



UNIVERSITAT
JAUME I

Corinthian: a ramified narrative video game with RPG genre mechanics

Santiago Llamas Pereira

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

May 22, 2023

Supervised by: Diego José Díaz García, PhD.



To Ana and my family

ACKNOWLEDGMENTS

First of all, I would like to thank my beloved partner Ana, my family and my friends for their unconditional support, and my Final Degree Work supervisor, Diego José Díaz García, PhD, for his guidance and advice.

ABSTRACT

Corinthian is a visual novel videogame that narrates the story of a federal agent who arrives in a fantasy village far from all the cities of the kingdom, where a murder has taken place, and all the inhabitants are suspects. This project consists of an implementation of RPG mechanics, narrative design and 2D art.

CONTENTS

| | |
|---|-----------|
| Contents | v |
| 1 Introduction | 1 |
| 1.1 Keywords | 1 |
| 1.2 Work Motivation | 2 |
| 1.3 Objectives | 2 |
| 1.4 Environment and Initial State | 2 |
| 2 Planning and resources evaluation | 5 |
| 2.1 Planning | 5 |
| 2.2 Resource Evaluation | 6 |
| 2.3 Time verification | 8 |
| 3 System Analysis and Design | 9 |
| 3.1 Requirement Analysis | 9 |
| 3.2 System Design | 10 |
| 3.3 System Architecture | 16 |
| 3.4 Interface Design | 16 |
| 4 Game Design Document | 19 |
| 4.1 Game Overview | 19 |
| 4.2 Game Mechanics | 20 |
| 4.3 Story setting | 20 |
| 4.4 Character design | 21 |
| 4.5 Art Direction | 25 |
| 4.6 Additional information | 28 |
| 5 Work Development and Results | 29 |
| 5.1 Work Development | 29 |
| 5.2 Results | 39 |
| 6 Conclusions and Future Work | 41 |
| 6.1 Conclusions | 41 |
| 6.2 Future work | 41 |

| | |
|-------------------------------------|-----------|
| Bibliography | 43 |
| A Source code and game build | 45 |

INTRODUCTION

Contents

| | | |
|-----|---|----------|
| 1.1 | Keywords | 1 |
| 1.2 | Work Motivation | 2 |
| 1.3 | Objectives | 2 |
| 1.4 | Environment and Initial State | 2 |

Choosing to make a visual novel has been a great motivation for my work and my creativity. By having the freedom to tell a story through images and text, I have had the opportunity to develop my ability to create characters and work on my artistic style. The visual novel has also allowed me to explore themes and emotions in a deeper and more expressive way than other forms of storytelling. It has also allowed me to explore themes and emotions in a deeper and more expressive way than other forms of storytelling. Overall, choosing to do a visual novel has been a catalyst for my creativity and has given me the opportunity to create something truly unique and meaningful.

1.1 Keywords

Narrative, visual design, role playing, visual novel, player decision making.

1.2 Work Motivation

In my first years in this degree, I have considered that video games can be a means of expression on a par with other artistic branches, such as cinema or literature. So this opportunity to do something for myself, to communicate my stories and have them read by gamers is something special.

The decision for the game format comes from my love for visual novels and tabletop role-playing games since I first tried them. Creating one from scratch was the driving force behind my motivation.

1.3 Objectives

The main objectives to achieve with this project are the following:

- ***Tabletop role-playing game adaptation:*** To bring the game experience to a wider audience and make it accessible to those unfamiliar with traditional role-playing games. Creating an exciting interactive experience that is accessible to both experienced and new players, and that respects the essence and excitement of the original format. With this in mind, making a visual novel with active player choice is one of the main objectives as it is a format that adapts well to the narrative adventure feel of classic RPGs.
- ***Character design:*** To design characters for a visual novel is about creating characters that are visually interesting, that reflect their personality and background, and that draw the reader into the story. Designing memorable and unique characters that connect emotionally with the reader and add depth and complexity to the narrative. To do this, I will create reference tables and differentiate the characters by their silhouette, so that the player can instantly recognise them and internalise their appearance.
- ***Narrative structure:*** To design a compelling narrative structure for a visual novel is about creating an immersive experience for the reader that holds their attention and engages them in the story. Writing a narrative structure that is coherent, exciting and surprising, and that allows the reader to experience the story in a distinctive way. Narrative design goes hand in hand with the first goal. designing a story that depends on player choice is no easy task, but adding the randomness factor of classic RPG skills can lead to interesting design decisions.

1.4 Environment and Initial State

The project has been developed by myself, where I have been responsible for its integrity, from programming to character design. Since this project progressed at the same time as my internship, my schedule went hand in hand with this stay, spending every afternoon on the project.

The style I chose for the game is very much inspired by visual novels that go outside the norm and add some kind of player interaction. This is the case of VA-11 HALL-A [6] (See Fig. 1.1). Also, RPGs have been one of my oldest hobbies and having their interaction system based on skill quantification seemed like an interesting idea, see Disco Elysium [24] as one of my biggest inspirations, both for its mechanics and its world, with strong political views and a psychological exploration of the protagonist and the world around him (See how this game treats ability checks in Fig. 1.2). It is also worth mentioning the great influence that indie games known for their engine, "RPG Maker" [5], have had on me when it comes to building a mysterious aura that almost borders on the macabre (See Fig. 1.3). Finally, it is worth mentioning my inspiration for David Lynch's [4] films, where reality is mixed with the subconscious, reflected in Corinthian by a narrator who not only describes what is seen, but also presents a conscious part of the protagonist agent commenting on everything that we would not dare to say out loud. Twin Peaks is his work that has influenced me the most to generate this idea.



Figure 1.1: VA-11 HALL-A mixes making drinks with conversational gameplay.

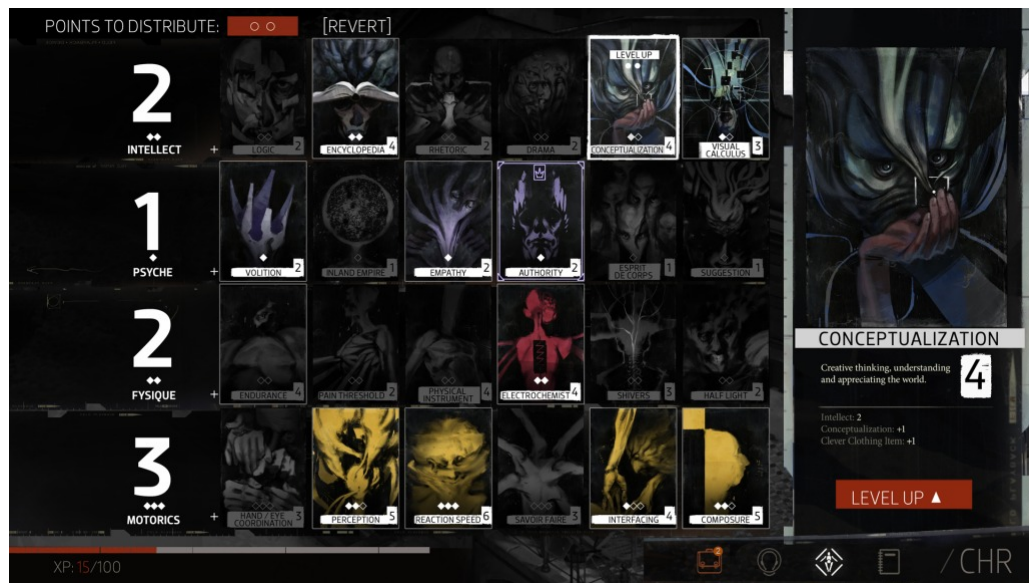


Figure 1.2: Disco Elysium has a wide range of character abilities tied to subconscious.



Figure 1.3: Fear and Hunger horror depiction.

PLANNING AND RESOURCES EVALUATION

Contents

| | | |
|-----|-------------------------------|----------|
| 2.1 | Planning | 5 |
| 2.2 | Resource Evaluation | 6 |
| 2.3 | Time verification | 8 |

2.1 Planning

The planning to be followed for the realization of the project will be following the well-known Agile methodology, more specifically, the scrum methodology.

Scrum is an agile framework used in software development and other complex projects. In this work scheme, work is organised into iterations called "sprints". Each sprint has a fixed duration, usually two to four weeks, during which the team focuses on completing an agreed set of tasks. At the end of each sprint, a functional increment of the product is delivered that can be presented to stakeholders and, if necessary, put into production. [13]

Based on this procedure, the different tasks in the following Gantt chart (see Figure 2.1) will be treated in sprints. Each sprint will consist on performing an assigned task of the current category, holding a meeting with the tutor to review the work and implement the necessary corrections or improvement proposals. Parallel tasks are considered in the same sprint.

2.2 Resource Evaluation

The costs associated with human and technical resources, broken down by software used and hardware used, will be described below.

1. Human resources: Due to the fact that the project has been carried out by only one person, but with different skills, it will be counted as if two workers had worked on the project. The total cost is €8534.4.
 - Programmer: In Spain, a programmer is paid around €14.62 per hour, making a project salary of €4,678.4.[20]
 - 2D artist: In Spain, an artist is paid around €12.05 per hour, making a salary of €3856 for the project.[19]
2. Most of the software used is free, except for one program and the two devices used. The total cost is €429.99.
 - A powerful computer: a computer with intermediate specifications can be used to create games, in this case a mid-range computer costing around €650 has been used.
 - Any graphics tablet: the price is not indicative of its usability, one may cost around €37.99. [10]
 - Unity: free cross-platform video game engine created by Unity Technologies.[21]
 - Visual Studio Code: free source code editor developed by Microsoft, used to write in C sharp.[8]
 - Github: free hosting projects using the Git version control system.[7]
 - Twine: open-source tool for telling interactive, nonlinear stories.[23]
 - Clip Studio Paint: illustration application for Mac OS X and Microsoft Windows developed by Celsys for the digital creation of comics. It costs €42, full price.[18]
 - Trello: free project management software with web and client interface for iOS and android to organise projects.[22]
 - Overleaf: free cloud-based collaborative LaTeX editor used for writing, editing and publishing scientific papers.[11]

The total cost of the project is €8964.39.

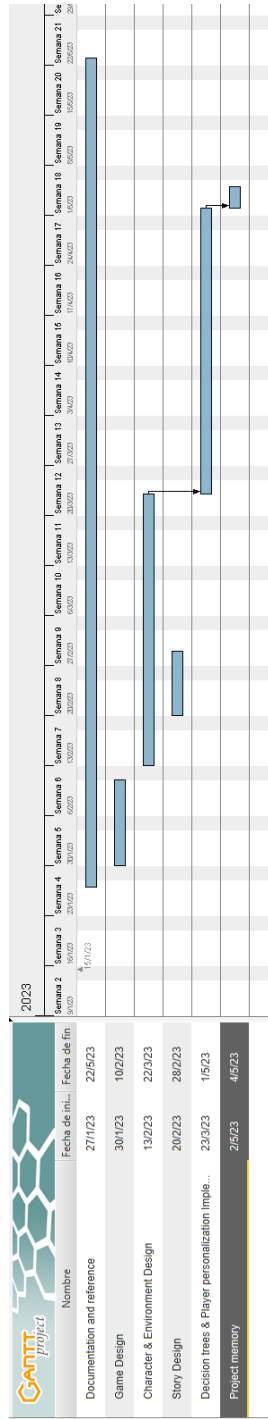


Figure 2.1: Corinthian project planning Gantt chart (made with GanttProject)

2.3 Time verification

First, the time distribution per task is shown in table 2.1 table 2.2 shows the deviation of the planned hours from the actual hours of the project development.

| Task | Type | Hours |
|-------------------------------------|---------------|--------------|
| Technical proposal and GDD | Documentation | 20 |
| Game design | Design | 2.5 |
| Writing story an organise narrative | Narrative | 23 |
| Characters | Art | 34 |
| Background | Art | 15 |
| Player class | Programming | 10 |
| HUD implementation | Programming | 15 |
| Chapter manager | Programming | 30 |
| Dialogue system | Programming | 30 |
| Choice menu | Programming | 21 |
| Dialogues via txt | Programming | 8 |
| Character prefabs | Programming | 27 |
| Project memory | Documentation | 60 |
| Presentation | Documentation | 8 |
| Total | | 303.5 |

Table 2.1: Time distribution by task.

| Task category | Initially planned hours | Time spent |
|--------------------------|--------------------------------|-------------------|
| Documentation and design | 60 | 90.5 |
| Art | 100 | 49 |
| Narrative | 20 | 23 |
| Programming | 120 | 141 |
| Total | 300 | 303.5 |

Table 2.2: Time deviation by task category.

The time deviation comes from the belief that programming would be a more relaxed affair given the genre of game developed. Hours of narrative and art were cut back to dedicate them to having a more polished playable result. The memory was intended to be done in fewer hours but due to various editorial changes the time investment has been lengthened.

SYSTEM ANALYSIS AND DESIGN

Contents

| | | |
|-----|--------------------------------|-----------|
| 3.1 | Requirement Analysis | 9 |
| 3.2 | System Design | 10 |
| 3.3 | System Architecture | 16 |
| 3.4 | Interface Design | 16 |

This chapter presents the requirements analysis, design and architecture of the proposed work, as well as, where appropriate, its interface design.

3.1 Requirement Analysis

First of all, the game consists of an initial menu, in which the player can choose between three buttons: Play, which loads the introductory scene of the story; Scene Selector allows the player to choose which act of the story to play; and finally, Exit allows the player to exit the game to their desktop.

Inside the game, the player can advance the story, talking with other characters, by using the Space key. In case a choice is found, a list of options will appear from which the player can click on only one. On the same gameplay screen, two clickable buttons appear above the dialogue area, one to return to the main menu called "Menu", and one that brings up a floating menu to display the player's skill bonuses, called "Stats".

3.1.1 Functional Requirements

Once the functions of the game have been introduced, the functional requirements are as follows:

- **R1**: the player can start the game.
- **R2**: the player can select an act to play.
- **R3**: the player can quit the game.
- **R4**: the player can advance the story.
- **R5**: the player can choose an option.
- **R6**: the player can return to the main menu.
- **R7**: the player can view their skill bonuses.

3.1.2 Non-functional Requirements

Non-functional requirements impose conditions on the design or implementation.

- R8: The game is playable on PC.
- R9: The UI must be simple.
- R10: The character dialogues must be short and easy to read.
- R11: The backgrounds must not attract more attention than characters.
- R12: RPG mechanics must be simple

3.2 System Design

| | |
|-----------------------|---|
| Requirement | R1 |
| Actor: | Player |
| Description: | The player can start the game by pressing <i>Play</i> |
| Preconditions: | <ul style="list-style-type: none"> 1. The player must be in the main menu. |
| Normal Sequence: | <ul style="list-style-type: none"> 1. The player presses the button <i>Play</i>. 2. The game loads the story's introduction.[?] |
| Alternative Sequence: | None |

Table 3.1: Functional requirement «Play»

| | |
|-----------------------|--|
| Requirement | R2 |
| Actor: | Player |
| Description: | The player can select an act of the story by pressing <i>Scene Selector</i> |
| Preconditions: | <ul style="list-style-type: none"> 1. The player must be in the main menu. |
| Normal Sequence: | <ul style="list-style-type: none"> 1. The player presses the button <i>Scene Selector</i>. 2. The main menu changes to the scene selection menu. 3. The player can now select between the four available scenes by pressing the corresponding button to each scene, or return to the main menu by pressing the <i>Main Menu</i> button. |
| Alternative Sequence: | None |

Table 3.2: Functional requirement «Scene selector»

| | |
|-----------------------|--|
| Requirement | R3 |
| Actor: | Player |
| Description: | The player can quit the game by pressing <i>Scene Selector</i> |
| Preconditions: | <ul style="list-style-type: none"> 1. The player must be in the main menu. |
| Normal Sequence: | <ul style="list-style-type: none"> 1. The player presses the button <i>Quit</i>. 2. The game shuts down. |
| Alternative Sequence: | None |

Table 3.3: Functional requirement «Quit»

| | |
|-----------------------|---|
| Requirement | R4 |
| Actor: | Player |
| Description: | The player can advance the dialogue using the <i>Space</i> key in the keyboard. |
| Preconditions: | <ul style="list-style-type: none"> 1. The player must be in a conversation in-game. 2. The choose menu is not on screen. |
| Normal Sequence: | <ul style="list-style-type: none"> 1. The player presses the <i>Space</i> key in the keyboard. 2. The game shows the next line of dialogue. |
| Alternative Sequence: | <ul style="list-style-type: none"> 1. The game cannot advance because the choose menu is waiting for the player to make a choice. |

Table 3.4: Functional requirement «Advance story»

| | |
|-----------------------|---|
| Requirement | R5 |
| Actor: | Player |
| Description: | The player can choose an option and new dialogue appears depending on the choice made. |
| Preconditions: | <ul style="list-style-type: none"> • 1. The player must be in a conversation in-game. • 2. Any character is asking the player for a choice. • 3. The choice menu is shown on screen. |
| Normal Sequence: | <ul style="list-style-type: none"> • 1. The player clicks an option on the menu. • 2. The chapter makes some action depending on the choice made. |
| Alternative Sequence: | None |

Table 3.5: Functional requirement «Make a choice»

| | |
|-----------------------|---|
| Requirement | R6 |
| Actor: | Player |
| Description: | The player can return to the main menu by pressing the <i>Menu</i> button. |
| Preconditions: | <ul style="list-style-type: none"> • 1. The player must be in a conversation in-game. |
| Normal Sequence: | <ul style="list-style-type: none"> • 1. The player clicks on the <i>Menu</i> button. • 2. The game loads the main menu. |
| Alternative Sequence: | None |

Table 3.6: Functional requirement «Menu Button»

| | |
|-----------------------|--|
| Requirement | R7 |
| Actor: | Player |
| Description: | The player can visualize their skill bonuses by pressing the <i>Stats</i> button. |
| Preconditions: | <ul style="list-style-type: none">• 1. The player must be in a conversation in-game. |
| Normal Sequence: | <ul style="list-style-type: none">• 1. The player clicks on the <i>Stats</i> button.• 2. A pop-up menu shows the bonuses according to your ability. |
| Alternative Sequence: | <ul style="list-style-type: none">• 1. The player clicks on the <i>Stats</i> button when the pop-up menu is being shown.• 2. A pop-up menu hides. |

Table 3.7: Functional requirement «Stats Button»

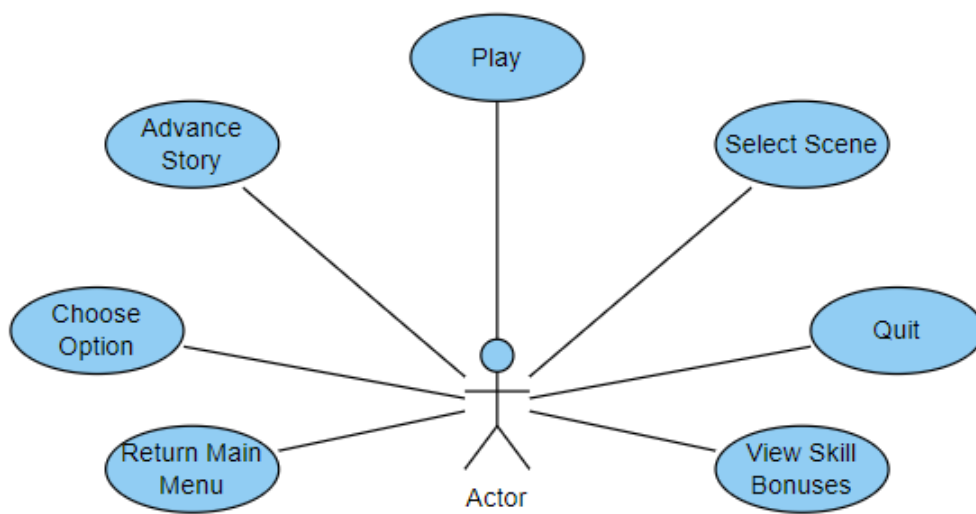


Figure 3.1: Case use diagram (made with Virtual Paradigm)

3.3 System Architecture

The minimum system requirements to play this project are:

- Operating system: Windows 7 (SP1+).
- CPU: X64 architecture with SSE2 instruction set support.
- Graphics: DX10, DX11, and DX12-capable GPUs.

3.4 Interface Design

The interface is one of the most important elements of Corinthian, accompanying the player throughout the story.

First, a simple menu with three buttons: "Play", "Scene Selector" and "Exit". "Play" loads the introduction, "Scene Selector" allows the player to choose which part they want to play and "Exit" closes the game. See Figure 3.2.

Throughout the gameplay, the interface consists of the following elements enclosed in a box: the name of the speaking character (when there is no name, the narrator speaks, see Figure 3.4), the text that is spoken, and a choice menu that is activated whenever needed. Apart from that, the player can access the main menu whenever he wants with the correspondingly named button, and a hidden menu to know the bonuses in the skills for their rolls. See Figure 3.3 [3]

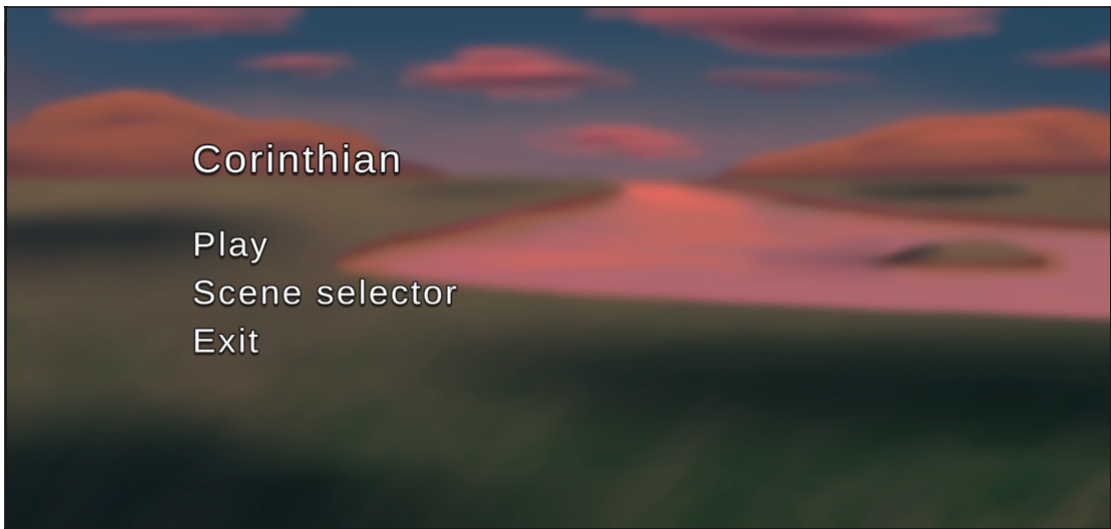


Figure 3.2: Corinthian main menu.

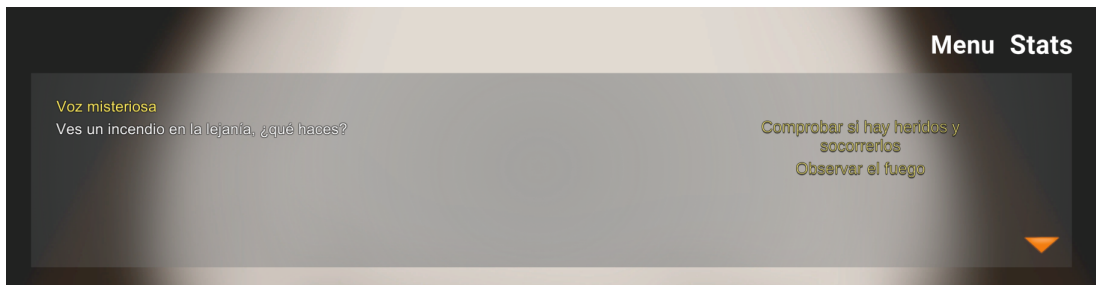


Figure 3.3: Gameplay UI with choice menu display.



Figure 3.4: Gameplay UI while narrator talks.

GAME DESIGN DOCUMENT

Contents

| | | |
|-----|----------------------------------|-----------|
| 4.1 | Game Overview | 19 |
| 4.2 | Game Mechanics | 20 |
| 4.3 | Story setting | 20 |
| 4.4 | Character design | 21 |
| 4.5 | Art Direction | 25 |
| 4.6 | Additional information | 28 |

4.1 Game Overview

The player will embody a federal agent who arrives in a village far from all the cities of the kingdom, Corinthian, where a murder has taken place and all the inhabitants are suspects. The game world is set in a classic medieval fantasy world. The player will act as an investigator and unravel the mystery behind a murder.

4.1.1 Game Genre

Corinthian is a visual novel, in which the player progresses through the game through conversations. it has a branching linear narrative, like Doki Doki Literature Club[12] or VA-11 Hall-A: Cyberpunk Bartender Action[6].

4.1.2 Target Audience

The game to be developed is aimed to be playable on PC, for an audience in a range of 16-30 years old.

4.2 Game Mechanics

The main mechanic of the game is to be able to choose what to do at different times in the gameplay. For example, if you need to know something about a character at a particular point in the story, the game will give you the option of being able to interrogate them with intimidation, or to ignore the situation and continue with the interrogations. This mechanic is based on Dungeons and Dragons system[9] and Disco Elysium[24].

In the case of choosing an action associated with a player's skill, the game will make a skill roll. This roll consists of a simulated roll of a 20-sided die with the sum of the bonus in the skill required for the action. The number obtained is compared to the numerical difficulty of the action, and if it exceeds it, the conversation continues along the positive route, and otherwise along the negative route.

Skill bonuses start at +0, so in the introduction, a series of questions are asked that modify them, positively or negatively.

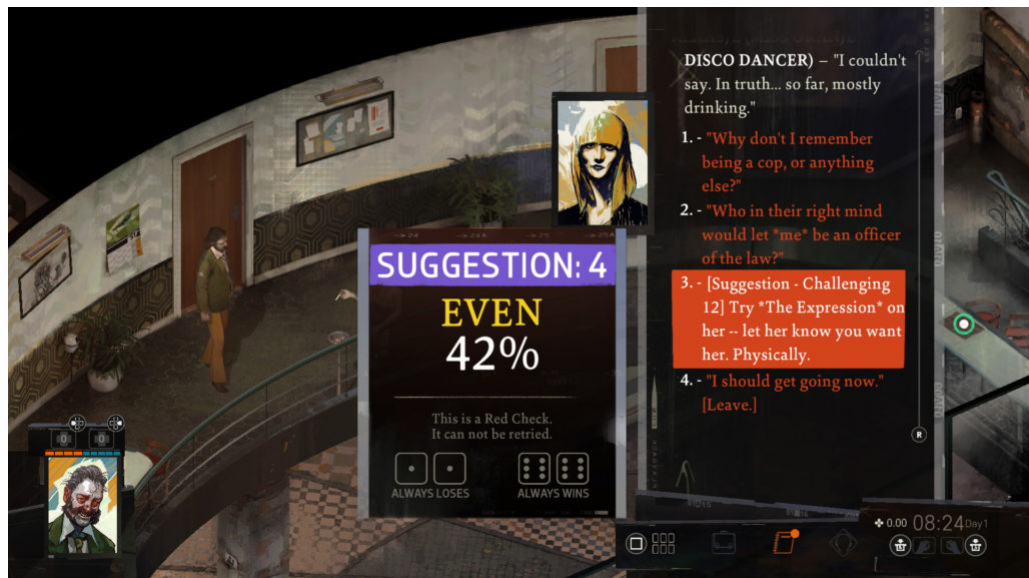


Figure 4.1: Disco Elysium (ZA/UM, 2019) conversation example.

4.3 Story setting

The story is divided into three acts: introduction, questions and answers.

First, the arrival to the crime scene and presentation of the main character's companion.

Second, the questions arc. This part is about gathering information from the villagers, establishing context to the village situation and relationships between the characters.

Finally, the answers arc. At this point, the main character will begin to understand the actions of every villager and, at the end, resolve the mystery of the murder.

The story follows a linear course, but if skill rolls are successful, new information from the story may become available. The flows of the story can be seen in the diagram.

4.4 Character design

These are the characters you can find in Corinthian:

Chai



Figure 4.2: Chai

Chief captain of military police. She is diligent and formal, she wants the work to be done properly and efficiently.

Ludovico



Figure 4.3: Ludovico

Corinthian nobleman, heavy-handed governor. He is reserved, with a forceful personality. Evil tongues say he is a powerful sorcerer but he does not show it.

His version of events: in self-defence, he killed the assailant of the mansion with a powerful magical blast that charred his face and hands; he does not remember anything because of the shock it caused him.

Sima



Figure 4.4: Sima

Ludovico's adviser. He is standoffish and classist, but attentive to what is going on in Corinthian. He is very concerned about the nobleman's condition, and it is even

rumoured that they are lovers.

His version of the events: at the time of the event, he was writing a correspondence requesting military reinforcements in view of the popular uprisings. He heard a loud bang and did not hesitate to go to check on Ludovico. He found the corpse already in an unrecognisable state.

Daran



Figure 4.5: Daran

Professional thief acting in the shadows. He is a melancholic man, grieving for an unrequited love. His whereabouts are unknown, and his testimony could be key given that he has eyes everywhere.

Almiera

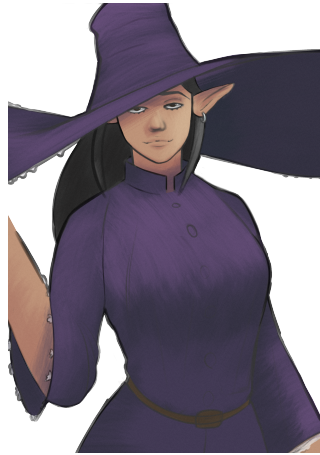


Figure 4.6: Almiera

By day, barmaid at El Muflón Mudo, by night, powerful witch. Her tavern is full of thugs, resting soldiers and work-weary peasants. Direct and seductive, she always gets what she wants.

Her version of the events: she was closing the tavern on the night of the incident. Asked about Daran, he says he has not heard from him since a day before the events.

Keres

Security officer at the noble family's palace. Is it human or machine? Nobody knows. All we know is that he is quiet but very serviceable.



Figure 4.7: Keres

Their version of the events: they went to the loud sound coming from inside the mansion, but by the time they arrived, it was all over; they say that as the man was safe, they didn't think much of it.

Sheran



Figure 4.8: Sheran

Leader of the peasants' union, she is very unhappy about the increase in working hours and strongly opposes the leadership of the nobility. She is angry but concerned about the situation of her people.

Her version of the events: opposes the investigation because of its rejection of the military police; her alibi says she was at the union house, organizing people for protests the next day.

4.5 Art Direction

The art of the game has been designed in these three sections:

- **UI:** Simple design, not tiring to the eye. It has a dark semi-transparent box to show the characters and the background behind them, but with a differentiated space on the screen to read the text. See Fig. 4.9.
- **Characters:** Semi-realistic proportions, keeping an animated style. Notorious dark line in contour of the character. Rendered shapes, no cell shading. See Fig 4.11 and ??.
- **Background:** Photograph diffused and with less saturated colours so that they do not attract attention, only the feeling of being in the place. See Figure 4.12.

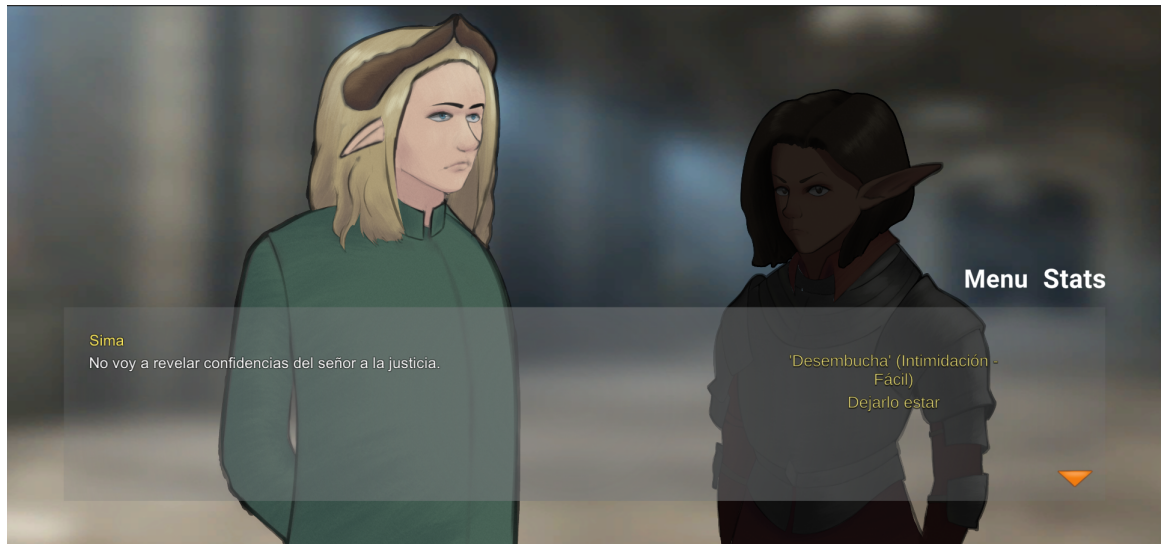


Figure 4.9: In-game UI example



Figure 4.10: Character styling.



Figure 4.11: Character styling.



Figure 4.12: In-game background example

4.6 Additional information

4.6.1 Related subjects

- VJ1218 – Narrativa hipermedia y análisis de videojuegos
- VJ1223 - Arte del Videojuego
- VJ1207 – Cultura visual y medios de masas
- VJ1233 – Modelos audiovisuales contemporáneos

WORK DEVELOPMENT AND RESULTS

Contents

| | | |
|-----|----------------------------|-----------|
| 5.1 | Work Development | 29 |
| 5.2 | Results | 39 |

5.1 Work Development

5.1.1 Code Development

Corinthian consists of five scenes: main menu, introduction, presentation of the case, questions and answers. As the main menu functioning is presented in section 3.4. Thus, the code explained below corresponds to the result of the entire gameplay.

Player Class

The Player class contains four variables, which represent the bonuses of each player skill. Skill bonuses are calculated by $(x - 10)/2$, being x the skill quantification. The bonus may be negative. The value of each skill can be accessed and modified. In the same class, the saving throws are made. These rolls consist of class functions, which generate a random number between 1 and 20 inclusive, simulating a natural 20-sided die, to which the bonus of the corresponding skill is applied, and returns the result.

Chapter Management

Chapter management focuses on reading text files that correspond to what is taught in the game, be it character dialogue, a choice or a background. In Fig. 5.1 it can be seen

an example of text file.

The chapter being read and each line is stored in an array and read from beginning to end. Each time the space is pressed on the keyboard, the chapter is advanced one line in the reading of the chapter.

The line read is checked, if it is not a choice, the `readLine()` function is called, which discerns whether there is a dialogue with function or just a function. To do this, it separates the written line by a special character, in this case ":" has been used. If there is dialogue, the first and second elements, corresponding to the name of the character who intervenes and what he/she says, are passed to the `itIsDialogue()` function. This function checks whether it is the same character speaking with respect to the previous intervention, represented by the special character "+". In any case, it updates the last speaking character "lastCharacterTalking" and calls the `say()` function of the dialogue system. The following section will expand on what this function does. Also, `itIsDialogue()` works in conjunction with the character sprite system, which will also be expanded in the subsection 5.1.1.

If an action is encountered, the corresponding chapter line argument is passed to the `itIsAction()` function. This function reads which type of function is to be executed. To do this, it separates the name of the desired function from the line using the special character "(" and ")", so that arguments can be extracted if necessary. By means of a switch, we read the type of function required, and the required function is called. The actions that can be executed in Corinthian are the following:

- ***setBackground:*** changes the background of the game to the one required, passing its name as an argument.
- ***Load:*** loads the chapter that is passed as an argument. For this function, it is needed to make sure that no chapter is running, so a check is made when loading a chapter to avoid having mixed data.[17]
- ***saveThrow:*** makes a skill roll from the player's sheet, uses numerical levels of difficulty, and based on whether it passes or fails, loads one chapter or another.
- ***changeAbilityScore:*** modifies a player's ability in his class.
- ***nextScene:*** changes scene in Unity.
- ***closeCharacter:*** used when it is known that a character in game is not gonna talk anymore or when another one is going to replace them in the same position of the screen. It calls `closeCharacter` in the character handling system.

In case it is an election, the coroutine assigned to work with it will start. First, two lists are created: one for the choices to be displayed on the corresponding buttons, and the actions for each choice. By means of a loop, the whole structure is read. The choice event starts ends with curly braces and on each line there will be a choice followed immediately by its assigned action. When all the choices have been read, the player is

expected to choose one. In the subsection 5.1.1 we will expand on how the menu works and how the texts are displayed and the choices are remembered.

Once the player chooses an available option, the corresponding action is executed, be it an extra line of dialogue such as a new chapter load or a save throw. In Fig. 5.2 and 5.3 is shown a skill check success and failure, respectively. In both images a new dialogue file is loaded corresponding with the narrative stream.

There is a chapter manager for each scene/act so that when choosing one or the other, you start with the required chapter.[15]

```

1 |
2 | Chai:Consejero Sima, ¿podemos hacerle unas preguntas acerca de lo sucedido?:
3 | Sima:Por supuesto, si mi testimonio facilita la investigación, lo haré encantado.:
4 | Chai:Perfecto, ¿dónde estaba en el momento del altercado?:
5 | Sima:Me encontraba en mis aposentos, escribiendo cartas para pedir refuerzos para defender nuestra propiedad del sindicato.:
6 | Chai:¿Oyó algo fuera de la estancia?:
7 | Sima:No, hasta el momento en el que se escuchó el estallido no había escuchado nada en absoluto.:
8 | Chai:¿Sabe algo de Ludovico? ¿Qué hacía en esos momentos?:
9 | Sima:Tenía conversaciones con sus capataces en el despacho del suceso.:
10 | +:Al parecer, al quedarse solo pasó todo.:
11 | Chai:¿Sabe el nombre de alguno de los que estaban allí con él?:
12 | Sima:No, Ludovico mantiene separados los trabajos del campo y las contrataciones de mí, yo solo me encargo de su vida personal.:
13 | Chai:Entonces, ¿podemos hablar con él de este asunto?:
14 | Sima:Dudo que podrías, ahora mismo no se encuentra en condiciones psicológicas para hablar.:
15 | Chai:Bueno...:
16 | narrador:Tienes la mosca detrás de la oreja.:
17 | +:Preguntas acerca de algún contacto privado del noble regente.:
18 | Sima:No voy a revelar confidencias del señor a la justicia.:
19 | choice
20 | {
21 |     "Desembucha' (Intimidación - Fácil)"
22 |     saveThrow(intimidacion,10,acto2_2a,acto2_2b)
23 |     "Dejarlo estar"
24 |     Sima:Los federales os creéis mejores que nadie.:
25 | }
26 | Chai:Muchas gracias por su declaración, señor Sima.:
27 | Sima:Espero haber ayudado.:
28 | narrador:Ambos salís al exterior, en búsqueda de los siguientes sospechosos.:stopTalking(Sima)
29 | Load(acto2_3)
30 |

```

Figure 5.1: Dialogue text file example.

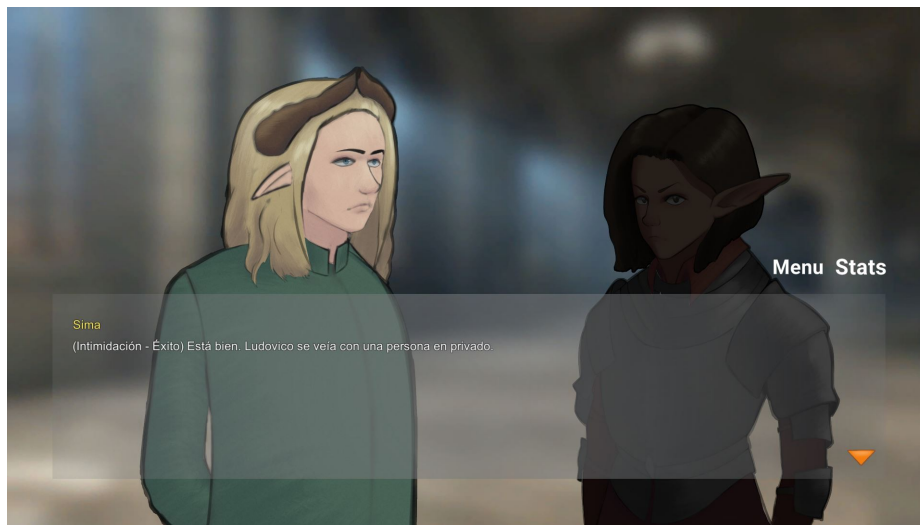


Figure 5.2: The player success a skill check.

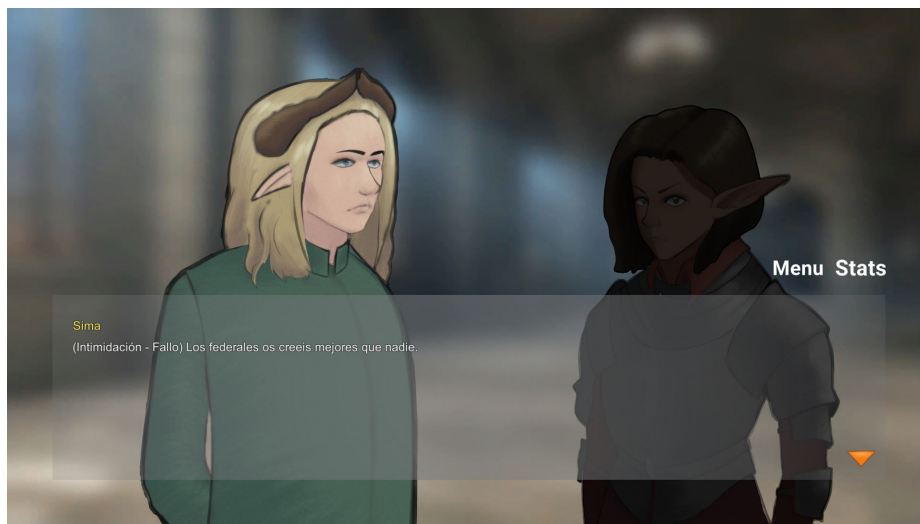


Figure 5.3: The player fails a skill check.

Dialogue System

The dialogue system is in charge of teaching what is said in the game. Unity takes the elements of the dialogue space, in this case, the name of the character speaking and the text they say. See Fig. 5.4 for a broken down elements in the dialogue class.

Dialog treatment consist of coroutines. First, the HUD background is activated, the dialog text is initialized to an empty string and it checks who is speaking, the continue button is activated, and waits for player interaction.

To check the name of the speaker, the `WhoTalking()` function is called, which omits the special name of the narrator in case the speaker speaks.

When everything is ready, to give the feeling that the text is being typed as it is read instead of loading the entire text box, a loop will write the dialog that is displayed on the screen until it matches the entire dialog . With the full text on the screen, the continue button appears and the player can hit space. Although the player can continue with the next intervention at any time, it was thought to add the continue button to let them know that nothing more is going to be written on the screen and it is a good time to hit next.

If text ends, dialogue background hides.[2]

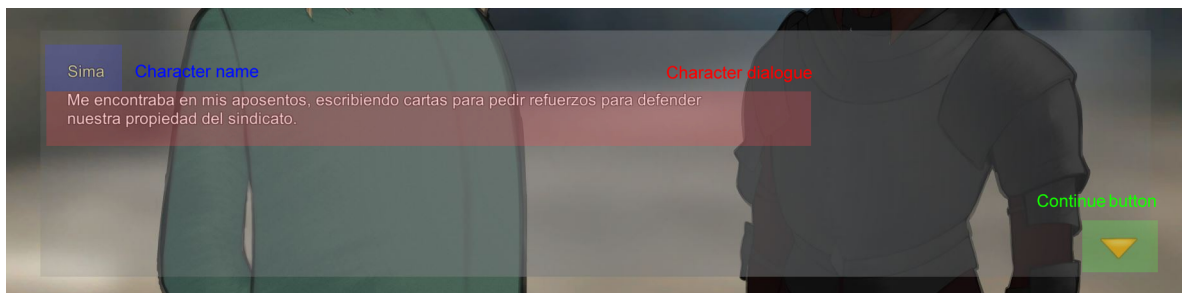


Figure 5.4: Dialogue system broken down.

Character Handling

The character system is based on the use of lists, dictionaries and the characterClass class. [14]

This class contains the name of the character in question, the place where it will be rendered and if it is activated in the hierarchy. The rendering image is placed in the rendering location.

Back in the characterManager, it consists of four functions:

`getCharacter()`: if the character exists in the dictionary, it will return its instance of characterClass saved in the character list at the assigned save index; otherwise, the `createCharacter()` function is called.

`createCharacter()`: Creates a new instance of the characterClass class, adds it to the dictionary next to the maximum index of the characterClass list, and then adds the instance to the list. Returns the instance as the previous function does when finding the character.

`showCharacter()`: searches the character with name passed as argument, if found, modifies the color parameters of the character's sprite so that it lights up. To make sure that the character is being displayed, the opacity is set to maximum; it makes sense because of the `closeCharacter()` function explained below

`hideCharacter()`: searches the character with name passed as argument, if it finds it, it modifies the color parameters of the character sprite so that it becomes darker.

`closeCharacter()`: lowers the opacity to 0 for the character image that is passed as a reference.

See how the characters appear in the game with the last two functions in Fig. 5.5. Alternatively, when a character is hidden, it is seen as in Fig. 5.6.

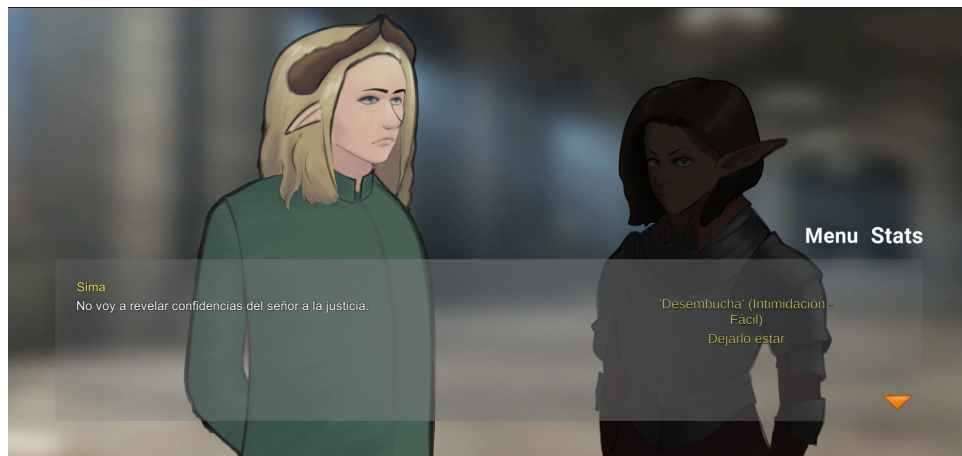


Figure 5.5: The character talking is highlighted, the silent other remains darkened.

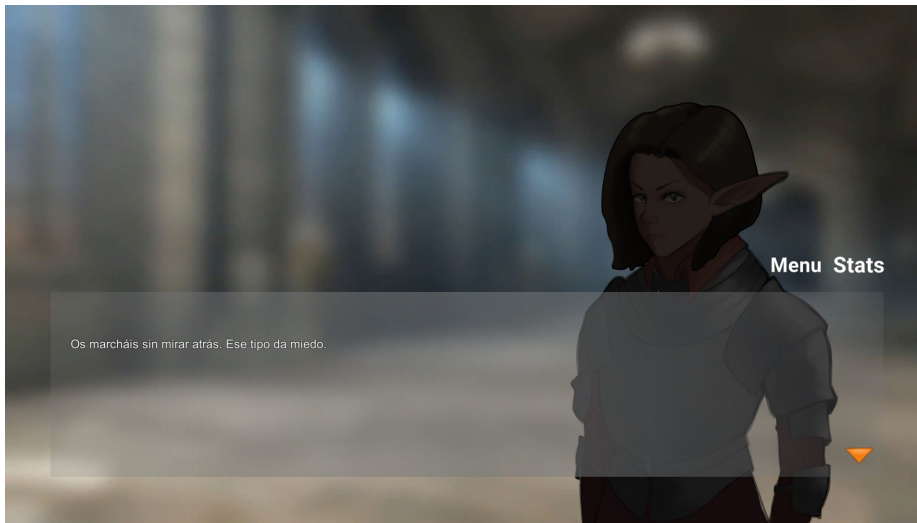


Figure 5.6: The conversation ends, the character in the left is gone.

Choice Menu

The choice menu consists of a `VerticalLayoutGroup` and a list of objects of class `choiceButton`. Regardless of the number of buttons required to be displayed, they will be stacked in the group. In Fig. 5.7 and 5.8 can be seen how the buttons are stacked depending on the number of elections.

The choice display coroutine instantiate `choiceButton` class prefabs, then modifies the text that appears on the button.

The `Choice` class stores the text of the choice, the index of the choice, and whether it was chosen using a boolean. From Unity, the button calls the `makeChoice()` function when clicked, which overrides the last choice made and closes the choice display routine having saved the index of the choice that the player has made.[16]

5.1.2 Character Design

An attempt has been made for each character to have a distinctive silhouette, which differentiates it from other characters so that you can quickly grasp who the character is without having to pay much attention to every detail. See Fig 5.9.

5.1.3 Narrative Design

A three-part structure was the most interesting option for a visual novel. Taking into account the way other visual novels such as *Umineko no Naku Koro ni*[1] have, it was decided to modify the classic three-part structure for one in two marked parts, a question arc, where all the characters and their situations are introduced, and another arc of answers, in which everything that is introduced in the first argumentative arc is resolved.

This form best serves the story of Corinthian to tell the mystery of the town without bothering the player with three classic acts (introduction, middle and denouement). See Figures 5.10, 5.11 and 5.12.

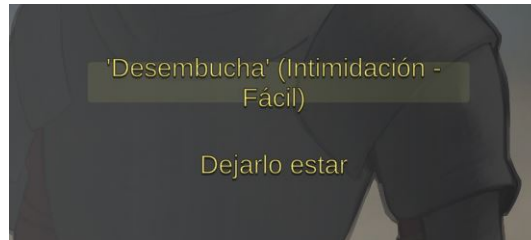


Figure 5.7: Two items choice.

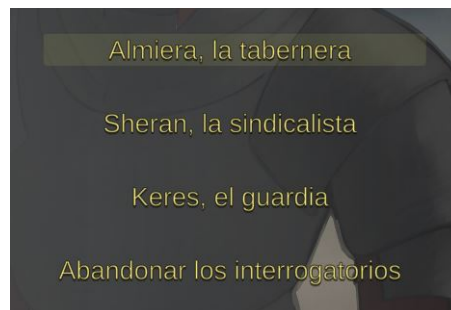


Figure 5.8: Two items choice.



Figure 5.9: Character's silhouettes example

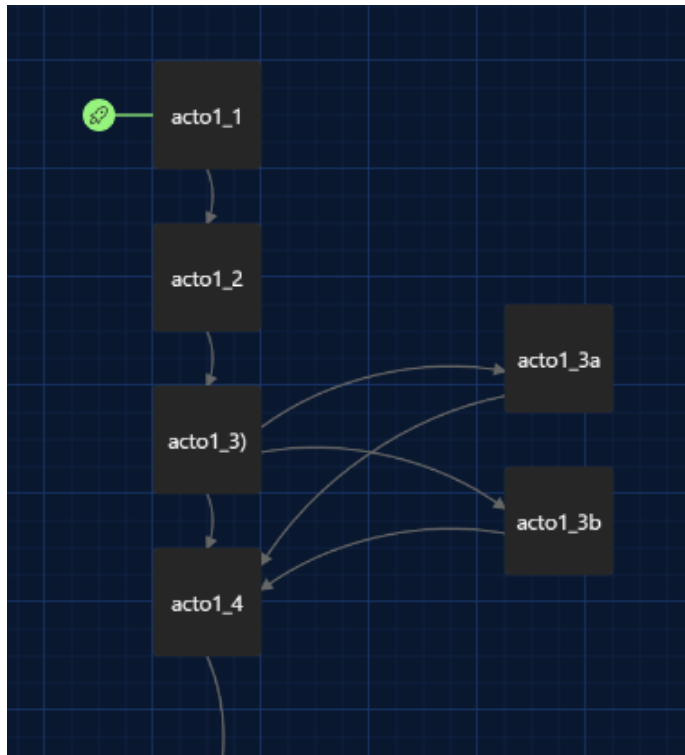


Figure 5.10: Corinthian: First few questions (made with Twine)

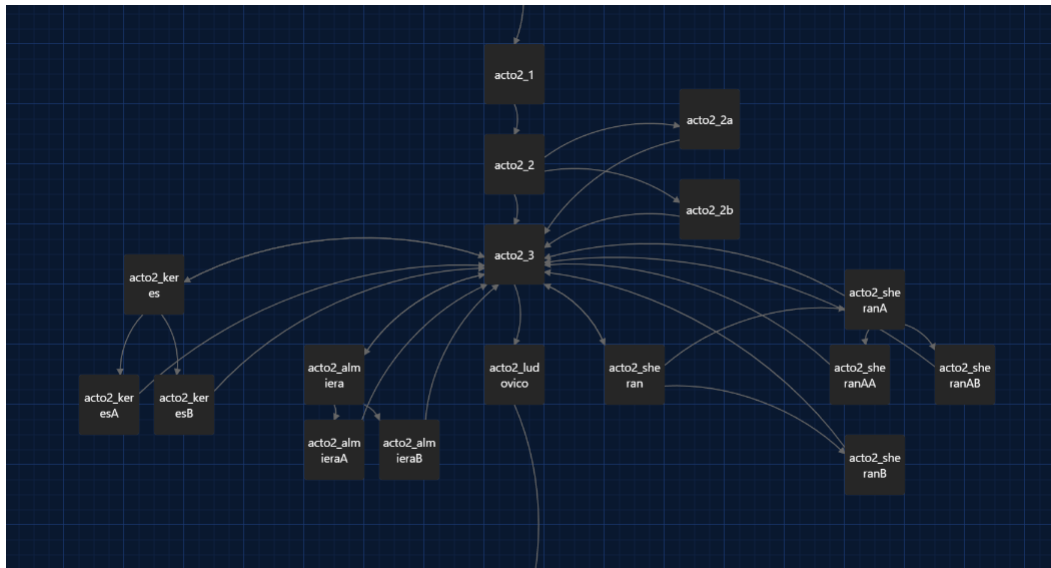


Figure 5.11: Corinthian: Questions arc (made with Twine)

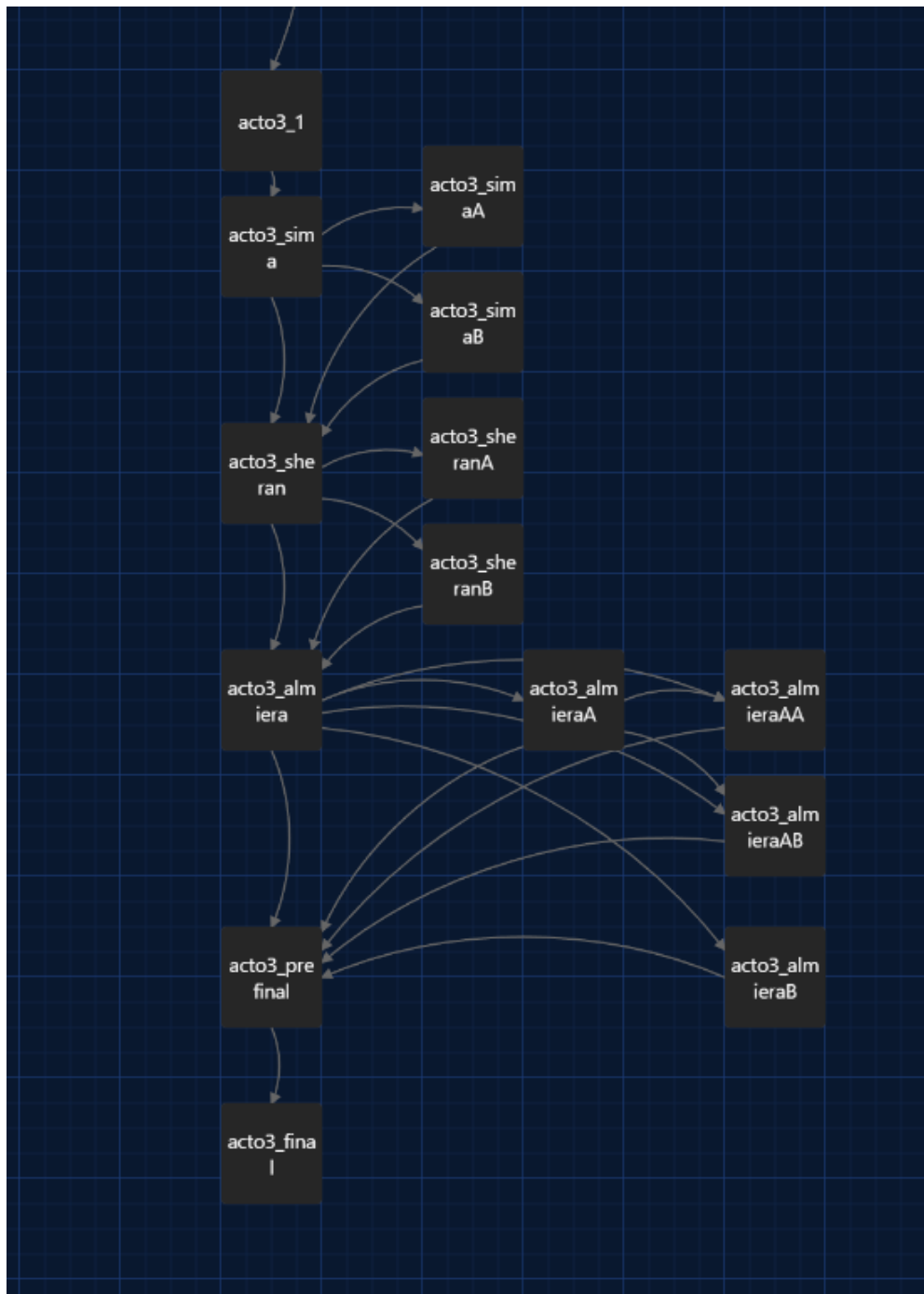


Figure 5.12: Corinthian: Answers arc (made with Twine)

5.2 Results

In conclusion, the objectives proposed in the first chapter have been satisfactorily met, giving a good game result. Mainly, the characters are flawless looking. The dialogue and chapter management system works perfectly. In case the story was expanded or make it more literary, the game could be expanded with a save system as a bookmark.

CONCLUSIONS AND FUTURE WORK

Contents

| | | |
|-----|-----------------------|----|
| 6.1 | Conclusions | 41 |
| 6.2 | Future work | 41 |

In this chapter, the conclusions of the work, as well as its future extensions are shown.

6.1 Conclusions

In conclusion, my experience in the video game design and development degree has been invaluable to my growth as a designer. This project has allowed me to see the potential of video games as a medium of expression and how it can be a powerful platform for storytelling and adapt classic games to a new platform. The freedom I had on this project allowed me to explore new ideas and creative approaches, which elevated me as a creator and left me with a new appreciation for the potential of video games in terms of means of expression.

I've also managed to get ahead with the programming part of the visual novel, educating myself as much as possible on how to translate my ideas into programming language. Also, having a space to tell my story and develop the character design has been a very good experience for future projects that I will take forward.

6.2 Future work

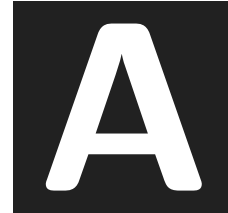
Corinthian can continue to improve. During the course of the project, I thought about those players who might want to leave the story for another time if it gets too long.

Employing a save system is the next goal, as well as expanding the skill system and integrating skills into the story. Furthermore, I would like to add music and sound effects, as well as character expressions. I will continue to work to improve the project and polish it as much as possible.

BIBLIOGRAPHY

- [1] 07th Expansion. Umineko when they cry. <https://en.wikipedia.org/wiki/Umineko_When_They_Cry>. *Accessed* : 2023 – 01 – 30.
- [2] Brackeys. How to make a dialogue system in unity. <https://www.youtube.com/watch?v=_nRzoTzeyxU>. *Accessed* : 2023 – 04 – 20.
- [3] Brackeys. Start menu in unity. <https://www.youtube.com/watch?v=zc8ac_qUXQY>. *Accessed* : 2023 – 04 – 15.
- [4] Colaboradores de Wikipedia. David lynch. <https://es.wikipedia.org/wiki/David_Lynch>. *Accessed* : 2023 – 01 – 30.
- [5] Colaboradores de Wikipedia. Rpg maker. <https://es.wikipedia.org/wiki/RPG_Maker>. *Accessed* : 2023 – 01 – 30.
- [6] Sukeban Games. Va-11 hall-a: Cyberpunk bartender action. <https://es.wikipedia.org/wiki/VA-11_HALL – A>. *Accessed* : 2023 – 01 – 30.
- [7] Github. Github. <https://github.com>. *Accessed*: 2023-02-05.
- [8] Microsoft. Visual studio code. <https://code.visualstudio.com>. *Accessed*: 2023-02-05.
- [9] Wizards of the Coast. Dungeons and dragons. <https://dnd.wizards.com/es>. *Accessed*: 2023-01-30.
- [10] XPPen Tienda Oficial. Star g960s star g960s plus tableta digitalizadora para profesores. <https://www.storexppen.es/buy/star-g960s-star-g960s-plus.html?gad=1&gclid=Cj0KCQjwr82iBhCuARIsAO0EAZwrVsyRTJgTE2s1-zCGT41xqpX7RWADU6Tps0z7MoC4O_As7PbqeEaArW7EALw_wcB>. *Accessed* : 2023 – 05 – 02.
- [11] Overleaf. Overleaf. <https://es.overleaf.com>. *Accessed*: 2023-01-30.
- [12] Team Salvato. Doki doki literature club. <https://ddlc.moe>. *Accessed*: 2023-01-30.
- [13] Scrum.org. What is scrum?. <https://www.scrum.org/learning-series/what-is-scrum/what-is-scrum>. *Accessed*: 2023-01-30.

-
- [14] Stellar Studio. Let's make a visual novel 05 - character management. <https://www.youtube.com/watch?v=D634iEi3n7E>. Accessed: 2023-04-22.
- [15] Stellar Studio. Let's make a visual novel 10 - text architect. <https://www.youtube.com/watch?v=M6ZDBOtUinw>. Accessed: 2023-04-22.
- [16] Stellar Studio. Let's make a visual novel 21 - choice screen. https://www.youtube.com/watch?v=Pfp_VSavD4. Accessed : 2023 - 04 - 22.
- [17] Stellar Studio. Let's make a visual novel 22 - dynamic story paths. <https://www.youtube.com/watch?v=d08oAEs3IC4>. Accessed: 2023-04-22.
- [18] CLIP STUDIO.NET. Comprar clip studio paint. <https://www.clipstudio.net/es/purchase/>. Accessed: 2023-05-02.
- [19] Talent.com. Salario para artista en españa - salario medio. <https://es.talent.com/salary?job=artista>. Accessed: 2023-05-02.
- [20] Talent.com. Salario para programador en españa - salario medio. <https://es.talent.com/salary?job=programador>. Accessed: 2023-05-02.
- [21] Unity Technologies. Unity. <https://unity.com/es>. Accessed: 2023-02-05.
- [22] Trello. Trello. <https://trello.com/es>. Accessed: 2023-01-30.
- [23] Twine. Twine. <https://twinery.org>. Accessed: 2023-02-10.
- [24] ZA/UM. Disco elysium - the final cut. https://store.steampowered.com/app/632470/DiscoElysium_TheFinalCut



SOURCE CODE AND GAME BUILD

First, through this link it can be viewed the developer's diary where the time is broken down by tasks: <https://docs.google.com/spreadsheets/d/1m4c9Rm7WHVdsNTDhpVZvEtvSYshXHeCH0ShmuEcHyh8/edit?usp=sharing>

Due to the length of the source code, here is a link to the Github repository: <https://github.com/al385752/Corinthian>

A build of the game can be downloaded at the following link: https://drive.google.com/file/d/1tp6NzaPeitty07pXBGLG0_vBignGYDFf/view?usp=sharing

From a previous playtesting, here are the results of the associated survey: <https://docs.google.com/spreadsheets/d/1C1QGsLTwzjjQynG4AnmSldWMRxFsQi5ltw8sAsFqu-k/edit?usp=sharing>

Finally, a commented gameplay of Corinthian is available at the following link: <https://youtu.be/r5vbhkYzgCw>

