animate a model that does not have rigged before. This page takes the model, ask the user to set the orientation of the model and its place markers and rigs it automatically. If it is possible, is better to export the model without textures, due to the size of this file will be very low without them. Brown's model size was 60 Mb textured and 1 Mb without textures. Mixamo is powerful, but is very slow too. If the file has a big weight, it is sure this tool will slow down.

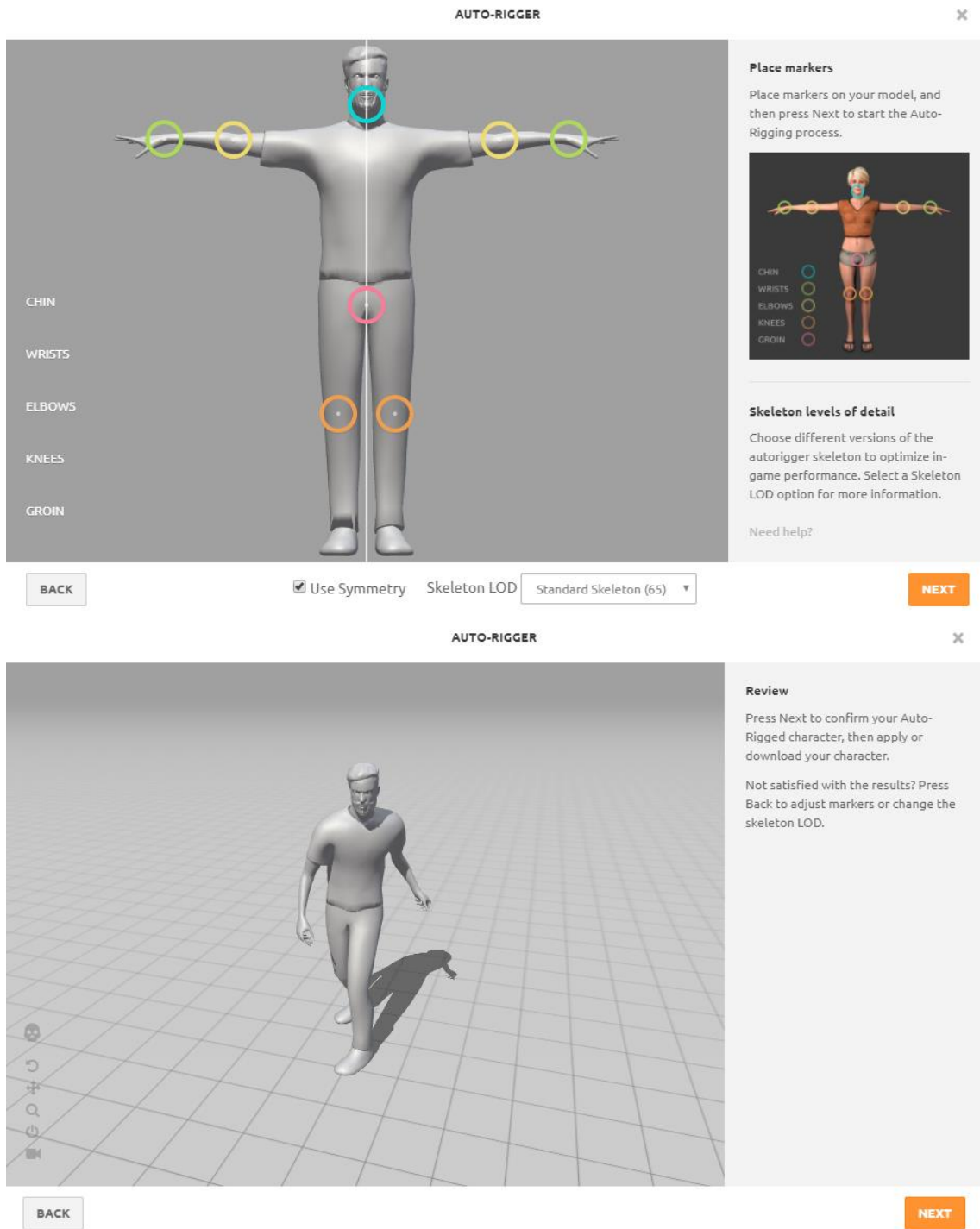


Figure 30: First steps in Mixamo

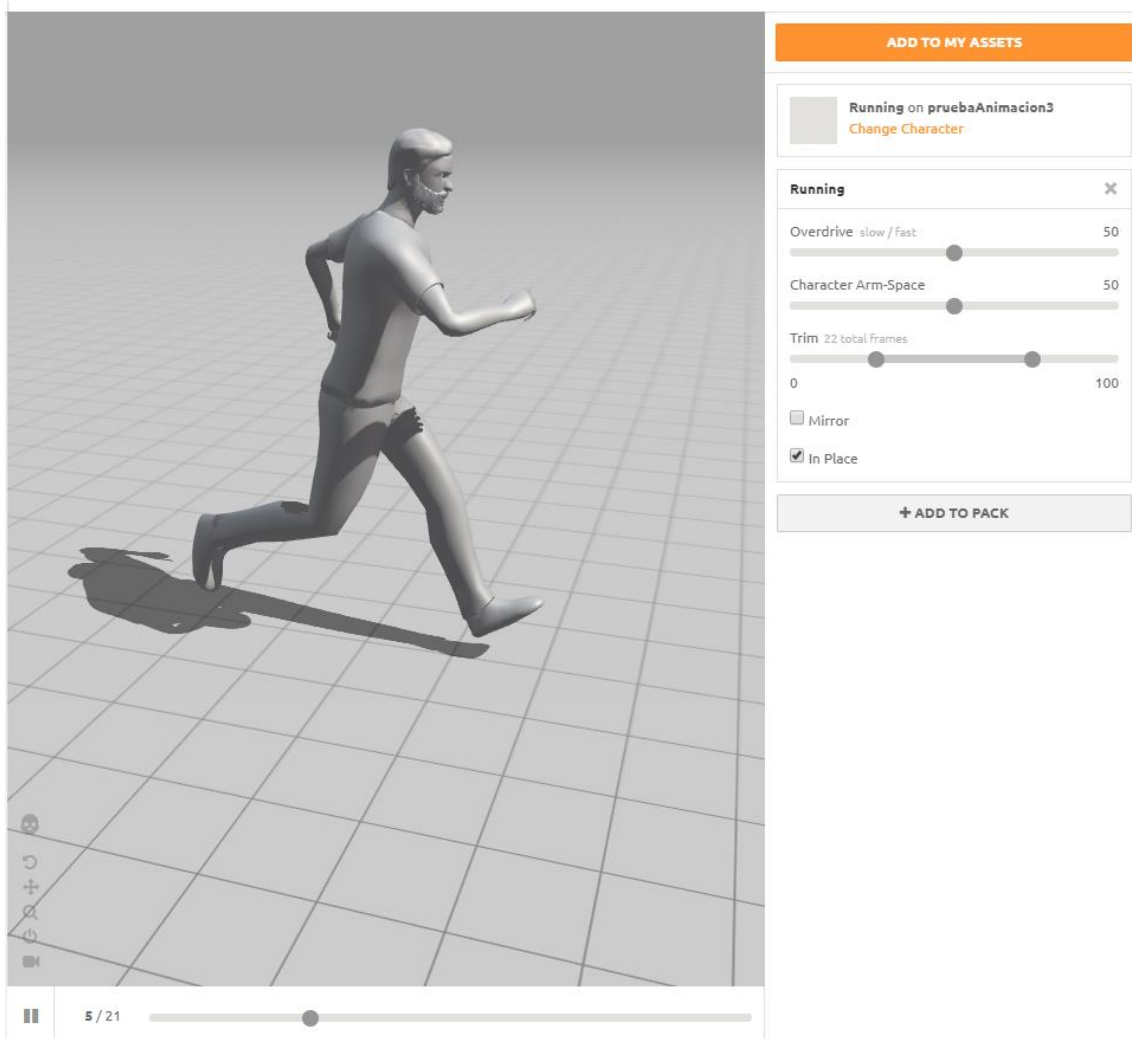


Figure 31: Animation selection and modification

Obviously, male and female models should not have the same animations. Mixamo solves that problem adding a long list of male and female options.

The process with Julie is the same, but selecting different animations.



*Figure 32: Male (top) and female (bottom) base animations*

When the user chooses the animations, they only must download them. Mixamo saves objects as fbx files, perfectly suitable in Unity. Each object has an animation. If the user downloads an animation pack instead of every animation in a different download, the model will be downloaded only once, and the other objects will be only the animation.

When all the animations needed are in Unity, the next step must be to create the Animator Controller and add it all the animations and transitions as needed.

The final step is as simple as add the Animator Controller to the Animation component in the GameObject.

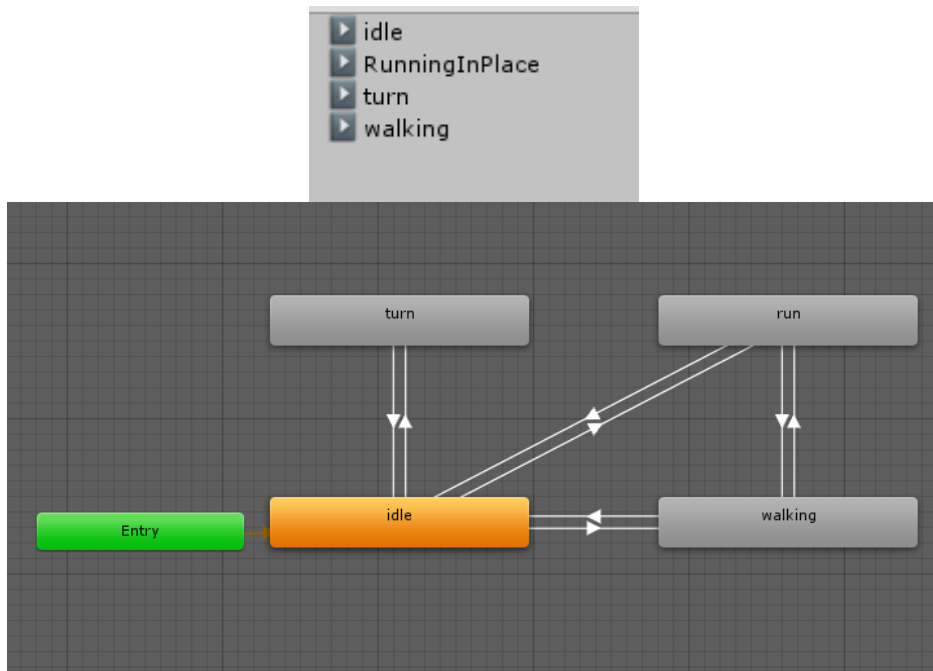


Figure 33: Animations selected and animation controller in Unity

The animations are added to the character, but it is necessary to script in the character movement class how and when the animations should be active.

```

void Moves(){
    //if the player does not press turn button
    if (movementInputValue >= 0) {
        //if the player presses run button and movement button
        //if only the run button is pressed nothing happens
        if (runInputValue != 0 && movementInputValue != 0) {
            isRunning = true;
        }
        //if the player press movement button
        else if (movementInputValue != 0) {
            isWalking = true;
        }
    }
}

void Animating(){
    anim.SetBool ("Walking", isWalking); //if the player is walking, the AC uses the Animation Walking
    anim.SetBool ("Running", isRunning); //if the player is running, the AC uses the Animation Running
}

```

Figure 34: Controlling animations with scripts in Unity

If this part is ended, the animation is completely finished, and the character should move correctly over the scene.

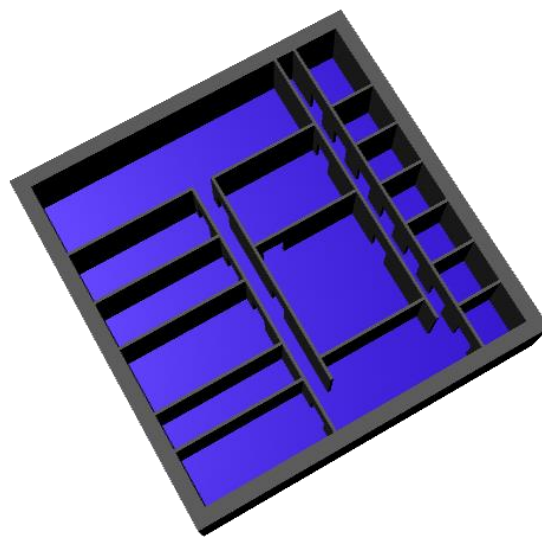




*Figure 35: Animated character in the game*

### **7.2.2. Environment**

The game is bounded in a building, where the characters live, so it should have all rooms needed to make a common life. This building has a bedroom zone, in the east side; a common area with living room and dining room in the centre; the work zone, in the west side; and the library, located in the north zone. The model of the building has been made in Autodesk Maya 2017 and Autodesk 3ds Max 2017.

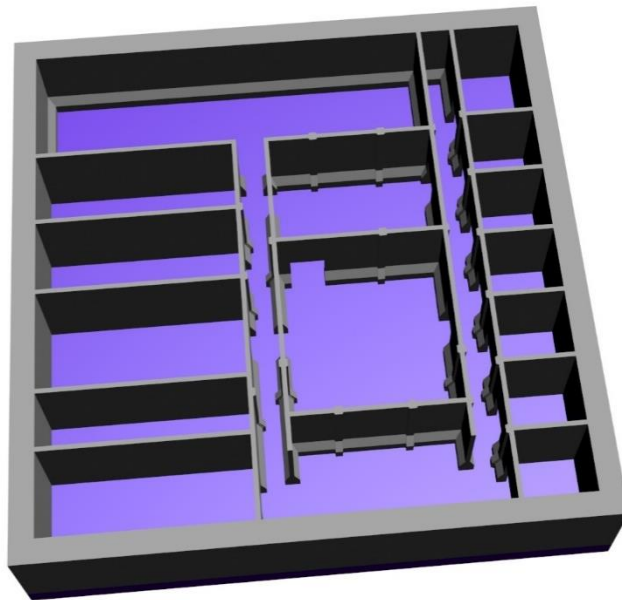


*Figure 36: First model of the building*



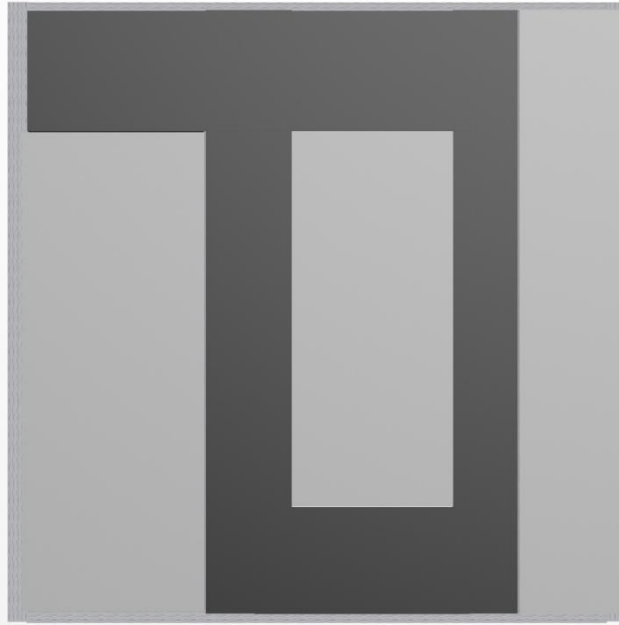
The first approach was a very simple model with floor, walls and door holes. In this model, and the following ones, the two corridors between zones were so narrow, but it was not discovered until its export to Unity.

The next step was to add modern building details, like iron columns or ventilation ducts.

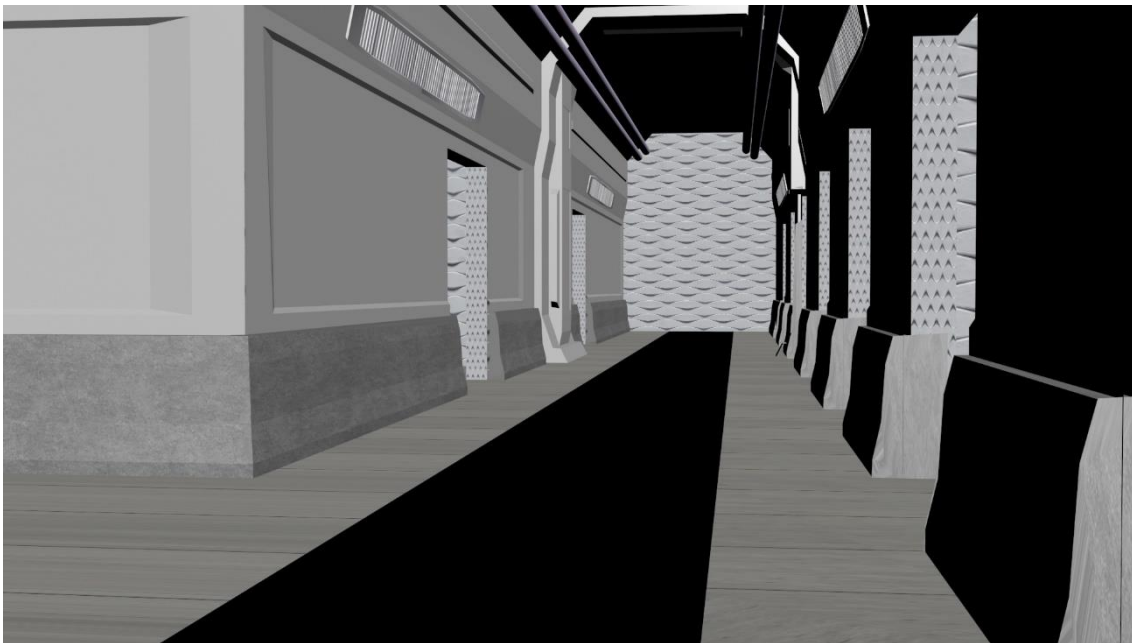


*Figure 37: Adding details*

Now, it is necessary to add the roof. There are two types of roof in this building: a flat one, on the rooms (west, centre and east), and a compound one in the corridors and the library.



*Figure 38: Roof distribution (compound roof in dark colour)*



*Figure 39: Detailed environment model with simple textures*

When the geometry is finished, it is the time to export this object to Unity. The appropriate materials for the environment are added in the game engine. Simple materials do not give a very realistic sensation, because of that, smart materials are a better option. These materials have normal maps and a variable level of randomness included, and other functions that enhance realism.

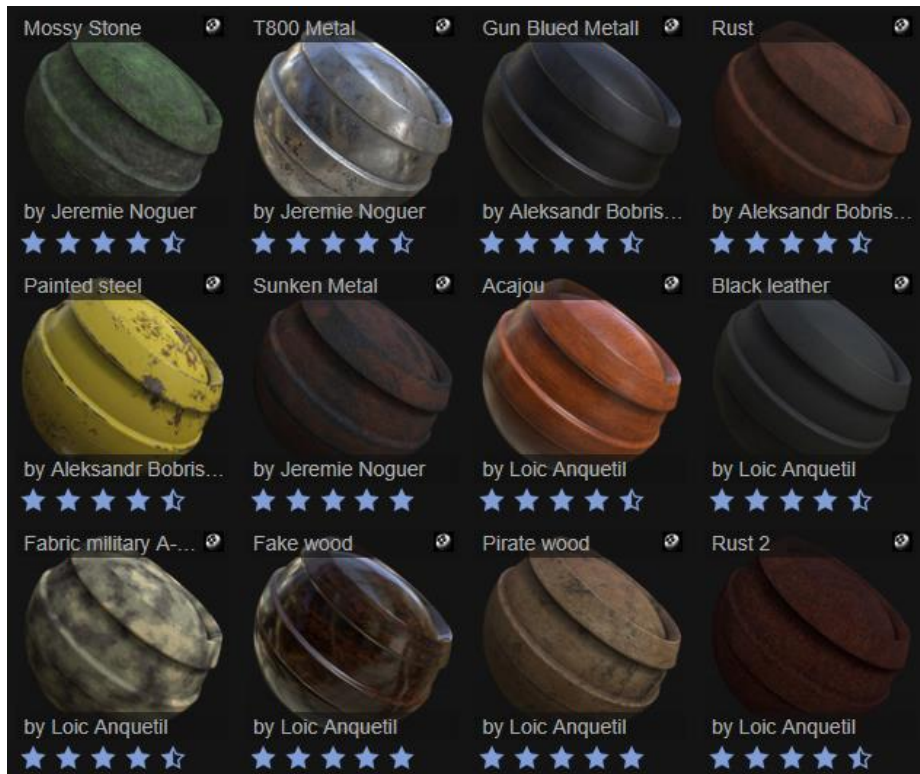


Figure 40: Smart materials in Substance Share

### 7.2.3. Props and objects

There are objects in the game that the player is not able to interact, but they are indispensable to make a scene. Without these objects, scene will be empty, and sensation of emptiness in a videogame is really bad.

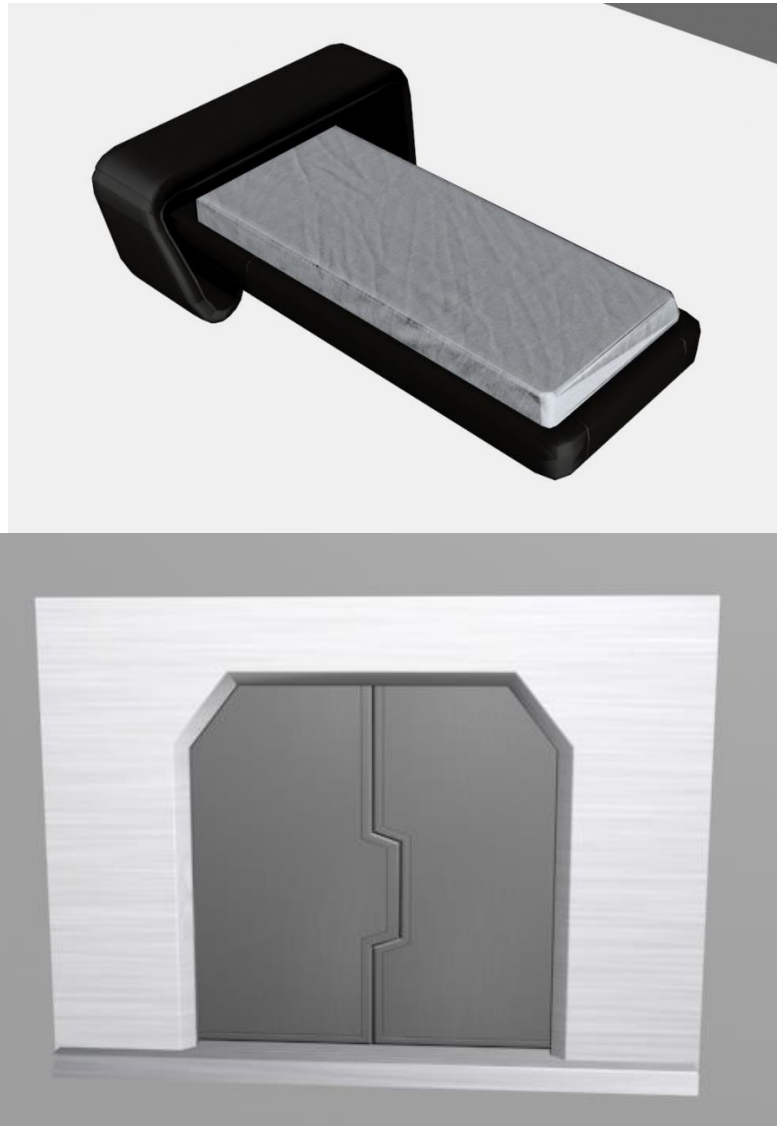
For the interactable objects it is necessary to know where are they going to be, and the size they will have.

For this project, some assets have been downloaded and imported directly on the Unity project, other have been modified, and other has been created right from the start.

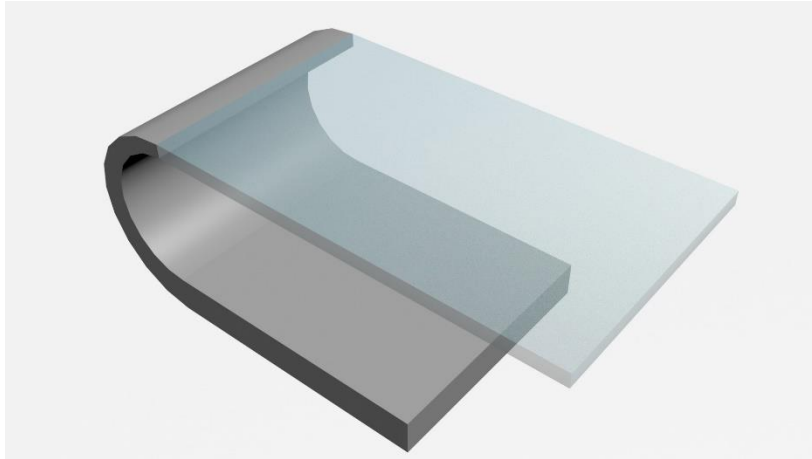
All these objects are 3ds Max or Maya models.



*Figure 41; Spaceship and intercom models. Directly imported to Unity*



*Figure 42: Bed and door models. Modified and imported.*

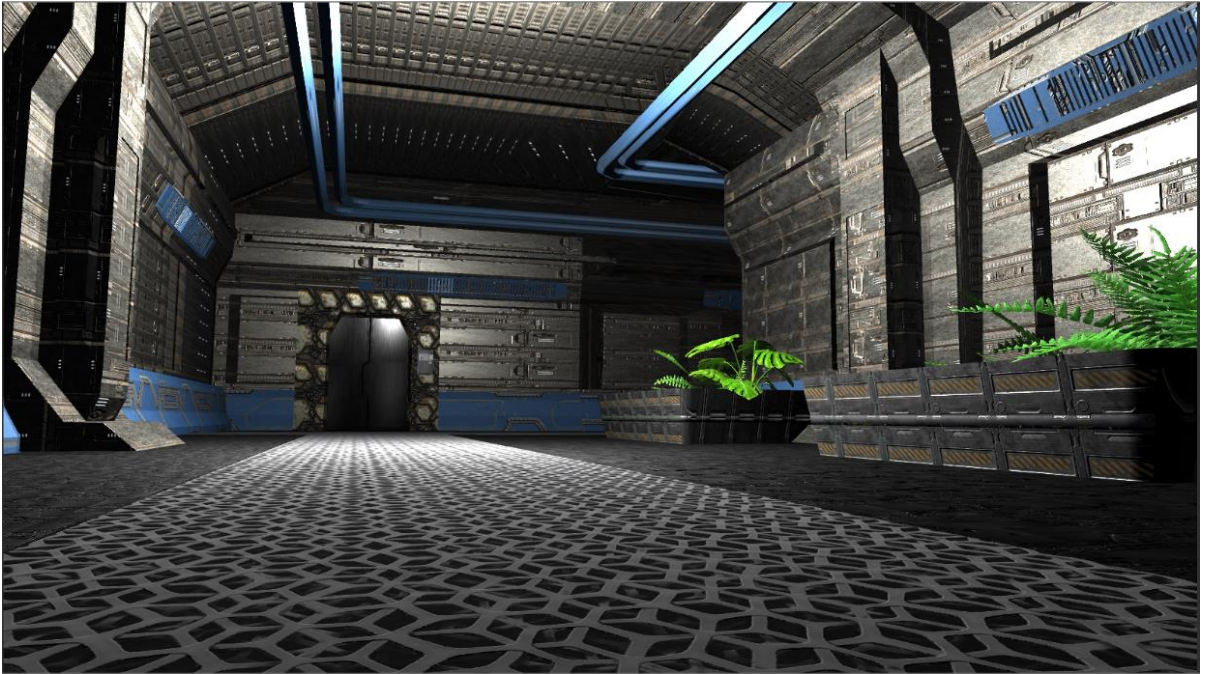


*Figure 43: Table modeled and imported to Unity. Added materials to the metallic part in the engine.*

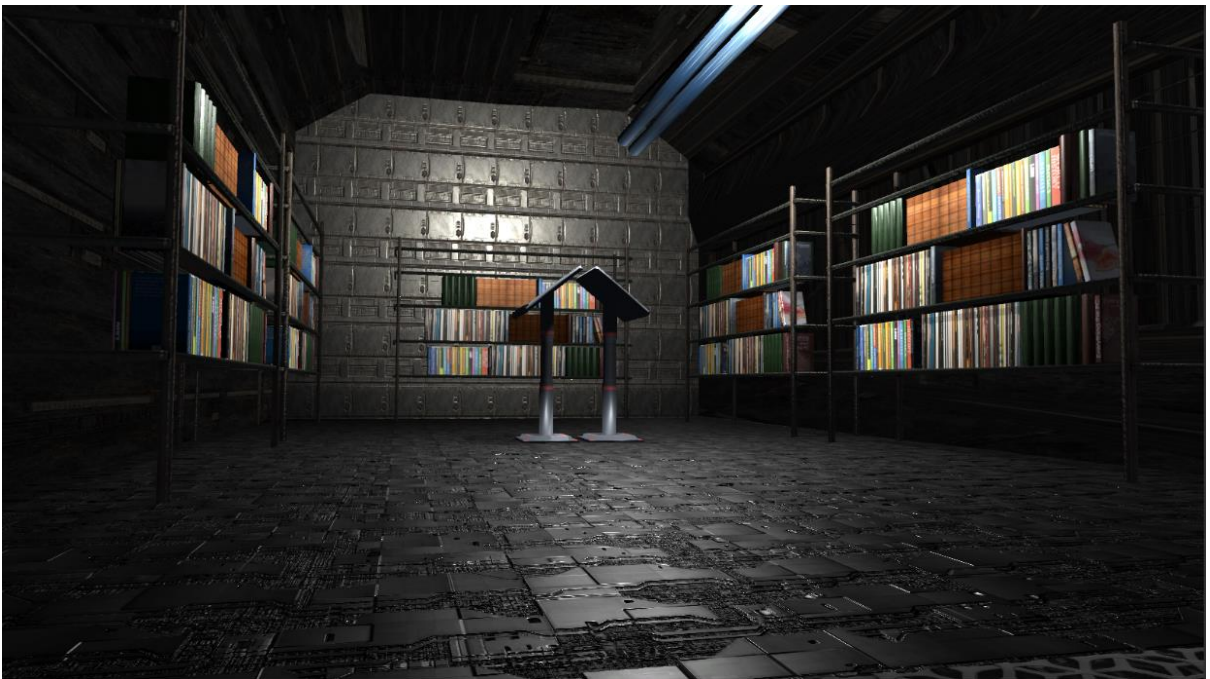


## 8. Results

In this section is shown the final result of the game in images, taken directly from the project.



*Figure 44: In-game capture 1*



*Figure 45: In-game capture 2*



Figure 46: In-game capture 3

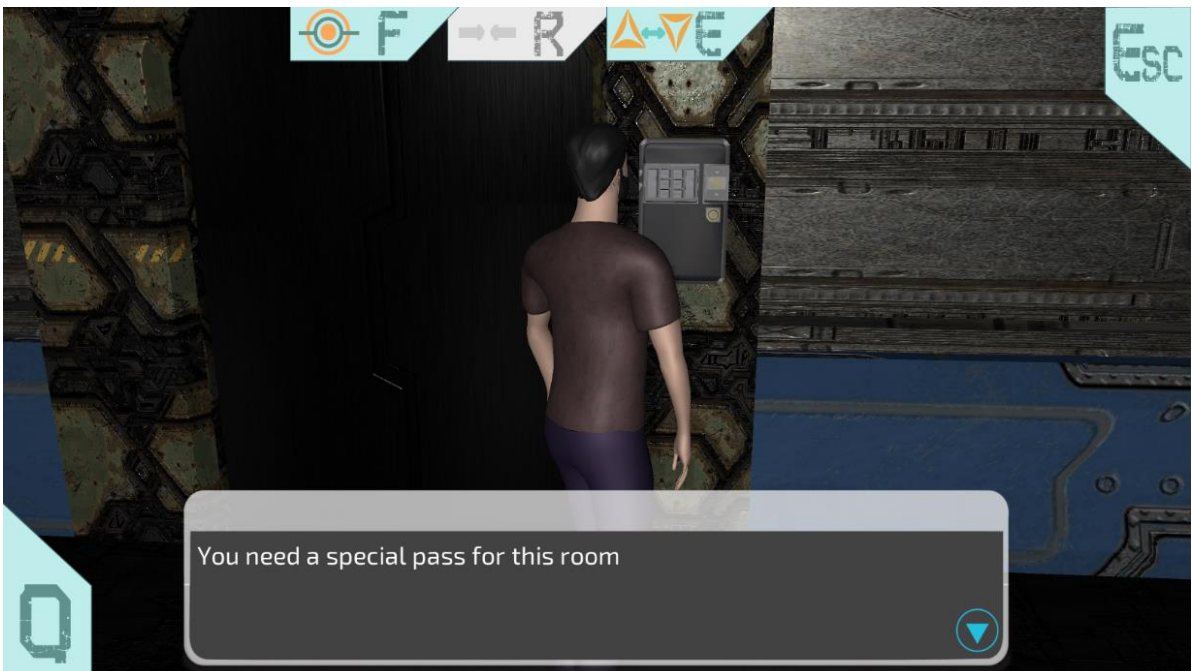


Figure 47: In-game capture 4





*Figure 48: In-game capture 5*



*Figure 49: In-game capture 6*





Figure 50: In-game capture 7

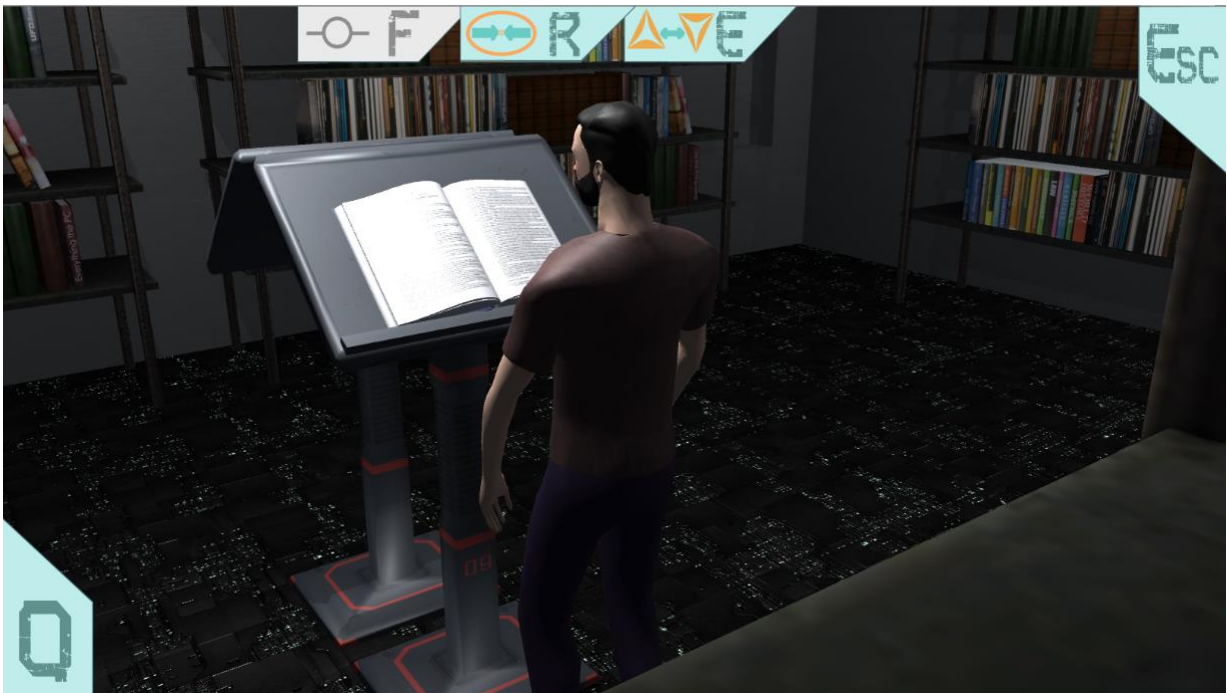


Figure 51: In-game capture 8



*Figure 52: In-game capture 9*

## 9. Conclusions

The final chapter of this Project exposes the goals achieved, the accordance with the planning and a personal conclusion.

### 9.1. Achieved goals

At the beginning of the project, some goals were proposed as orientation to make a good project. Now these goals are dissected and reviewed:

- Designing a game map with different zones where the characters inhabit.
  - ✓ The environment is created and implemented, and the zones are well differentiated.
- Modelling two realistic characters in ZBrush.
  - ✓ Two characters are modelled in this tool. The quality of this models should not be evaluated by the designer.
- Creating assets and putting them into the scene, in order to enhance the story.
  - ✓ Some assets have been created, and other ones have been downloaded from free 3D model webpages.
- Creating a visual and interactive story, that can be completely understood only if the player pay attention to the scenario and items.
  - ✓ The story moves with the player, and to understand it completely the player must find the objects in the scene.
- Creating an Artificial Intelligence that follows the character controlled by the player and allows them to dialogue with it.
  - ✓ This AI is programmed and implemented. The player can change the character and the AI changes too. When the character is next to the AI, they can interact.
- Animating and implementing the couple of characters, with walking, running, and idle animations.
  - ✓ Both characters have been animated and implemented in the game.
- Create a good camera movement that allows the player to pay attention to the scene details.
  - ✓ The camera moves with the character, and does not interference with the gameplay.

## 9.2. Real workflow

The initial planning was not faithfully fulfilled, tasks length was not exactly as prevised, and some tasks cannot be completely finished.

The game is playable, with a beginning and an end, but some points cannot be finished:

- The Item menu of the HUD can be opened and closed, and it displays the picked objects, but the movement between that objects is not implemented.
- The options in the Options menu are not implemented.
- The dining room is empty (this room do not have objects, items or furniture at all).

	FEB	MAR	APR	MAY	JUN	JUL	OVERALL
TASK 1	25	5					30
TASK 2	5	20					25
TASK 3	5	25	10				40
TASK 4			10				10
TASK 5	5	15	5				25
TASK 6				10			10
TASK 7			20	10			30
TASK 8			15	25			40
TASK 9				10	10		20
TASK 10					10		10
TASK 11			5	10	5		20
TASK 12					40		40
OVERALL	40	65	65	65	65	0	300

*Table 9: Initial schedule*

	FEB	MAR	APR	MAY	JUN	JUL	OVERALL
TASK 1	14	7	7				28
TASK 2	12	12	5				29
TASK 3		18	23				41
TASK 4				6			6
TASK 5	5	17	7				29
TASK 6				9			9
TASK 7			16	21			37
TASK 8			15	19			34
TASK 9				10	12		22
TASK 10					13		13
TASK 11			10	10	5		25
TASK 12					32	6	38
OVERALL	31	54	83	75	59	6	311

*Table 10: Final schedule*

### **9.3. Personal conclusions**

The last part of this project is a personal evaluation of the work and the final result.

Personally, I am gratefully surprised due to the final outcome of the game. There are some aspects to enhance, of course, but I think the game can be a good skill show.

I prefer modelling or designing instead of programming, but in this project, the section that I enjoyed more. Unity allows people without great programming knowledge to make games, and this is an aspect to thank.

I learnt a lot designing and developing this project, and that is the part I am proudest. Some tools were strange to me in February, but now I know more about how ZBrush, Maya or Unity work.

This game has an open end, and it makes me think if a second part is viable. I do not like to keep with the same style and characters much time (I think this is not a good learn mode), but this story can be worth to try.

## 10. Bibliography

Ambient narrative: <http://www.efefuturo.com/noticia/la-narrativa-del-videojuego-quiere-eclipsar-despojarse-la-levedad/>

Lore: <http://dictionary.cambridge.org/dictionary/english/lore>

[www.turbosquid.com](http://www.turbosquid.com)

[www.CGTrader.com](http://www.CGTrader.com)

[www.Free3D.com](http://www.Free3D.com)

HUD: [https://en.wikipedia.org/wiki/HUD\\_\(video\\_gaming\)](https://en.wikipedia.org/wiki/HUD_(video_gaming))

Unity webpage: <https://unity3d.com/>

GitHub webpage: <https://github.com/>

ZBrush webpage: <http://pixologic.com/>

Maya webpage: <https://www.autodesk.es/products/maya/overview>

3ds Max webpage: <https://www.autodesk.es/products/3ds-max/overview>

Marvelous Designer webpage: <https://www.marvelousdesigner.com/>

Adobe Illustrator webpage: <http://www.adobe.com/es/products/illustrator.html>

Fungus webpage: <http://fungusgames.com/>

Fungus in Unity Asset Store: <https://www.assetstore.unity3d.com/en/#!/content/34184>

NPC: [https://en.wikipedia.org/wiki/Non-player\\_character](https://en.wikipedia.org/wiki/Non-player_character)

Plugin: [https://en.wikipedia.org/wiki/Plug-in\\_\(computing\)](https://en.wikipedia.org/wiki/Plug-in_(computing))

ZBrush subtool: <http://docs.pixologic.com/reference-guide/tool/polymesh/subtool/>

ZBrush brushes: <http://pixologic.com/zbrush/features/brushes/>

Zbrush polypaint: <http://pixologic.com/zbrush/features/PolyPaint/>

Flowchart: <https://en.wikipedia.org/wiki/Flowchart>

Smart materials: [https://share.allegorithmic.com/libraries?by\\_category\\_type\\_id=7](https://share.allegorithmic.com/libraries?by_category_type_id=7)