
EDUCATIONAL VIDEO GAME AND TOOL TO LEARN AND PRACTICE DIFFERENT WRITING SYSTEMS



Águeda Gómez Cambronero

Advisor: Dr. Raúl Montoliu Colás

Universitat Jaume I

This dissertation is submitted for the bachelor's degree of
Video Game Design and Development.

ABSTRACT

This Degree's Final Project consist of the development of an educational video game to learn and practice handwriting in different languages. The video game embodies a tool which lets the user to add new symbols and to edit the game's puzzles. The purpose of the game is to make an effective video game which is educational and amusing at the same time. It is a Point and Click game made with Unity3D for PC and touchscreen devices. The video game simulates a Escape Room where the player is locked and must escape using his logic, wit and his handwriting skill.

Keywords: Educational, Serious Game, Handwriting, Languages, Point And Click, Escape Room.

INDEX

| | |
|--|----|
| LIST OF FIGURES..... | 5 |
| LIST OF TABLES | 6 |
| 1 TECHNICAL PROPOSAL | 7 |
| 1.1 INTRODUCTION AND MOTIVATION OF THE PROJECT | 7 |
| 1.2 RELATED SUBJECTS | 8 |
| 1.3 OBJECTIVES | 8 |
| 1.4 PROJECT SCHEDULE | 9 |
| 1.4.1 PHASE DOCUMENTATION | 9 |
| 1.4.2 PHASE GAME DEVELOPMENT..... | 9 |
| 1.4.3 PHASE GAME DESING..... | 10 |
| 1.4.4 PHASE GAME ART | 10 |
| 1.4.5 PHASE GAME EVALUATION | 11 |
| 1.5 TOOLS..... | 12 |
| 1.5.1 DOCUMENTS..... | 12 |
| 1.5.2 PROGRAMING..... | 12 |
| 1.5.3 2D AND UI DESIGN | 12 |
| 1.5.4 3D MODELING AND TEXTURING..... | 12 |
| 1.5.5 VERSION CONTROL SYSTEM | 12 |
| 1.6 EXPECTED RESULTS..... | 13 |
| 2 LITERATURE REVIEW | 14 |
| 3 DESIGN..... | 16 |
| 3.1 GAME CONCEPT | 16 |
| 3.2 HISTORY | 16 |
| 3.3 CONTROLS..... | 17 |
| 3.4 GAME FLOW | 18 |
| 3.5 GAME MECHANICS | 19 |
| 3.6 GAME ENVIRONMENT | 20 |
| 3.6.1 ENVIRONMENT DESCRIPTION | 20 |
| 3.6.2 INTERACTABLE OBJECTS..... | 20 |
| 3.6.3 RIDDLES PUZZLES | 25 |
| 4 DEVELOPMENT..... | 28 |
| 4.1 CREATION OF A FUNCTIONAL PROTOTYPE | 28 |
| 4.1.1 HANDWRITING IN UNITY..... | 29 |
| 4.1.2 FILE PERSISTENCE..... | 31 |

| | | |
|-------|--|----|
| 4.1.3 | CHARACTERS COMPARISON | 32 |
| 4.1.4 | PROTOTYPE FUNCTIONALITY..... | 34 |
| 4.2 | IMPLEMENTATION OF THE VIDEO GAME | 37 |
| 4.2.1 | SCENES DESIGN..... | 37 |
| 4.2.2 | INTERACTION BETWEEN SCENES | 38 |
| 4.2.3 | INTEGRATION OF NARRATIVE..... | 39 |
| 4.3 | BUILDING THE GAME | 39 |
| 5 | RESULTS..... | 40 |
| 6 | TESTING AND EVALUATION | 41 |
| 6.1 | VERIFICATION AND VALIDATION..... | 41 |
| 6.1.1 | VERIFICATION | 41 |
| 6.1.2 | VALIDATION..... | 41 |
| 6.2 | EFFICIENCY | 42 |
| 7 | PROJECT DEVIATIONS..... | 43 |
| 7.1 | GAME CONCEPT | 43 |
| 7.2 | PLANNING | 43 |
| 8 | CONCLUSIONS..... | 47 |
| 8.1 | FUTURE WORK | 47 |
| 8.1.1 | EXTENSION OF THE GAME..... | 48 |
| 8.1.2 | EXTENSION OF THE TOOL..... | 48 |
| 8.1.3 | FUTURE RESEARCH LINE..... | 48 |
| | REFERENCES | 49 |

LIST OF FIGURES

| | |
|---|----|
| Illustration 1 Left and right mouse click funtions | 18 |
| Illustration 2 Flowchart of Game scenes..... | 19 |
| Illustration 3 Concept art of the room..... | 20 |
| Illustration 4 Closed objects in the room | 22 |
| Illustration 5 Objects of progress in the room | 24 |
| Illustration 6 Clue in a crumpled paper | 24 |
| Illustration 7 Clue in the post-it | 25 |
| Illustration 8 Clue in a book sheet | 25 |
| Illustration 9 Logical sequence of the game | 26 |
| Illustration 10 Encryption | 27 |
| Illustration 11 Structure of the Development Phases | 29 |
| Illustration 12 Calculation of the vertices of the mesh | 30 |
| Illustration 13 Result of the handwriting tool | 31 |
| Illustration 14 Code of save data in Persistence Controller | 32 |
| Illustration 15 Main Menu of the prototype | 34 |
| Illustration 16 Enter symbol name | 35 |
| Illustration 17 Symbol drawing | 35 |
| Illustration 18 Game list | 36 |
| Illustration 19 Prototype environment | 36 |
| Illustration 20 Game Object persistence code | 38 |

LIST OF TABLES

| | |
|--|----|
| Table 1 Phase Documentation Planning | 9 |
| Table 2 Phase Game Development Planning | 10 |
| Table 3 Phase Game Design Planning | 10 |
| Table 4 Phase Game Art Planning | 11 |
| Table 5 Phase Game Evaluation Planning | 11 |
| Table 6 Summary Planning | 11 |
| Table 7 General overview of closed objects | 22 |
| Table 8 Phase Documentation - Differences in the planning | 44 |
| Table 9 Phase Game Development - Differences in the planning | 44 |
| Table 10 Phase Game Design - Differences in the planning | 44 |
| Table 11 Phase Game Art - Differences in the planning | 45 |
| Table 12 Phase Game Evaluation - Differences in the planning | 45 |
| Table 13 Summary planning - Differences in the planning | 45 |

1 TECHNICAL PROPOSAL

This chapter presents an overview of the project. The motivation of the project, a list of the principal objectives to achieve with work done, a planning of this development and the expected results are also explained in the following sections.

1.1 INTRODUCTION AND MOTIVATION OF THE PROJECT

Nowadays the use of video games like complementary material in learning is really common. This genre of video games is underrated because the gameplay and educational purpose are separated. This causes demotivation and the lack of interest in the player [1]. Moreover, many of them do not attain their educational aim insomuch as the goal design and feedback are not adequate [1]. Likewise, many of this type of video games have a plot related with techniques areas of knowledge like mathematics or physics. Very few are focusing in philology.

This project has as a main objective the development of a game to cover this lack. The video game has been designed depending on the target needs, integrating into the gameplay and plot the educational purpose and giving correct help and feedback. This way lets the player learn but always has a challenge [2]. In addition, its edition will be allowed and thus will convert in an educational tool.

The Escape Rooms [3] are the inspiration of this video game, whose challenges require the environment exploration and the use of logic and wit. This facilitates the adherence of the player to the mechanics of the game. This type of video games are a subgenre of the Point and Click games. Among its features is a first person perspective where the player interacts with the environment through clicks. In this case, the resolution of riddles and key input are based on the handwriting of different symbols which have a link with the game environment. This presents an added challenge since the player must demonstrate his handwriting skills in a new language in addition to having to relate them conceptually to the environment to decipher the riddles that are presented to him. The design of the puzzles can be edited introducing new symbols to teach and even their corresponding phonemes. Some games that have served as a reference are *Can You Escape* [4] and *Cube Escape* [5] series.

1.2 RELATED SUBJECTS

Several subjects of the degree design and development of video games are related in this project but they emphasize the following:

- VJ1235 Advanced Interaction Techniques: The objective of this subject is to explain techniques of interaction that go beyond the common ones. This aligns perfectly with the purpose of this work given that the drawing of symbols raised requires the use of this type of interfaces or alternative interpretations.
- VJ1238 Basics for the Design of Educational Video Games: The subject aims to raise awareness of the learning that players perform in every video game. It also has the purposes to provide students with a base knowledge needed to design videogames that, in addition to the usual ludic goals, will offer valuable learning opportunities for players. It is necessary that the multidisciplinary components working in a video game share a set of basic concepts about learning, teaching and development.
- VJ1222 Conceptual Design of Video Games: This subject is looking for a game design based on the balance of all the ingredients that make up a video game. This should result in more complete and differentiating products. All this based on the understanding of the wide range of existing game systems, their convergence and evolution.
- VJ1216 3D Desing: This subject pretends to teach techniques of modelling and texturizing of 3D objects and for its later insertion in videogames. In addition to studying the generation of illumination maps of the models to adapt them to the virtual scenes.

1.3 OBJECTIVES

The following objectives are those that are intended to be achieved with the realization of this project.

- Develop a video game in which it be can use and practice handwriting.
- Convert handwriting in the key piece to resolve the game's puzzles.
- Make this video game a suitable tool for anyone.

1.4 PROJECT SCHEDULE

This section describes the division of project's tasks and its estimate of the required time in hours. The totality of the project has been divided into five phases, each one containing a set of tasks and its estimated duration.

1.4.1 PHASE DOCUMENTATION

Table 1 shown all of the required documentation tasks like write the technical report and prepare the project defence presentation that includes a video for Degree's Final Project fair.

Table 1 Phase Documentation Planning

| Documentation | |
|------------------------------|--------------------------------------|
| <i>Tasks</i> | <i>Estimated duration (in hours)</i> |
| Technical Proposal | 8 |
| Technical Report | 40 |
| Video | 2 |
| Project Defense Presentation | 10 |
| Total | 60 |

1.4.2 PHASE GAME DEVELOPMENT

In this subsection, the development tasks are presented. These are divided into five blocks for simplification and are shown in *Table 2*.

Digital handwriting block contains:

- The implementation, in a convenient way, in Unity the stroke creation with mouse and touch.
- The translation of the drawing into a data structure useful for symbol recognition.

Symbol recognition block contains:

- To create data base with possible symbols.
- To train a model for stroke recognition.
- To save a new strokes in data base.

- To train a model that learn the new symbols.

Turn video game into a tool needs:

- To code the video game edition.
- To allow the introduction of new symbols in its edition.
- To implement the save new game option.

Gameplay and mechanics programming block contains:

- To implement screen flow.
- To implement the game loop incorporating the handwriting and symbols recognition.

Table 2 Phase Game Development Planning

| Game Development | |
|------------------------------------|--------------------------------------|
| <i>Tasks</i> | <i>Estimated duration (in hours)</i> |
| Digital handwriting | 20 |
| Symbol recognition | 40 |
| Turn video game into a tool | 50 |
| Gameplay and mechanics programming | 30 |
| Total | 140 |

1.4.3 PHASE GAME DESING

In this phase the game design will be elaborated (*Table 3*) that is contained in the game design document (GDD) and which will specify the game mechanics, controls, game story, game screens and game flowcharts. This document is added like a chapter in this technicar report.

1.4.4 PHASE GAME ART

Elaboration of all the visual aspects of this project like the modelling, GUI elements and all the concepts of art and images included in this technical report. In *Table 4* shown the estimated duration of this tasks.

Table 3 Phase Game Design Planning

| Game Desing | |
|--------------------------|--------------------------------------|
| <i>Tasks</i> | <i>Estimated duration (in hours)</i> |
| Environment desing | 10 |
| HUD and GUI desing | 5 |
| Puzzles and clues desing | 15 |
| Total | 30 |

Table 4 Phase Game Art Planning

| Game Art | |
|-----------------------------------|--------------------------------------|
| <i>Tasks</i> | <i>Estimated duration (in hours)</i> |
| Environment modelling | 15 |
| Interactable objects modelling | 10 |
| User Interface art | 10 |
| Concepts art | 6 |
| Game Over, menu and final screens | 9 |
| Total | 50 |

1.4.5 PHASE GAME EVALUATION

Test the project: evaluate the difficulty of the game, its intuitiveness, effectiveness in learning, the degree of fun and finding errors in the operation.

Table 5 Phase Game Evaluation Planning

| Game Evaluation | |
|---------------------------|--------------------------------------|
| <i>Tasks</i> | <i>Estimated duration (in hours)</i> |
| Validation of the project | 15 |
| Project performance | 5 |
| Total | 20 |

In Table 6 shown a resume of this planning.

Table 6 Summary Planning

| Summary table | |
|------------------|-------------------------------|
| Tasks | Estimated duration (in hours) |
| Documentation | 60 |
| Game Development | 140 |
| Game Design | 30 |
| Game Art | 50 |
| Game Evaluation | 20 |
| Total | 300 |

1.5 TOOLS

This section lists the tools that will be used during the development on this project. They are grouped according to the purpose for which they have been used.

1.5.1 DOCUMENTS

- **Microsoft Word:** word processor belonging to the Microsoft Office suite.
- **Overleaf:** online platform for scientific writing based in LaTeX.
- **Google Slides:** program for creating and editing slide show presentation.

1.5.2 PROGRAMING

- **Unity 5.6:** game engine for make 3D and 2D games.
- **Visual Studio 2015:** IDE that support C# and is liked with Unity.

1.5.3 2D AND UI DESIGN

- **Adobe Illustrator CS6:** vector graphics editor.

1.5.4 3D MODELING AND TEXTURING

- **Autodesk 3ds Max 2016:** software for 3D modeling.

1.5.5 VERSION CONTROL SYSTEM

- **Git:** open source distributed version control system.

- **GitHub:** a Git repository hosting service.

1.6 EXPECTED RESULTS

The resulting project is expected to:

- To be a video game that combines its didactic and entertainment purposes.
- To be easily scalable for future expansion.
- To be a project with the good finish.
- To be a versatile tool that supports various types of signs.
- To be a video game that can be easily adapted to a very wide target, from people who begin to learn to write new languages to the public that seeks to entertain themselves with challenges that require a different skill, passing through those that have some difficulty and need to practice that they already know.

2 LITERATURE REVIEW

The last twelve years the development of video games has grown exponentially, this is due in large measure to the reduction of the economic entry barriers entailed by access to a video game engine [6]. About the year 2006 some of the most important video game companies in the world opened their development engines at very affordable prices or even completely free [7].

This has had a significant impact on the number of areas in which video games are applied, have ceased to be a tool focused only on leisure to support fields such as medicine [8], nursing [9], psychology [10] or military training [11]. In all these fields the appearance of the video game is presented as an additional component to the existing task that provides good results. This type of video game has been defined as Serious Games [12].

One of the fields where the Serious Games have had a great impact, besides those mentioned above, has been the one of the education [13][14][15][16]. The results presented in these papers are generally positive, increasing the implication, improving the scores in the tests performed and enlightening adhesion. This is due to the combination of active pedagogy techniques and their playfulness with learning objectives that are based on knowledge or skills [17].

Specifically in language learning this potential has been analyzed by multiple authors, pointing out the opportunities presented in linguistic interaction [18], highlighting the options they present in computer-assisted language learning (CALL [19]) and emphasizing the possibilities offered by multiplayer environments (in online video games).

In spite of all these circumstances, reviewing the literature, references to the use of Serious Games in the learning of the writing are not found. There is some work that talks about the influence of the videogames in the writing of the students [20], expanding their vocabulary, but not in the technique of learning and improving the signs when writing them by hand. In relation to this gap is defined the first question that is made at

the beginning of this work: WHAT. What is this work trying to bring to the state of the art? A new way to teach writing to players.

The next question that arises is the "HOW?", having the objective defined and justified, an analysis is made on the different methodologies that have been applied in Serious Games, in [17], Patiño et al. made a review of the literature and conclude that the type of game should be aligned with the pedagogical design of the same, the number of players and the complexity of the interaction. However, a specific typology that guarantees good results is not described.

For this work the case of Escape Rooms has been considered [21], a kind of game that is being applied increasingly in the field of leisure (And generating more and more benefits [22]). The literature does not present previous work using this methodology, so it is considered that developing a Serious Game based on it and having as educational objective the learning of signs, will make an important contribution in this field.

3 DESIGN

This chapter discusses the Game Design Document (GDD) which outlines the developmental and design specifications of the game as its history, its mechanics and controls and the environment where it is developed.

3.1 GAME CONCEPT

This project is a video game of the genre Point and Click, in first person, for PC and tactile platforms. In it, the player is alone and locked in a room. His goal will be to get out of it in the shortest possible time since a countdown of three minutes has begun. To achieve this he must explore the environment and find the clues that together with the use of logic and cleverness will become the tools that will give him the keys to discover how to get out. These keys are based on different characters whose input mechanism will be to write them by hand, so the player must acquire or develop the skill of handwriting to overcome the game.

The purpose of this video game is for the player to achieve the learning and / or improvement of different characters integrated in the game mechanics. To provide it with versatility and add a replayability component, the possibility to edit the writing keys of the game and save it for future games has been implemented. In this way, it becomes a tool to be able to learn different writing systems.

3.2 HISTORY

Stunned, you open your eyes, you do not recognize what you see. A white light floods the room, a room without windows or direct contact to the outside. You do not take long to see a digital clock in which a countdown whose outcome you do not know has already begun that.

The room is flooded with papers and clippings in which there is talk of a catastrophe and any reading invites you not to leave there but you cannot find the way or the reason to survive in that place. You look for the exit door because you do not want to find out what the end of that countdown entails but a code is guarding your lock and crazily you

start looking for any clue that takes you to it. With a general look you realize that the door is not the only thing that requires a key or a code, so you set out to investigate and open all the locks to discover the mystery of why you are there and what is waiting for you outside. All the locks in the room are small machines that interpret what is written on paper so armed with a pencil you start your deep search.

After a while you find a revealing clue. The clock consumes the last minute and you can open the door. As soon as you leave the room you find some stairs that help you get out, but as you go forward some photographs and several notes reveal the external disaster from that you had hidden. The door closes cruelly behind you and at that moment you accept that the end comes and you curse the time you decided to leave that room.

The room is a punishment that has induced in your mind and it consists in provoking the anxiety and despair of being locked in a place that you can not leave. You run out of time, a time that does not trigger anything but you do not know. When you leave, you wake up in that prison from which your punishment comes and they induce you over and over again causing a hell in loop.

3.3 CONTROLS

The controls of the game for PC are:

- **Left mouse click:** Move the camera (zoom in or out of a room section), select or pick up objects and write.
- **Right mouse click:** Inspect a selected object or rotate in room ninety degrees.

For touch devices like tablets or Android mobiles:

- **Tap:** Move the camera (zoom in or out of a room section), select or pick up objects and write.

- **Touch:** Inspect a selected object or rotate in room ninety degrees.

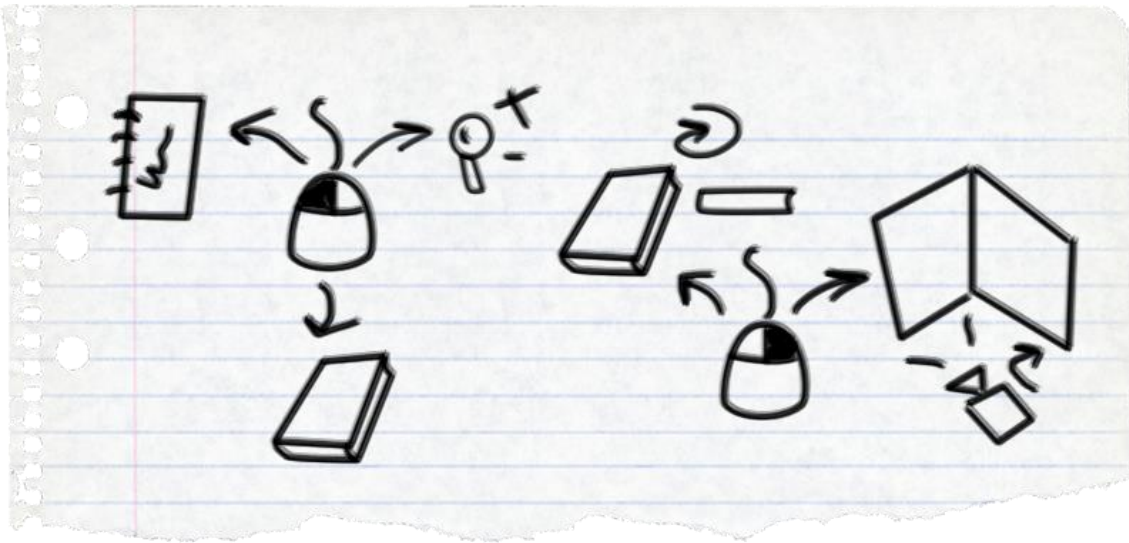


Illustration 1 Left and Right mouse click functions

3.4 GAME FLOW

The main menu is the first screen that appears in game after its execution. Here is the game title and options to create a new game, play an existing game or exit the game.

If the Create option is chosen, instructions to follow to create a new game will be shown to the user. He must give a name and draw twelve characters. The user will be able to navigate this screen by using two arrows at the sides that will allow access to the introduction of the next or previous character. By this way the user can edit the name or drawing of any of the twelve symbols that are being currently entered. At the end you can choose the option validate and save the game under a name. When playing a saved game, the characters to be solved will shuffle randomly to give the game replayability. After saving the game the main menu will be shown again.

If the Play option is chosen, a saved games list will be shown. The game will start when one of these is chose.

The game will end if game time is over or if you open the door key.

Illustration 2 represents a flowchart of the game screens described above.

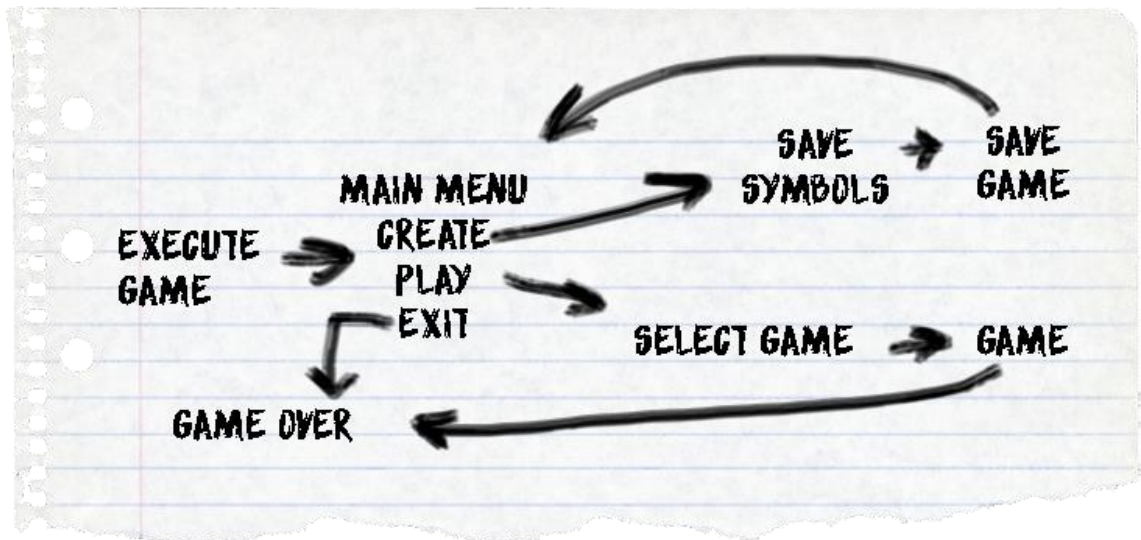


Illustration 2 Flowchart of Game scenes

3.5 GAME MECHANICS

Main game mechanics are:

- **Move the camera:** Camera management allows the player to explore the environment. It allows zooming in specific areas of the room to get a greater detail of the objects and turn at certain angles to have access to all the walls that make up the room.
- **Interact with objects:** Several objects of the environment are necessary for the resolution of puzzles so that they can be selected to save in the inventory. Another type of objects with which they can interact is those in which it is allowed to write the keys to open the locks.
- **Make actions:** The objects that are stored in the inventory allow to carry out concrete actions as a result of the logic and cleverness that are considered part of the challenge proposed by the game.
- **Write:** In the corresponding objects (papers and post-it) it is possible to write the keys to open the locks.

3.6 GAME ENVIRONMENT

3.6.1 ENVIRONMENT DESCRIPTION

The game is played in a one-room bunker. The first-person camera is in the middle of it. In the room you can see the exit door, a desk with several papers and a tablet, a bulletin board hanging on the wall, a trashcan with some wrinkled papers, a bookshelf and a ventilation grid. The watch that holds the countdown is at the top of the shelf. In *Illustration 3* the concept art of the room with the layout of the elements can be seen.

3.6.2 INTERACTABLE OBJECTS

The objects with which you can interact in the environment of this video game can be of two types: closed objects and objects of progress.



Illustration 3 Concept art of the room

Closed objects are those that require the introduction of a key to open them or show what they keep inside. The objects of progress are those that provide some clue about

the keys or trigger a decisive action for the progress of the game. They can also be closed objects.

3.6.2.1 CLOSED OBJECTS

These objects need the input of a key and the correct key has a consequence. It is necessary for the course of the game to enter the correct key in all these objects: the player cannot finish the game satisfactorily without having opened all the closed objects. They must also be opened in order as they depend on performing different actions such as finding progress, opening closed objects or a combination of both.

The key of each of the objects is different and is generated randomly. They also have a difficulty component in progressive order because the length of the last key to enter is greater than the previous one, the first being the shortest and therefore the easiest to write. In order for the key to be correct it is required to write each of the characters that compose it in the most accurate way possible and in the correct order. Therefore, if the key is wrong it may be for these two reasons that the player will be notified. Two faults are allowed when entering the key in the same closed object. The third and subsequent mistakes carry a penalty of a decrease of the countdown in twenty seconds.

The closed objects that can be found are the following:

- **Box Office:** It is the first object that can be opened. The only dependency is to find the objects of progress that have the clues to be able to solve the riddle in which the keys are based. These are one of the paper balls that are in the trash, the sheet of a book and a post-it hanging on a bulletin board.
- **Tablet:** Getting the key that protects the tablet requires a succession of actions besides opening the box office. These are to open the grid that hides the screwdriver, take the screwdriver, unscrew the plate from the bottom of the locker to find the switch, press the switch to turn off the light and look around to find the key painted on the wall.
- **Exit Door:** It is the final closed object and the one that allows achieving the objective of the game. Therefore it is the most difficult object to open. To find

its key is necessary to have followed the logical sequence of the game and to have surpassed all the puzzles until the moment.

Table 7 se shows a general summary of each of these objects and their concrete properties: The order, the length of the keys, their dependency and the consequence when opening them. In *Illustration 4* the closed objects have been highlighted to see more clearly their distribution within the room.

Table 7 General overview of closed objects

| <i>Order</i> | <i>Object</i> | <i>Length</i> | <i>Dependencies</i> | <i>Consequences</i> |
|--------------|---------------|---------------|--------------------------|---|
| 1 | Box Office | 2 | Find objects of progress | Give access to the light |
| 2 | Tablet | 4 | Find objects of progress | Show the riddle to find the key to the door |
| | | | Open Closed Object | |
| 3 | Exit Door | 6 | Open Closed Object | Exit |



Illustration 4 Closed objects in the room

3.6.2.2 OBJECTS OF PROGRESS

These objects provide clues to decrypt the keys of closed objects or facilitate other forward objects.

They are accessed by exploring the environment, opening closed objects or performing actions.

The objects of progress that can be found are:

- **Paper ball:** Contains one of the clues to solve the riddle of the keys. It will be next to other crumpled paper balls inside the bin.
- **Post-it:** Attached to the bulletin board, is part of the puzzle.
- **Switch:** Located behind the locker so it is necessary to open it and remove the bottom plate. The switch turns off the light of the room making visible the key on the wall.
- **Key on the wall:** Drawn on one of the walls of the room, can only be seen in the dark and shows the combination of symbols to write on the tablet.
- **Ventilation Grille:** The ventilation grille is at the top of a wall and inside keeps another object necessary for the advancement of the game: the screwdriver. It is necessary to hit it to make it fall.
- **Screwdriver:** Is an object that can be stored in the inventory for later use. It is inside the grille and will have to be used to remove the locker plate and uncover the switch.
- **Book:** A book placed among several on a shelf in the room, it keeps among its pages another clue.
- **Book sheet:** It is the last piece of the riddle, it allows the player to connect the symbols with the characters to be inserted into the closed objects.

Illustration 5 shows where these objects are positioned in the room. *Illustrations 6, 7 and 8* reveal how keys are encrypted and how they are displayed in the game.



Illustration 5 Objects of progress in the room

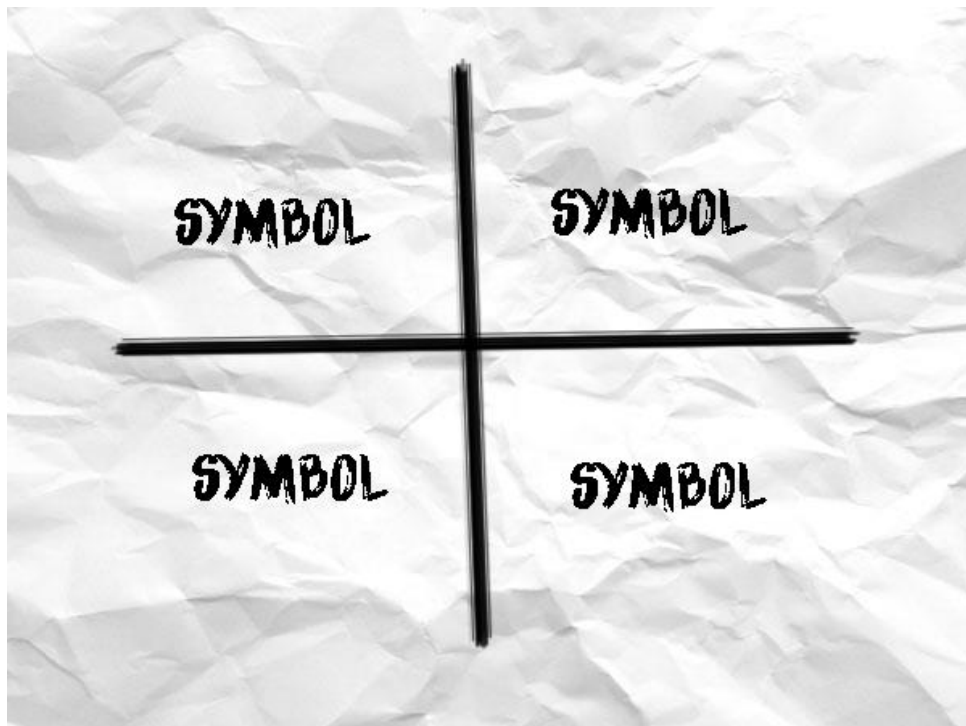


Illustration 6 Clue in a crumpled paper

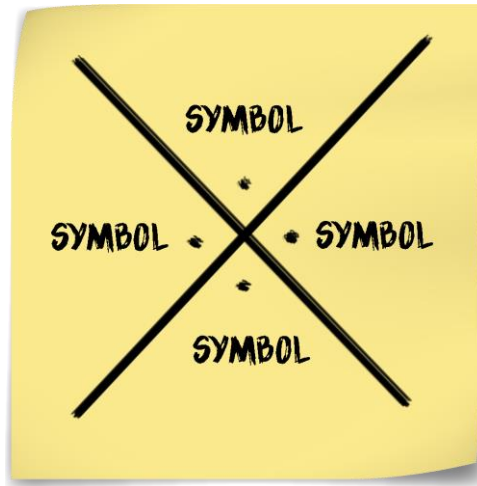


Illustration 7 Clue in the post-it

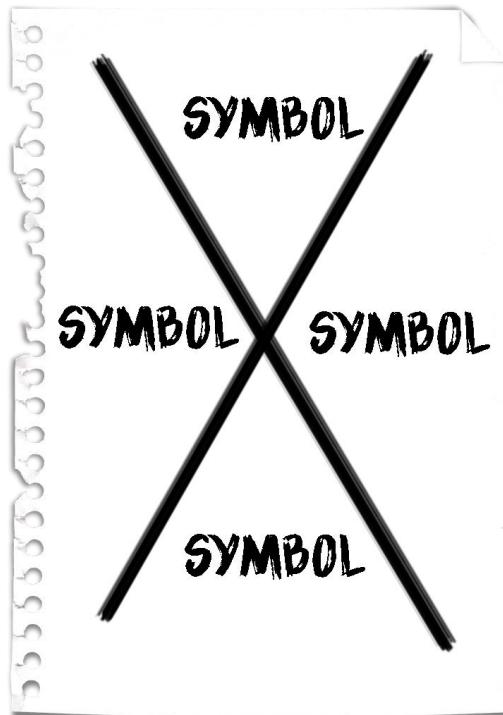


Illustration 8 Clue in a book sheet

3.6.3 RIDDLES PUZZLES

For this game of escape a sample of puzzles and riddles of different types have been designed. The puzzles require that the player explores of the environment, the use of the logic and the ingenuity and the connection between concepts, besides the use of memory and the development of the ability to write by hand. *Illustration 9* presents a diagram showing the logical sequence of the game.

Exploration is one of the key pieces in this genre of games because it allows, in the first place, to locate the closed objects and the characteristics of the key or code that each of them requires and, secondly, find clues and objects of progress to deal with the course of the game. All this is necessary to devise a strategy to follow.

When finding objects it is necessary to use logic and ingenuity to use them so that they take you to the next track. An example is making use of the screwdriver to remove the sheet from the box office or push the switch to discover something that with light is not seen.

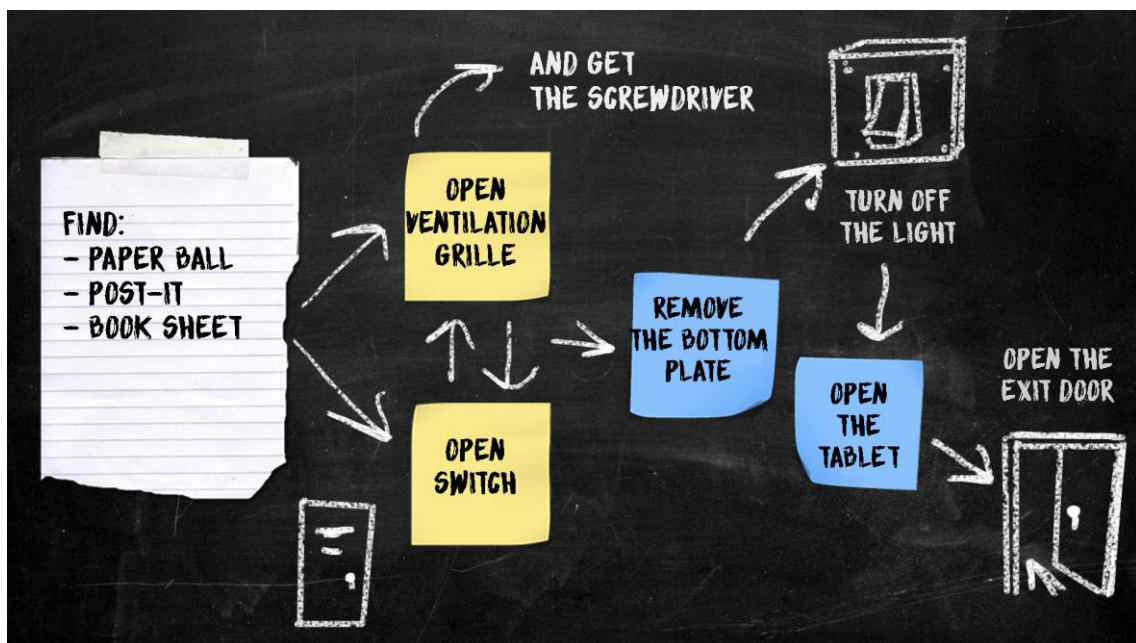


Illustration 9 Logical sequence of the game

The keys of closed objects are encrypted, *Illustration 10* shows an example of this encryption. To decipher them it is necessary that the player connect the tracks found in other objects of the game with the meaning of these keys.

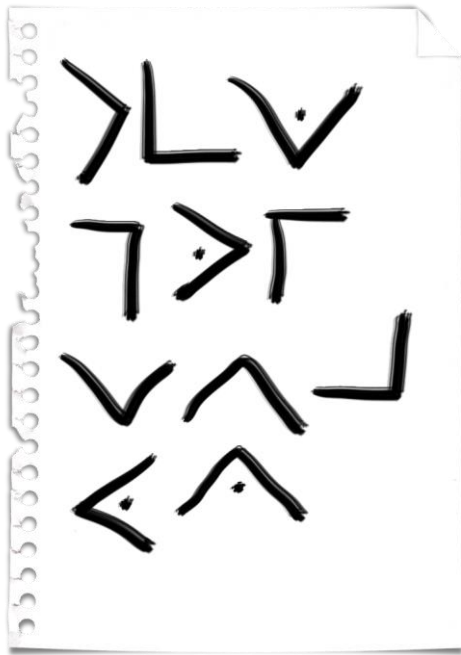


Illustration 10 Encryption

4 DEVELOPMENT

This chapter explains the in-depth development phase highlighting the implementation work done in detail.

The structure of this chapter is linked to the methodology carried out. The initial planning and requirements gathering phases have followed a classic life cycle. However, with the project progresses to the analysis and development phases, an agile methodology based on Sprints has been applied, in which the small and complete objectives must be met in order to give solution to the different components of which the project is composed. These sprints have a duration of one, two or three weeks depending on the complexity and difficulties that have appeared. In this chapter therefore do not enter into the complete description of the whole development process but it gives the pertinent information to the main components of the game, always related with the sprints in which they are included.

These sprints are in two big tasks which divide the development into two large blocks of work. The first of them is the development of the prototype, where all of mechanics and technologies with which it had to be released, are encompassed. The second one is the game development where the previous utilities are applied to carry out the final game. The *Illustration 11* presents the structure of the different development phases and sprints.

4.1 CREATION OF A FUNCTIONAL PROTOTYPE

The first of the two major development blocks aims to address the technological challenges presented by the video game. Handwriting, persistence between games, comparison, functionality and assembly of the whole prototype are included in this block. Once the functional prototype meets all the needs, the development of the video game can focus on other aspects such as the design of the scenes, among others. Thus streamlining the process.

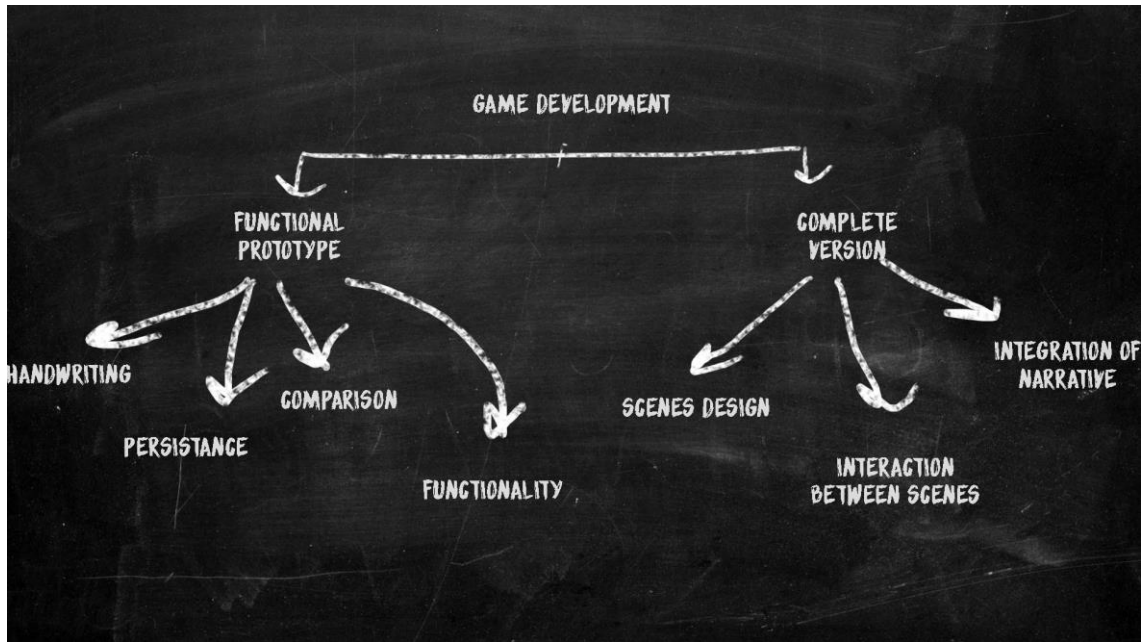


Illustration 11 Structure of the Development Phases

4.1.1 HANDWRITING IN UNITY

The main mechanic in this video game is the handwriting. Also it is the educational competence to achieve within this. The implementation of this mechanic in Unity can be realized in several ways. The first option is create, to each line segment forming the drawing, a Game Object with a Line Rederer component whose result would hundreds of Game Objects inside the scene. The second option is to render lines in a textures called Render Texture which are created and are updated in execution time. The textures are applied to the camera and they can be rendered. The exact pixels to which each line point corresponds would be obtained and Bresenham's line algorithm, for image processing, would be applied. The last option considered has been create a procedural mesh. This option has been chosen for the implementation.

To implement this method in Unity, two scripts are used. One of them is in charge of capturing the iterruptions caused by the left click of the mouse on the area in which it is possible to write. The other is responsible for creating the stroke and show it in the game world as explained below.

In one frame, if the left click of the mouse is pressed and its coordinates are inside of drawable area, the coordinates are saved as initial point of the stroke. To the next frame, if the lef click of the mouse is held, the coordinates are saved as end point of the same

stroke. To render a mesh in Unity is necessary specify the vertex and triangles of the 3D object which the mesh represents, the transformation matrix of the mesh, the type of the material and the number of layer in which the render will be performed. To create the mesh is necessary to obtain the vertices and triangles that form the planes representing the line segment. The four vertices of the planes can be calculated from the start point and the end point of the stroke captured in each frame. *Illustration 12* shows how the vertices are calculated.

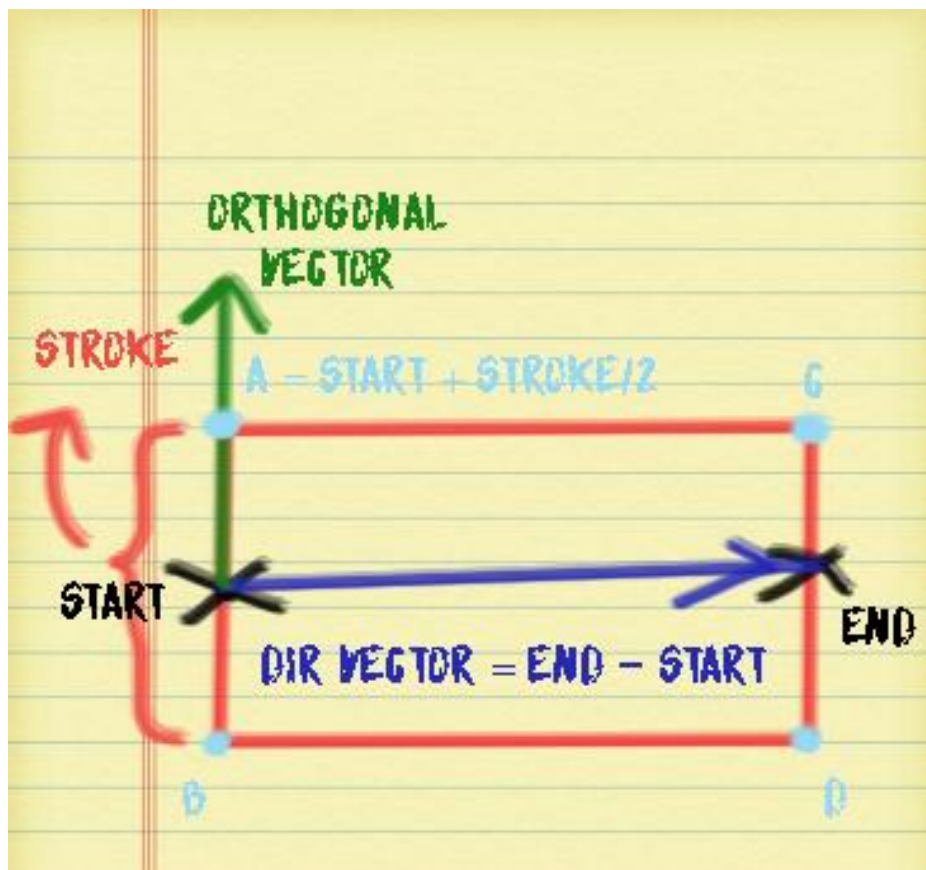


Illustration 12 Calculation of the vertices of the mesh

After finding the vertices are assigned to the mesh next to the six indexes of the vertices that form the two triangles that make up the plane. Once the first line fragment is achieved, the following fragments are added to it and they are created in the same way.

When the left click of the mouse is released a Game Object is created with the mesh that was previously created as component. An example of the result achieved with this implementation is shown in *Illustration 13*.

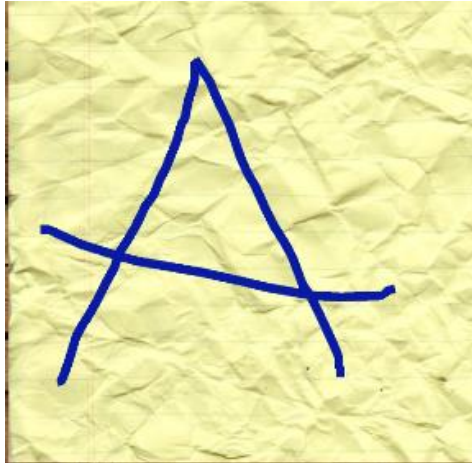


Illustration 13 Result of the handwriting tool

4.1.2 FILE PERSISTENCE

To save a new game, the necessary information is saved in a file in disk. This information is composed of the game name, the name of the twelve symbols and their corresponding shape in format png in addition to the matrix of integers that corresponds with this shape. To save the drawing in format png is necessary to create a 2D texture coded in bytes.

Unity offers three solutions to persistence problems. The first is to use the PlayerPrefs object that allows to store key-value pairs locally. This technique offers little flexibility and lacking security as far as the stability of the game is concerned.

The second option is to use a database with which to interact, developing intermediaries able to communicate the game in real time with database. This solution is powerful and offers safety guarantees but is more complex than necessary for this video game.

The third option, which has been applied, consists in serialize the required content and store it binary. C# offers utilities that make the process simpler, ensuring certain security in the game and streamlining the development process.

For developing this a static method in the Persistence Controller class is responsible of create a file with extension .dat in which all data will be saved and the file will be serialized. *Illustration 14* shows a small piece of the code responsible for this.

```

public class PersistenceController {
    public static bool guardarDatos(ObjetoJuego partida)
    {
        try
        {
            string nombreFichero = partida.nombrePartida;
            BinaryFormatter formateador = new BinaryFormatter();
            FileStream file = File.Create(Application.persistentDataPath + "/"
+ nombreFichero + ".dat");
            formateador.Serialize(file, partida);
            file.Close();
            return true;
        }
        catch (Exception e)
        {
            Debug.Log("Error Guardando: " + e);
            return false;
        }
    }
}

```

Illustration 14 Code of save data in Persistence Controller

The Class ObjetoJuego has to be Serializable, and with this easy code the file is properly saved. The same happens in the other way, when loading a game the objective of the script is to extract from the binary object the serialized data.

When selecting a game, another static method of the previous class, check if there is the file with the name of the selected game. If so, open it and deserialize it.

There are some classes related to the persistence that are interesting from the point of view of the programmer. For example the "Application.persistentDataPath" method helps a lot because it allows you not to worry about the routes where the game will be saved. The `BinaryFormatter` is the one that helps with the binarization of the data.

4.1.3 CHARACTERS COMPARISON

To character comparison, a integer matrix has been chosen as data structure to save the path the player makes with the mouse. The drawing displayed and the data matrix are separate functionalities. To save the stroke in the matrix, the width and height of the

drawable area are divided in as many rows and columns as the dimension of this matrix. Each frame the coordinates, in which the player clicks, are collected if these are inside of the drawable area. The column corresponding to the point at which the player has clicked can be calculated from its x-coordinate as shown below.

$$coorx = orix + cellWidth * column$$

Being *coorx* the “x” coordinate gathered by mouse click, *orix* the coordinate “x” of the canvas upper left corner of the Rect Transform (`lienzoRectTransform.offsetMin.x`), *cellWidth* the resulting cell width of having divided the canvas width between the matrix dimensions and *column* from where it can be extract the column index of the corresponding matrix.

From the latter relationship, it can be clear away *column*. Its whole part corresponds to the column index matrix.

From the “y” coordinate of the mouse click position the row can be calculated. Unity coordinates origin corresponds to the lower corner, this means that the mouse “y” coordinate have to be inverted and subtracted to the screen height.

$$coory2 = Screen.height - coory$$

Being *coory2* the mouse “y” inverted coordinate, *oriy* the upper left corner “y” coordinate of the Rect Transform canvas (`-lienzoRectTransform.offsetMax.y`) and *cellHeight*, the resultant cell height of having divided the canvas height between the matrix dimensions; the row relation states as follows.

$$coory2 = oriy + cellHeight * row$$

The whole part of *row* corresponds to the matrix row index.

Once the corresponding mouse clicks coordinates of the row and the column have been calculated, it is stored a one at the row-column of the matrix which initially was filled with zeros.

Testing if a new character introduced corresponds with the stored one is done by a static method that receives two matrices and returns a float represent the percentage of accuracy.

4.1.4 PROTOTYPE FUNCTIONALITY

This milestone was thought to be able to test the previous functionalities without taking account the game loop and at the same time set up a strong basis from where to build the mechanics of the final game.

The prototype has a main menu from which it can be start a game or create a game (*Illustration 15*). When creating a game the user is requested to introduce the name of the symbol and to write its shape one at a time, twelve times veces (*Illustration 16 and 17*). All this happens at the same scene where two panels alternate, the first one has an input field to write the name of the symbol and the latter has an active canvas. Once the sequence is completed it is allowed to validate the twelve symbols and save the game or discard everything.



Illustration 15 Main Menu of the prototype

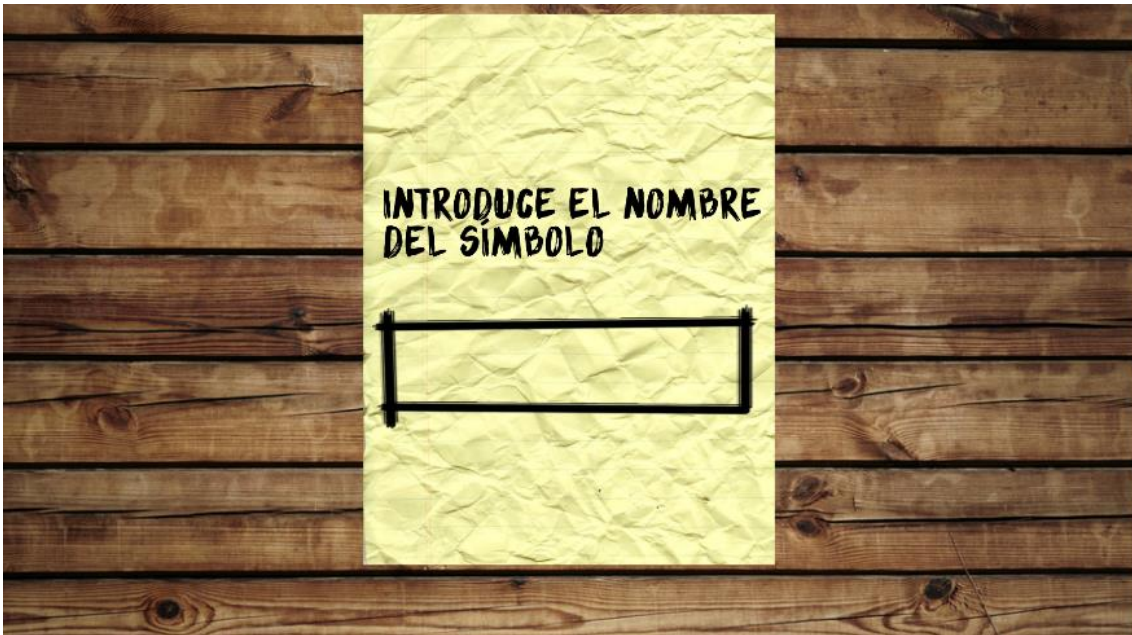


Illustration 16 Enter symbol name



Illustration 17 Symbol drawing

Hitting Play, a saved games list is showed (*Illustration 18*). At *Illustration 19* it can be observed the game prototype environment. At this environment, it can be clicked onto any object seen triggering three seconds in which the symbol is shown and in which the player has to memorize to write it later. If all the symbols drawn are correct, the game is over.



Illustration 18 Game List



Illustration 19 Prototype environment

All the images included at the prototype have been treated and have pursued the same theme regarding the game. It has been also included a font that resembles to the marker strokes to achieve integration of all the elements.

4.2 IMPLEMENTATION OF THE VIDEO GAME

While the prior section concentrates at overcome the technical problems by means of a prototype, the sprints that come at this phase will focus at the design, develop and narrative of the game itself. Before starting with the sprints the develop surrounding is adequated, the repository, all the assets generated and obtained are grouped and structured (audio, models, textures, useful scripts done previously...) inside the project folder. From there, three clearly different phases begin: Scene design as were described at the GDD, the communication and persistence between them and the integration of all the narrative and exportation. The next step would be verification and validation, however at the present document it has been excluded from this chapter and it has been placed at chapter six.

4.2.1 SCENES DESIGN

The designed scenes correspond to those indicated at *Illustration 2*. Thus, it creates, design the interface elements, assets elements and cameras needed are placed. The scene that carries the higher workload is the game scene itself. The rest don't differ at functionality from the prototype.

For the bunker scene it has been implemented a script that handles the camera zoom movement whenever the player performs a left mouse click at any of the interactable objects. It also handles the ninety degree rotation that happens with the right mouse click. The camera by default is looking towards a wall.

At this point is important to notice that, regarding the design, the art of the videogame has pursued a non-realistic aesthetic. The room and all the objects have been modeled. For texturing, it has been created, searched and converted textures to PNG format. Every element of the HUD has been created and draw.

In addition, there have been written all the scripts needed to time management, object interaction, updating and functionality of the user interface. As well as all the consequences for every possibility that are described at the correspondent section of the GDD.

Although at the chapter this sprint is exposed as an only section for readability, it took some small iterations which made the scene work properly altogether. The usage of the Unity3D tool expose at this phase its greater complexity, given that the animations, the interface updates, raycasts, camera gestion and even the shader appliances (pre-generated by Unity) are found at this phase to serve as visual feedback to the user.

The final outcome of this phase is a series of scenes that work properly individually but still have to be assembled and work correctly altogether.

4.2.2 INTERACTION BETWEEN SCENES

Once all the scenes have its proper function, they have to be communicated. To do that it is necessary to know what information needs one from another. At this point, it also joins the action all the functionality generated for the system persistence.

Generally Unity destroy every object at every scene change, filling each new scene with its own gameObjects, however there are several occasions in which it is needed to keep objects between scenes. With that purpose there is a method called DontDestroyOnLoad (*Illustration 20*).

```
using UnityEngine;
using System.Collections;

public class ExampleClass : MonoBehaviour
{
    void Awake()
    {
        DontDestroyOnLoad(transform.gameObject);
    }
}
```

Illustration 20 Game Objects persistence code

By using this object, it is possible to keep the functionality that the script has as component.

Generally, the objective of this sprint is to keep under control that the game loop plays correctly, analyzing all its conditions.

4.2.3 INTEGRATION OF NARRATIVE

Besides the game like it already is could be consider ready to be built, this phase focuses at the integration of the narrative. It has to be guaranteed that the user is conscious of the state that he is, and that it is given to him the minimum information needed for the player to understand the narrative depicted at the GDD. This task is partially executed at the analysis, however it happens that when the game loop is functioning is when most of the narrative problems are detected. Assumed information that was not given or redundant information that bores the player.

To this phase it is necessary again to keep track of all the assets, contents and shaders needed to make them all to fit. The musical rhythm, for instance, has to respond to the game timing, to the distance to the end that the player actually is. It has to provide ambience and even feedback from which rush up or to know that something correct has been done.

This phase is not focused at the generation of new scripts, although it could be needed, its purpose is to conclude positive with the planned narrative and the narrative that finally happens to be.

4.3 BUILDING THE GAME

This phase is not found at any sprint diagram because it happens transversally along all the development. The game has been built several occasions to guarantee the functioning and the performance over different systems.

Building the game with Unity is relatively easy, it comes with an assistant that allows to config various aspects and to generate the file needed for the pointed resolutions. Given that the used version was the free one, there are multiple features that cannot be modified.

The tool generates an .exe file along with a data folder whenever it is exported to PC and directly to an APK file if it is an Android exportation. Both files have been tested at different devices to verify that the incremental implementations of the different scripts put out the expected results.

5 RESULTS

The project, the development of the idea and the work that has been carried out for its development have been presented in the previous chapters.




The executable can be downloaded in <https://drive.google.com/open?id=0B-EZCkNSy5xhdm9iYW5BVEdzTHc>. In it the result of the previous chapters can be observed. Is necessary download the full contents of the folder for the game works. The presentation and the video to be used during the project defense are accesible at <https://drive.google.com/open?id=0B-EZCkNSy5xhMHE4V0lOeHlyODQ>.

Developing this project has been possible thanks to many of the subjects taught in the bachelor's degree of Video Game Design and Development. Some of these subjects are mentioned in chapter one but many more are related to this project like VJ1227 Games engines, which teaches the operation of a game engine (especially Unity); VJ1224 Software Engineering, which teaches agile methodologies or VJ1223 Art of video game, which teaches the creation of the graphic material that a video game needs.

A paper about educational games has been written in addition to this work. This paper that has been accepted and submitted at the Congreso de la Sociedad Española para las Ciencias del Videojuego (CoSECiVi). This paper presents a literature review of educational games and it has served as complementary work to this project.

A list of the objectives to achieve are presented in the chapter one. Here it will be checked which of the elements of this list have been achieved.

Objectives:

- Develop a video game in which it be can use and practice handwriting. 
- Convert handwriting in the key piece to resolve the game's puzzles. 
- Make this video game a suitable tool for anyone. 

All the objectives, that were proposed in the beginning, has been achieved. The results of the project have been expected but can still be improved.

6 TESTING AND EVALUATION

This chapter describes the tests performed to ensure the correct operation of the game as well as its quality control. This phase is responsible for the final video game does not present errors and the user experience was as expected. It can be raised from two differentiated phases. The user experience is worked in the first phase and it is known as verification and validation. Tests on devices with different capacities are performed in the second phase. This allows to evaluate the yield by means of the analysis of some key points described in the corresponding section.

6.1 VERIFICATION AND VALIDATION

The validation process focusing in check that final product meets the requirements. The verification focusing in the review of each develop phases.

6.1.1 VERIFICATION

The work of verification, with the limitations of the project, has been made entirely by the developer with the collaboration of the supervisor. This process has been performed in parallel throughout the duration of the project development. This is a looping task and it has ensured that the project, as a whole, remains consistent and its behaviour is the desired. All this whenever their pieces were modified or added.

6.1.2 VALIDATION

People accessible and outside the project, like family and friends, have cooperated to check the final product. These people have performed informal user test which have given information of two types.

- On the one hand software bugs and errors, that were not detected in the verification process, haven been highlighted, thus helping to correct them.
- On the other hand new ideas and suggestions have arisen. This allows a better game experience.

The temporary limitations of the project have not allowed to develop this phase with usability test like SUS (System Usability Scale [23]) or unit tests [24].

6.2 EFFICIENCY

For the performance tests three key points have been selected that will be analyzed in two devices with different performances and with different resolutions.

The points to be analyzed are described below:

- **Initial charge time (TEST1):** The time it takes the game from the user to execute (with double click) until the execution begins. This will be measured with a manual stopwatch.
- **Scene loading time (TEST2):** The time it takes to load all objects in the scene until it is accessible. This measurement is done by means of an integrated counter in the game in the development version. This counter is subsequently deleted in the publication.
- **Disk access to manage persistence (TEST3):** In the process of save a symbol, the system must access the disk to store both the generated image and the created matrix and the attached metadata (symbol name, game,...). The time is calculated from the time the storage order is executed until it has been successfully completed. This measurement is done by means of an integrated counter in the game in the development version. This counter is subsequently deleted in the publication.

All these tests have not been carried out by the temporal scope but are planned for a future. The tests will be carried out on medium performance and high performance computers based on their characteristics of processor, graphics card and RAM memory.

In addition the tests will be performed using different resolutions, all in 16: 9 ratio.

- Low resolution (**1280x720**)
- High resolution (**1920x1080**)

The expected result is the guarantee of the correct operation of the application in equipment with average characteristics. In addition, teams with better features will provide a faster user experience.

7 PROJECT DEVIATIONS

The project has suffered some modifications from its initial idea. This section will show the changes that have been both as to the game conception and to the planning that has been taken on.

7.1 GAME CONCEPT

The game concept has not undergone substantial changes, the main idea has been kept and it has only been changed the magnitude. Initially, the project was conceived as a longer gameplay experience and more complex environment. For a matter of time and ease to testing, it has been decided to create a three minutes prototype that embraces a significant sample of the possibilities that the project offers, it also states the basis to ease a future enlargement.

The game concept as a tool, this is the possibility of create a new game, has changed. At the beginning, it was thought that it will be shown to the user the game scene and the user could choose the elements that would be the encryption solution. Besides it would be allowed to choose that would appear at each clue. The game was going to be saved through an option present at the environment of the game. Whether some of symbols or keys were not been modified, the game had some of them by default option. This could lead to several problems and be boring for the user, so, twelve symbols have been put that the user must write and name. At the time of playing, the symbols will be randomized to give the game replay value.

7.2 PLANNING

At section 1.4 of this document, an initial planning was presented to confront the development of the project, however at the time of being undertake it has been modified. *Tables 8, 9, 10, 11, 12 and 13* show the comparison between the hours estimated and the actual work hours done.

Table 8 Phase Documentation - Differences in the planning

| Documentation | | |
|------------------------------|--|---------------------------------------|
| <i>Tasks</i> | <i>Estimated duration (in hours)</i> | <i>Actual duration (in hours)</i> |
| Technical Proposal | 8 | 8 |
| Technical Report | 40 | 70 |
| Video | 2 | 2 |
| Project Defense Presentation | 10 | 8 |
| Total | 60 | 88 |

Table 9 Phase Game Development - Differences in the planning

| Game Development | | |
|------------------------------------|--|---------------------------------------|
| <i>Tasks</i> | <i>Estimated duration (in hours)</i> | <i>Actual duration (in hours)</i> |
| Digital handwriting | 30 | 15 |
| Symbol recognition | 60 | 45 |
| Turn video game into a tool | 50 | 30 |
| Gameplay and mechanics programming | 40 | 60 |
| Total | 180 | 150 |

Table 10 Phase Game Design - Differences in the planning

| Game Desing | | |
|--------------------------|--|---------------------------------------|
| <i>Tasks</i> | <i>Estimated duration (in hours)</i> | <i>Acutal duration (in hours)</i> |
| Environment desing | 10 | 8 |
| HUD and GUI desing | 5 | 8 |
| Puzzles and clues desing | 15 | 6 |
| Prototype desing | - | 2 |
| Total | 30 | 24 |

Table 11 Phase Game Art - Differences in the planning

| Game Art | | |
|-----------------------------------|--|---------------------------------------|
| <i>Tasks</i> | <i>Estimated duration (in hours)</i> | <i>Actual duration (in hours)</i> |
| Environment modelling | 15 | 8 |
| Interactable objects modelling | 10 | 20 |
| User Interface art | 10 | 7 |
| Concepts art | 6 | 4 |
| Game Over, menu and final screens | 9 | 6 |
| Prototype image processing | - | 5 |
| Total | 50 | 50 |

Tabla 12 Phase Evaluation - Differences in the planning

| Game Evaluation | | |
|---------------------------|--|---------------------------------------|
| <i>Tasks</i> | <i>Estimated duration (in hours)</i> | <i>Actual duration (in hours)</i> |
| Validation of the project | 15 | 10 |
| Project performance | 5 | - |
| Total | 20 | 10 |

Tabla 13 Summary planning - Differences in the planning

| Summary table | | |
|----------------------|--|---------------------------------------|
| <i>Tasks</i> | <i>Estimated duration (in hours)</i> | <i>Actual duration (in hours)</i> |
| Documentation | 60 | 88 |
| Game Development | 180 | 150 |
| Game Design | 30 | 24 |
| Game Art | 50 | 50 |
| Game Evaluation | 20 | 10 |
| Total | 300 | 322 |

The art and design planning phase have been modified, besides on its duration, on the number of tasks. It was necessary to include the design and the processing that needed the images for the prototype.

8 CONCLUSIONS

This document presents an educational videogame which is both, a learning and different systems writing practice tool.

With this project it has been intended to fill a Serious Games weakness, specifically in the education field as there has not been found any reference of video games being used to teach to write. This is depicted at the state of the art, chapter two of this document, or at the paper “Misión desbloqueada: el diseño de los juegos educativos” mentioned at chapter five. A work that has been done simultaneously with this project and that has been accepted and submitted at the Congreso de la Sociedad Española para las Ciencias del Videojuego (CoSECiVi). The final result of this project reflects the potential of this type of games.

It has been proven that the escape genre offers an adequate framework to develop numerous skills and therefore for education. The riddles which can be included in the game can be designed to work the *hard skills* and the *soft skills* as the multiplayer could be easily implemented and allow then meet new competences like teamwork, communication, time management, critical thinking or self-leadership among others. As the main mechanic is the handwriting, it is allowed to develop this skill through the game, “hiding” then the educational purpose. This means that the challenge of create an educational videogame that unites its didactic purposes and the entertainment ones have been fulfilled. Besides, it has been allowed the editing of the game including new symbols by any user achieving this way the videogame to be also a tool so that anyone could augment its capabilities and teaching purposes.

8.1 FUTURE WORK

It has been thought several expansions of this project as a future work that are depicted at the following sections.

8.1.1 EXTENSION OF THE GAME

The most immediate is to design a more complex room so the game could last longer and therefore to have a larger number of riddles and puzzles. This way, it will achieve a greater practice at handwriting and a bigger acquisition and development at the rest of skills needed by the player at a escape game and that have been already mentioned. It have been also thought of including more objects for becoming part of the emerging narrative and from where the game story to take off.

8.1.2 EXTENSION OF THE TOOL

Nowadays, the game may be edited introducing twelve new symbols. At some point it could be release to the user a greater control at the time of create new games, for instance, letting him choose the quantity of new symbols that want to introduce, choosing the placement inside the environment or modifying the time to accomplish the escape.

8.1.3 FUTURE RESEARCH LINE

This project can be part of a future investigation lines opened at the paper “Misión desbloqueada: el diseño de los juegos educativos”. One of the possible investigation lines is creating an educational video game as an edutainment which teaches the player “secretly” avoiding the lack of interest just for being an educational video game. This project could build the basis of this investigation which has to proof the educational benefits hidden at this game scientifically.

REFERENCES

- [1] Dondlinger, Mary Jo (2007). Educational video game design: A review of the literature. *Journal of Applied Educational Technology* (pp. 21-31).
- [2] Fisch, Shalom M. (2005). Making educational computer games educational. *Proceedings of the 2005 conference on Interaction design and children* (pp. 56-61).
- [3] Wikipedia. Escape room.
<https://play.google.com/store/apps/details?id=air.com.mobigrow.canyouescape&hl=es>. [Last Access 08/02/2017]
- [4] GooglePlay - Can you escape
<https://play.google.com/store/apps/details?id=air.com.mobigrow.canyouescape&hl=es>. [Last Access 08/02/2017].
- [5] Cube Escape Series. <http://www.cubeescape.com/> [Último Acceso 13/06/17].
- [6] Torres-Ferreyros, C. M., Festini-Wendorff, M. A., & Shiguihara-Juárez, P. N. (2016, October). Developing a videogame using unreal engine based on a four stages methodology. In *ANDESCON, 2016 IEEE* (pp. 1-4). IEEE.
- [7] F. Zhu, [online] Available: <http://fzdschool.com/entertainment-design/industry-outlook>.
- [8] dos Santos Ruivo, J. M. A., Karim, K., O'Shea, R., Oliveira, R. C. S., Keary, L., O'Brien, C., & Gormley, J. P. (2017). In-class Active Videogame Supplementation and Adherence to Cardiac Rehabilitation. *Journal of Cardiopulmonary Rehabilitation and Prevention*.
- [9] del Blanco, Á., Torrente, J., Fernández-Manjón, B., Ruiz, P., & Giner, M. (2017). Using a videogame to facilitate nursing and medical students first visit to the operating theatre. A randomized controlled trial. *Nurse Education Today*.
- [10] Miralles, I., Castilla, D., & Botella, C. (2014). Desarrollo de un entorno virtual para la inducción emocional. In *CoSECivi* (pp. 230-238).
- [11] Orvis, K. A., Moore, J. C., Belanich, J., Murphy, J. S., & Horn, D. B. (2010). Are soldiers gamers? Videogame usage among soldiers and implications for the effective use of serious videogames for military training. *Military psychology*, 22(2), 143.
- [12] Michael, D. R., & Chen, S. L. (2005). *Serious games: Games that educate, train, and inform*. Muska & Lipman/Premier-Trade.
- [13] Brown, H. J. (2014). *Videogames and education*. Routledge.
- [14] Bogost, I. (2005). Videogames and the future of education. *On the Horizon*, 13(2), 119-125.
- [15] Gros, B. (2007). Digital games in education: The design of games-based learning environments. *Journal of research on technology in education*, 40(1), 23-38.
- [16] Barab, S. A., Scott, B., Siyahhan, S., Goldstone, R., Ingram-Goble, A., Zuiker, S. J., & Warren, S. (2009). Transformational play as a curricular scaffold: Using

- videogames to support science education. *Journal of Science Education and Technology*, 18(4), 305.
- [17] Patiño, A., & Romero, M. (2014, July). Identifying pedagogical uses of serious games for learning English as a second language. In *International Conference on Games and Learning Alliance* (pp. 31-43). Springer International Publishing.
- [18] Gee, J. P. (2003). What video games have to teach us about learning and literacy. *Computers in Entertainment (CIE)*, 1(1), 20-20.
- [19] Godwin-Jones, B. (2004). Emerging technologies. *Language in action: From webquests to virtual realities. Language Learning & Technology*, 8(3), 9-14.
- [20] Prensky, M. (2003). Escape from planet Jar-Gon: Or, what video games have to teach academics about teaching and writing. *On the horizon*, 11(3).
- [21] Kolar, T. (2017). Conceptualising tourist experiences with new attractions: the case of escape rooms. *International Journal of Contemporary Hospitality Management*, 29(5).
- [22] French, Sally. (2015), Online. The unbelievably lucrative business of escape rooms. Available: <http://www.marketwatch.com/story/the-weird-new-world-of-escape-room-businesses-2015-07-20> [Último Acceso 02/06/2017]
- [23] Brooke, J. (1996). SUS-A quick and dirty usability scale. *Usability evaluation in industry*, 189(194), 4-7.
- [24] Unity Test Tools. <https://unity3d.com/es/unity/qa/test-tools>. [Last Access 21/06/2017]