



Grado en Ingeniería Informática
Universidad Carlos III de Madrid
Escuela Politécnica Superior

Autor: Jennifer García de la Calle

Tutor: Gustavo Recio Isasi

Co-tutor: Alejandro Baldominos Gómez

Fecha: 24/02/2015

Resumen

La aparición de internet como medio de comunicación ha supuesto que el acceso a la información sea sencillo y rápido. La mayor parte de esta información reside en las páginas Web, que presentan texto e imágenes en dos dimensiones, sin embargo el mundo real es tridimensional, por lo que al reducir el “mundo” Web a sólo dos dimensiones se está perdiendo información, de ahí la conveniencia de la integración de una tercera dimensión que permita, por ejemplo, recorrer las instalaciones de un museo hasta llegar a la información que le interese al visitante.

Actualmente la gran mayoría de los museos de España no cuentan con una visita virtual, y los que cuentan con ella no lo hacen con una visita libre, en la mayoría te mueves con imágenes, pero el usuario no interactúa de forma directa con el entorno, ya que en todas ellas estas muy limitado.

Las visitas virtuales se encuentra dentro de un importante campo de las Ciencias de la Computación denominado Realidad Virtual, este campo tiene importantes aplicaciones en la educación, y es muy útil para estimular el proceso de aprendizaje. Las aplicaciones de realidad virtual consiguen un efecto llamado inmersión, según el cual los usuarios pueden interactuar completamente con el ambiente artificial utilizando los sentidos del tacto, el oído, y la vista mediante dispositivos especiales que están conectados al ordenador, como el Leap Motion, Oculus Rift o Brain Computer Interface. Estos aparatos tienen sensores que detectan el movimiento de forma precisa, repercutiendo en el mundo virtual.

En el presente documento se expone la creación de una visita virtual para satisfacer esta necesidad, por lo que se ha creado una visita 100% libre, donde el usuario tenga libertad de movimiento por el escenario, y permitiéndole que interactúe con los objetos de la escena en la que se encuentra. Para ello se han utilizado diferentes formas de interacción, tanto actualmente utilizadas, como son el uso del teclado y ratón o el uso de un mando de consola, como el uso de formas totalmente novedosas mediante el uso de Leap Motion, pequeño dispositivo de control gestual que se coloca frente a la pantalla de nuestro ordenador. Además se ha utilizado un sistema de inmersión audiovisual, mediante el uso de gafas de realidad virtual, las Oculus Rift.

En el siguiente enlace puede verse un video que contiene el resultado final del desarrollo presentado:

<https://www.youtube.com/watch?v=P8t9yj6OXfl>

Abstract

The emergence of Internet brought simple and fast information access. Most of this information resides on Web pages, presenting text and images in a two-dimensional way. The real world, though, is three-dimensional and reducing the Web "world" to two dimensions is missing information. Hence, integrating a third dimension that allows, for example, creating virtual tours of a museum with information that attract the visitor, is a desirable practice.

Currently, the vast majority of Spanish museums do not have a virtual visits, and those that have it, does not allow free movement for the users. They provide movement over images, but the user does not directly interact with the environment.

Virtual visits are part of a major field of computer science called Virtual Reality. This field has important applications in education, and it is very useful for stimulating the learning process. Virtual reality applications get an effect called immersion, in which users can fully interact with the artificial environment using the senses of touch, hearing, and see through special devices that are connected to the computer, as the Leap Motion, Oculus Rift or Brain Computer Interface. These devices have sensors that detect movement accurately, impacting the virtual world.

This document presents a virtual visit to meet this need, in which a 100% free tour is created. In this visit, users are free to move around the scenario, and interact with objects in the scene in which they are located. To this end, users can interact with different artifacts of interaction, some are common, such as using the keyboard and mouse or use a remote PS3, and some are entirely new ways, as using Leap Motion, a small device that detects gestures, or a system of visual immersion, using a virtual reality helmet, the Oculus Rift.

A video of the implementation result can be seen in the following link:

<https://www.youtube.com/watch?v=P8t9yj6OXfl>

Summary

In this document, the creation of a virtual tour of a museum is proposed, as currently, museums do not have totally free virtual visits. A 100% free visit, where the user is free to move around the scenario, and to interact with objects in the scene in which it is created.

State of the art

Currently, the vast majority of Spanish museums do not have a virtual visits, and those that have it, does not allow free movement for the users. They provide movement over images, but the user does not directly interact with the environment.

Creating a virtual tour, has associated the same work as the creation of a video game. In its essence is the same, although in the case of a virtual tour, AI is not present. A little analysis on the current situation in this field is presented below.

Games: Video games, as everything that is related to technology, have had a major change in few years. They have evolved from just a few rudimentary means, to having a technology capable of recreating amazing virtual realities.

3D design is the field of computer science devoted to creating computer-generated three-dimensional models, being the basis of video games. It is one of the fields with greater development in recent years. Along computer technology, 3D design also achieves higher definition and processing capabilities, leading to increasingly realistic results. All games consist of two basic phases: modelling and game engine.

Modelling: 3D modelling is the process by which characters, objects and scenes that are used for both video games and movies are created. It represents one of the vital stages of game development and that more time and resources requires. In this area the most used programs are: Blender, 3ds Max and Maya.

Game Engine: A game engine is a term that refers to a set of programming routines that allow the design, creation and representation of video games. There are game engines operating in both game PS3s and operating systems. The basic functionality of an engine is rendering, i.e. displaying graphics on the screen for both 2D and 3D. In this area, the most used programs are: Unity 3D, Unreal Engine and CryEngine.

Another interesting aspect of this project is the use of new forms of interaction and virtual reality.

Interaction: New ways of interaction with video games are being developed, some already available, although not in their final versions. These new systems allow us greater immersion in gaming, providing a much more direct and real experience. Examples are: Leap Motion, Nuimble VR or Brain Computer Interface.

Virtual Reality: In the field of virtual reality there are also powerful tools currently under development, such as: Oculus Rift or 13th Lab.

Defining the problem

After analyzing the different technologies for the development of the project, this development will be explained:

Because the process of creating a virtual tour is very similar to making a video game, it will be used as a base for creating a FPS type game. Thus the project will have a dual purpose:

- Creating a virtual tour.
- Creating a FPS type game, to demonstrate that the creation of a virtual tour and a game start at the same point and with the same base

Modelling: The first thing is the modelling of 3D designs. For this, Blender is used. The following elements were modeled in this step:

- Building museum of arts and sciences Prince Felipe, this building's facade is modeled. It has been chosen since its structure is quite unique and complex, as it contains large scattered windows across the facade. Although it contains parts with the same structure, which is an advantage.
- Building inside. A simple inside will be modeled. It will put minimal load when rendering the game, giving users the feeling of being in an office.
- Outside the building. Like the inside, an equally simple outside is created, a forest that surrounds the main building will be included.
- Gun to shoot, when the game is driven with the PC standard input device, keyboard and mouse (hereinafter Named As PC).

Game Engine: The next step is to include all the models created in the game engine, Unity 3D, where the following tasks are needed:

- Including the models created with Blender.
- Texturing of these models, to give them the required realism.
- Creating scripts to control interaction with other objects, creating a virtual tour. Also, in an alternative version, enemies will be added creating a FPS game, where scripts are used to control the enemies and interaction with them.
- Creating menus that allow the user to choose the desired mode.
- Music is added to the various scenes of the game.

Interaction: two commonly used ways of interaction will be incorporated:

- PC (keyboard and mouse).
- A PS3 controller.

In addition, a very novel way of interaction is incorporated:

- LeapMotion: with LeapMotion, the user makes hand movements that he sees reflected in the output device, as if their own hands were in the game. The movements made by the user will be associated with actions that allow users to move around the environment in which it is located, as well as interacting with objects that are around.

Virtual reality: visualization environments shape lifelike be incorporated:

- Oculus Rift: These glasses allow the user to stand directly in the environment that has been created, having the feeling of being there. The glasses will be used to replace the screen, offering a much more realistic look.

As an alternative to this system, it is possible to use the PC screen, but losing the feeling of really being there.

Development

Analysis: This section aims to identify the features and options that the proposed project should provide. To do this tables of functional and non-functional requirements, and their associated use cases, have been created. This way we ensure that it covers everything that is required.

Design

Application architecture: The architecture of the application allows a high level design of the system and can clearly identify the former parts, and the relationship between them. This relationship can be seen schematically in Illustration 1.

Input: captures the action.

Core: The most important part of the game, as it encompasses the application, engine and logic.

Output: responsible of displaying the rendered environment

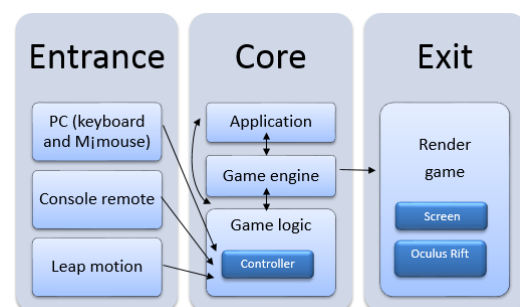


Illustration 1. Application architecture

Modelling: The whole scenario has been created from scratch for this project, for which we used Blender. We have chosen the design of the Prince Felipe Museum of Sciences Valencia, Illustration 2, this building was selected because of its complexity, as it presents a great challenge due to its complexity.

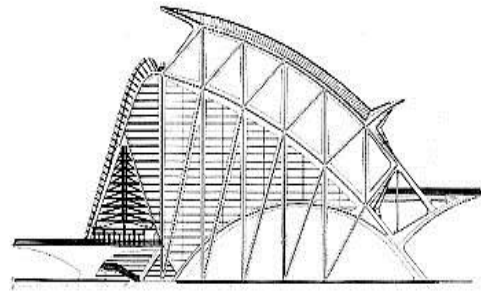


Illustration 2. Principe Felipe Science museum. Outer side view scheme

Game engine: Since the game has different ways to play, a menu has been created for the users to select which option they want to play. Each menu has a title so that the user knows where he is, and buttons to select the desired option. Illustration 3.

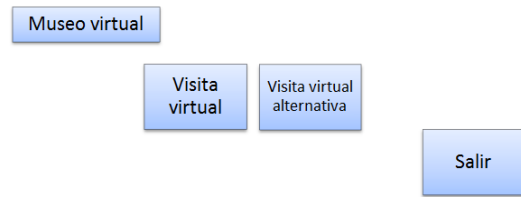


Illustration 3. Main menú design

The HUD (Head Up Display), Illustration 4, is the communication interface with the player during the visit alternative, what we see on screen information: life, ammo, etc.



Illustration 4. Leap motion HUD design

Implementation

Modelling:

The Príncipe Felipe Science Museum, is the centerpiece of the project. A design close to the original building has attempted, and it consists of west, southwest and lateral facades and roof. The first one created is the southwest facade, because it is the most complex of all.

The first thing to mention about this building is the fact that it contains a pattern, Illustration 5. This pattern has facilitated the design work because it only had to be created once, and then replicated.



Illustration 5. Principe Felipe Science museum. Southwest facade details

The modelling of this section can be seen in Illustration 6, marked in blue the area has been modeled. While its creation, materials had to be added to different parts of the building, Illustration 7 and Illustration 8. For simplicity, only three materials were added: two white materials representing the building structure, and one blue representing crystals. This is the same color that will show throughout the creation of the building.



Illustration 6. Principe Felipe Science museum. Southwest facade details

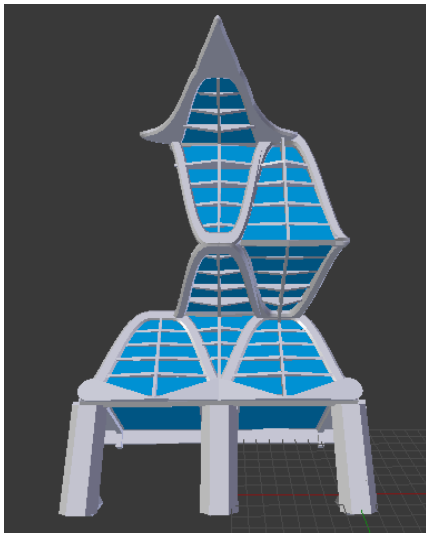


Illustration 7. Southwest facade Blender model. Front view



Illustration 8. Southwest facade Blender model. Side view

For other parts of the building were performed similar steps. To obtain the finished building, Illustration 9 and Illustration 10.

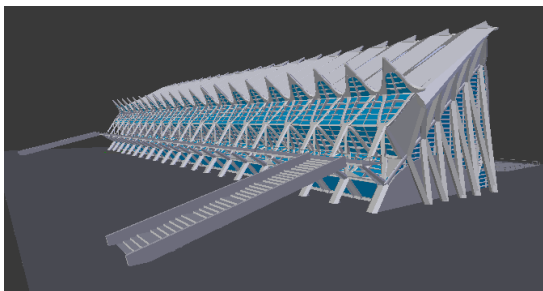


Illustration 9. Southeast facade Blender model

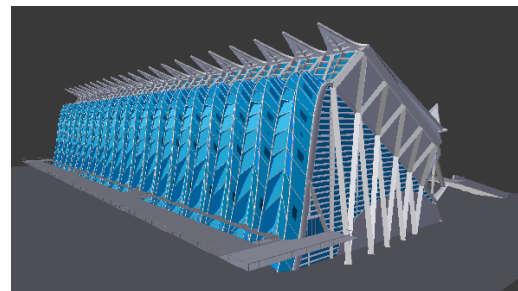


Illustration 10. Northeast facade Blender model

To decorate the inside of the building, a simple inside, lest minimal load when rendering the game has been designed, giving the feeling that we are in an office. The elements created were tables, chairs, lamps, screens, screen projection, projectors and computers, among others, example Illustration 11. Some of these objects are those which will be used to interact with during the virtual tour.

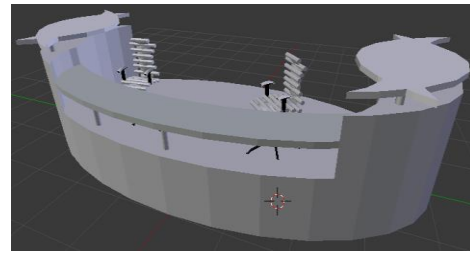


Illustration 11. Counter blender model

For the alternative virtual tour some more objects were designed, such as:

Weapon: When using the keyboard and mouse, or PS3 controller, you need a gun to shoot enemies that appear in the game. With Leap Leap Motion and motion with Oculus Rift options, a shooting hand66 will be used.

Life and ammunition: When enemies are defeated, there is a possibility to take extra objects. These objects are a life kit and a box of ammunition.

Materials and textures: For textures to render correctly in Unity, each material added in blender have been modified. Two types were taken into account, one for opaque materials and one for glass.

Opaque materials: The facade material, represented in white, has been designed to then being completed with an image. To that end, UV unwrap method is used which allows to apply a two-dimensional image on a 2 dimensional model. Images will be applied later in Unity3D. The right side of the Illustration 12, shows the two-dimensional representation of three-dimensional model, which is on the left side.

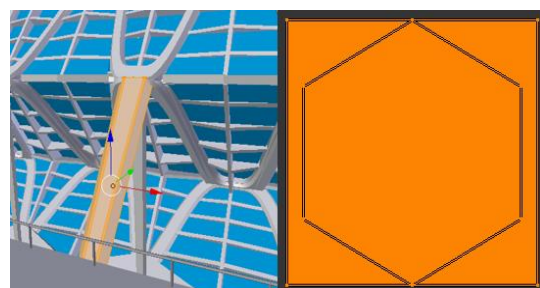


Illustration 12. Mapping process in Blender

The crystal blue material is also complex, but not so hard to get. In this case no image is applied. A transparent material that reflects light has been created, thus achieving a very realistic look to through the windows. In Illustration 13 perfectly crystal clarity can be seen. A green rectangle has been placed behind the facade, so the effect can be clearly seen.

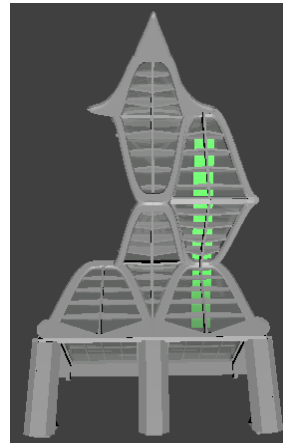


Illustration 13. Window's glass in Blender

Game engine:

Outside decoration: For the outside of the building a simple decoration has been created, so the weight of this part is very small and allows a fast rendering of the project.

Menus: Here it is shown how the design of the main menu has been implemented, Illustration 14.



Illustration 14. Final initial menu

Lighting: An equally important aspect of the game is enlightenment. It permits to achieve maximum realism of textures.

Ambient light is halved, to maintain light at the whole scene, but so it does not cover the directed light. Directed light gives the realism to the scene, and it has been strategically placed to give the feeling that it comes from the sun.

Inside the building, ceiling lamp and table were added. Also, ceiling lights can be seen on all floors of the building, each of which has three point lighting

Interaction: The main character is a specific object created with Unity, which contains all the necessary objects and scripts to interact with the game/visit. The character does not have any type of design, as it is a first person experience. Only the gun and fire ball were designed and modeled. Hands models were imported with Leap Motion plugin. The main character has the following elements:

- Main Camera → game camera, being a first-person game. The field of this camera is what will be seen by the player.
 - HandController → driver that enables us to use Leap Motion in the game. It was imported with the Leap Motion plugin.
 - Shot → in the alternative virtual tour, it is the object shoot to enemies, is within the HandController as the trigger is released by opening the right hand, and this is controlled by the HandController.

It also has two speakers:

- The one closest to the HandController, beeps when shooting, this sound is enabled only in the alternative virtual tour.
- The one in the character's body will reproduce music during the virtual tour.

In the Illustration 15 you can see some of those parts.

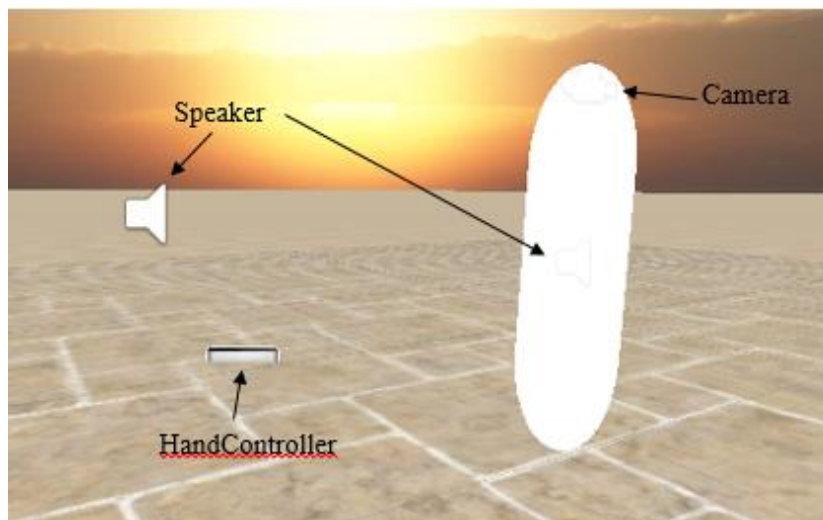


Illustration 15. Characters for the virtual tour with Leap Motion

In the game engine, if the option to make the virtual tour with Leap Motion is selected, the character would be controlled by scripts as follows:

- Move camera: turn left or right, the left wrist.
- Move:
 - Forwards: close left fist
 - Backwards: not implemented. Instead, turn the camera on one side and walk forward.
 - Left: does not have this option in place should rotate the camera left and walk forward.
 - Right: it has no choice, instead you turn the camera to the right and go forward.
- Shoot: open right hand. This option is only available for alternative virtual tour.
- Reload firing energy: close right. This option is only available for alternative virtual tour.

Interaction with objects: In the virtual visit, the character is located inside the building. The player here can interact with some objects, like the lamp shown. To make this interaction, objects must have an attribute called collider. This attribute lets us know when another object or character collides with it. When this happens, the game displays a description of that particular object. Depending on the object with which the interaction happened, the information will be one or another. In the example of the lamp, Illustration 16, bottom image, information about the object is shown, and the light is kindled.

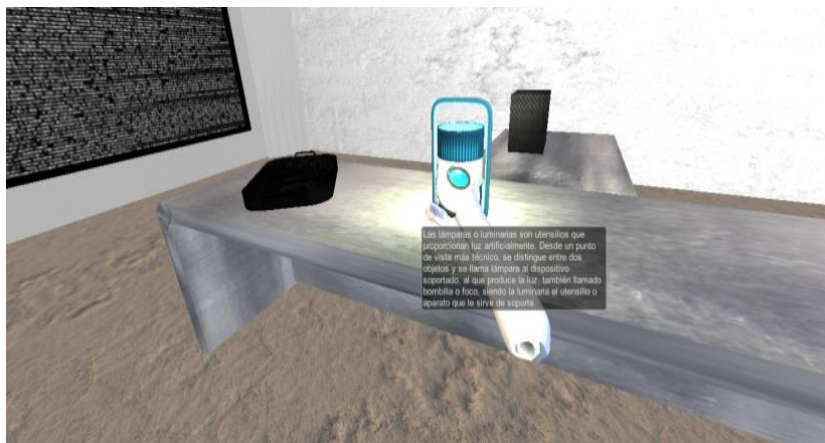


Illustration 16. Interaction with objects, Touched lamp

Kit and ammunition: The kit and ammunition are two objects that the player can pick up colliding with them. The kit increases the character's life. The ammunition object completely restores ammo.

Virtual reality: For the use of Oculus Rift, an OVR Player Controller is created, which contains:

- OVRCameraRig → this camera is special and different from the other two, emitting two projections. These projections contained in LeftEyeAnchor and RightEyeAnchor, for each eye in the Oculus Rift.
 - HandController → driver that enables us to use Leap Motion in the game. It was imported with the Leap Motion plugin.
 - Shot → in the alternative virtual tour, it is the object shoot to enemies, is within the HandController as the trigger is released by opening the right hand, and this is controlled by the HandController.
 - LeftEyeAnchor → allows us to see the image for the left eye.
 - CenterEyeAnchor → center of both images.
 - RightEyeAnchor → allows us to see the image for the right eye.

It also has two speakers, with the same functionality as seen before.

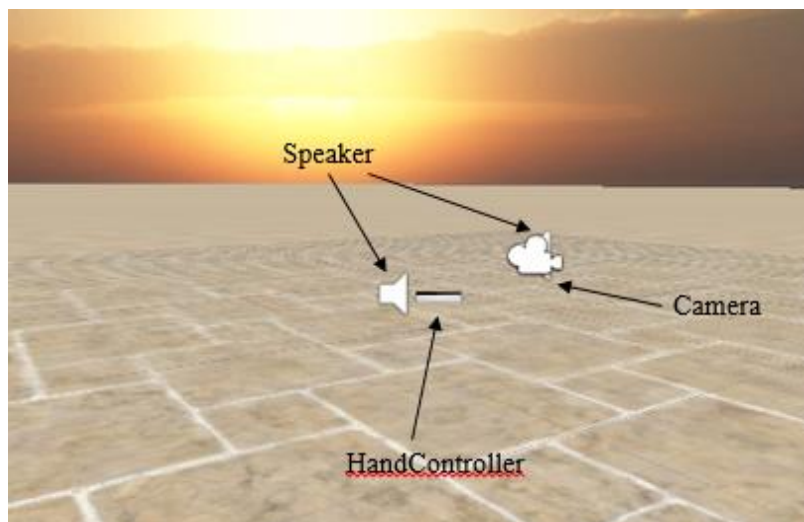


Illustration 17. Characters for the virtual tour with Leap Motion and Oculus Rift

If the option to take a virtual tour with Leap Motion and Oculus Rift is selected, the character is controlled just like in the case in which only Leap Motion is used as the controller is the same. In the Illustration 17 you can see some of those parts.

Testing: To verify the correct functioning of the application, it will be tested by six people. For its realization, users will be able to access to the user manual and the goals they have to do. After they tested the game, they were given a quiz to check how their experience went, and to get feedback of their experience.

Tests have two phases: the first in which they have to perform a series of goals, and the second in which they must answer a few questions about the operation of the application.

Tests have given good results, as expected. Every user has been able to perform all proposed goals without any trouble. The application evaluation shows that, in general, it has responded quite well, although it should be noted that there were rendering problems, since the project cost will run at times and the image get slowed. Another suggestion made is that the rotations are performed in an analog way. These are important facts to keep in mind for future improvements.

Conclusions

Contributions made: This project has met all the goals that were initially marked, and there have been some extras:

- We conducted a study on the developmental stages of a game.
- We carried out a study on current and new forms of interaction.
- The development of the game for a virtual tour, created from scratch:
 - Modelling Museum of Arts and Sciences in Valencia, with corresponding textures.
 - Modelling of all inside decoration, with corresponding textures.
 - Building outside design.
- We have created two different forms of game:
 - Virtual tour: here you can interact with objects in the scene, when an object is touched displays information about it.
 - Alternative virtual tour: in addition to showing the above information, enemies have been added, creating a first-person game in which you must try to survive.
- Different forms of interaction were added. Some are currently used by associating keys to actions in the game:
 - Keyboard+mouse or PS3 controller used to take an alternative virtual tour.
- Incorporation of new forms of interaction between the user and the system, associating actions with hands to actions in the game:
 - Leap Motion with or without Oculus Rift is used to make a virtual visit and an alternative virtual tour.
 - A menu that guides the user through the game has been added, depending on the type of interaction they select.
- Music added to the game, which helps the user to better integrate into it.

Problems encountered:

- We have simplified the side facade of the Museum of Science Prince Felipe, because it contained several vertices at the same point, which made it difficult to render.
- To get the textures displayed correctly, a material was assigned to objects in Blender, for then being added an image that textures it in Unity 3D.
- For proper display of texts with Oculus Rift, it was necessary to associate an extra script to camera glasses so that it displays the information of the objects.
- By using the Leap Motion and Oculus Rift option, it was necessary to rotate the HandController location, turning 300° and 180° on the X axis in the axis Z. Thus controller project is in the same position as in reality.

Future work: As future works, some possible improvements are presented. All these improvements could positively affect their operation, as detected in the evaluation.

- Rendering by parts. After completion of the tests, it has been seen where the program fails rendering, getting too slow sometimes. Partial rendering of stage would improve the result.
- Analog rotation. Another suggestion made in the tests is that the rotations are performed in an analog way. This could be implemented along with the actual functionality. User could use a menu of options, to choose which way to render.
- Increasing the range of forms of interaction. As new forms of interaction appear in the market, making the user experience and immersion can be improved, due to the improvement of existing technologies.
- Add the game cinematics, e.g. doors open when player is near.
- Implementation of a multiplayer system. The virtual tour would be even more interesting if players could do it with friends or family.
- Integration into a single executable. The part of the game with Oculus Rift requires to be compiled independently for it to work. Integrating it into the other compilation would enrich the complete project.

Índice de Contenidos

1. Introducción	23
Estructura del documento	26
2. Estado del arte	27
Visitas virtuales en los museos de España	27
Industria de los videojuegos	29
Diseño 3D	32
Modelado	32
Motores de juego	36
Interacción	42
Actuales formas de interactuar con los videojuegos	42
Nuevas formas de interactuar con los videojuegos que ya son una realidad.....	42
Realidad virtual	45
Conclusiones del análisis.....	47
3. Definición formal del problema	50
4. Planificación y presupuesto	52
Planificación	52
Metodología de desarrollo.....	54
Presupuesto	54
5. Desarrollo	57
Análisis.....	57
Requisitos del sistema.....	57
Casos de uso.....	58
Diseño.....	60
Arquitectura de la aplicación	60
Modelado. Escenario.....	61
Motor de juego.....	65
Implementación	68
Introducción	68
Modelado	69
Motor de juego.....	82
Interacción.....	90

Realidad virtual	95
6. Pruebas.....	97
Plantilla de pruebas.....	97
Resultados de las pruebas.....	99
Conclusiones de las pruebas	100
7. Marco regulador.....	102
8. Entorno socio-económico	104
9. Conclusiones.....	106
Aportaciones realizadas	106
Problemas encontrados y trabajos futuros.....	107
10. Bibliografía	109
Anexo I. Acrónimos	111
Anexo II. Glosario de términos.....	113
Anexo III. Metodología de desarrollo Scrum.....	115
Anexo IV. Requisitos del sistema	116
Requisitos software funcionales	117
Requisitos software no funcionales	121
Anexo V. Casos de uso.....	123
Anexo VI. Gestión de datos	128
Anexo VII. Manual de usuario	130
Anexo VIII. Análisis de la planificación y coste del proyecto tras su finalización.....	134

Índice de tablas

Tabla 1. Comparativa programas de modelado.....	36
Tabla 2. Comparativa motores de juegos	40
Tabla 3. Estimación presupuesto recursos humanos.....	55
Tabla 4. Estimación presupuesto recursos materiales hardware	55
Tabla 5. Estimación presupuesto recursos materiales software	55
Tabla 6. Estimación total del presupuesto.....	56
Tabla 7. Catálogo de requisitos funcionales.....	58
Tabla 8. Catálogo de requisitos no funcionales	58
Tabla 9. Catálogo de casos de uso	60
Tabla 10. Modelo de pruebas, metas.....	97
Tabla 11. Modelo de pruebas, preguntas	98
Tabla 12. Catálogo de resultados de las pruebas, metas.....	99
Tabla 13. Catálogo de resultados de las pruebas, preguntas	100
Tabla 14. Plantilla requisitos software funcionales y no funcionales	116
Tabla 15. RSF-01. Menú principal.....	117
Tabla 16. RSF-02. Menú visita virtual.....	117
Tabla 17. RSF-03. Menú visita virtual alternativa.....	117
Tabla 18. RSF-04. Sistema de juego FPS, para las modalidades PC o mando	118
Tabla 19. RSF-05. Sistema de juego FP, para las modalidades que utilizan Leap Motion	118
Tabla 20. RSF-06. Iniciar una nueva visita virtual, con Leap Motion.....	118
Tabla 21. RSF-07. Iniciar una nueva visita virtual, con el Leap Motion y las Oculus Rift	119
Tabla 22. RSF-08. Iniciar una nueva visita virtual alternativa, con PC o mando de consola	119
Tabla 23. RSF-09. Iniciar una nueva visita virtual alternativa, con el Leap Motion	119
Tabla 24. RSF-10. Mostrar escenario Museo Príncipe Felipe.....	120
Tabla 25. RSF-11. Desplazamiento personaje básico	120
Tabla 26. RSF-12. Desplazamiento personaje avanzado.....	120
Tabla 27. RSF-13. ESC sales de la pantalla de juego.....	121
Tabla 28. RSF-14. ESC Oculus Rift sales del juego	121
Tabla 29. RSNF-01. Compatibilidad con resoluciones.....	121
Tabla 30. RSNF-02. Ficheros ejecutables.....	122
Tabla 31. RSNF-03. Música y efectos de sonido.....	122
Tabla 32. Plantilla casos de uso.....	123
Tabla 33. CU-01. Visita virtual	124
Tabla 34. CU-02. Visita virtual alternativa.....	124
Tabla 35. CU-03. Salir de la aplicación	124
Tabla 36. CU-04. Visita virtual con Leap Motion.....	124
Tabla 37. CU-05. Visita virtual con Leap Motion y Oculus Rift.....	125
Tabla 38. CU-06. Cargar menú principal desde el menú visita virtual	125
Tabla 39. CU-07. Visita virtual alternativa con PC o mando.....	125

Tabla 40. CU-08. Visita virtual alternativa con Leap Motion	126
Tabla 41. CU-09. Cargar menú principal desde el menú visita virtual alternativa	126
Tabla 42. CU-10. Mover cámara.....	126
Tabla 43. CU-11. Mover personaje.....	127
Tabla 44. CU-12. Disparar.....	127
Tabla 45. CU-13. Salir de la pantalla de juego, al menú principal con ESC	127
Tabla 46. CU-14. Salir del juego con ESC para las Oculus Rift	127
Tabla 47. Coste real recursos humanos	135
Tabla 48. Coste real recursos materiales hardware.....	135
Tabla 49. Coste real recursos materiales software	135
Tabla 50. Coste real total del presupuesto	136
Tabla 51. Calculo de ganancias/perdidas del proyecto.....	136

Índice de ilustraciones

Illustration 1. Application architecture	6
Illustration 2. Principe Felipe Science museum. Outer side view scheme	7
Illustration 3. Main menú design	7
Illustration 4. Leap motion HUD design	7
Illustration 5. Principe Felipe Science museum. Southest facade details	7
Illustration 6. Principe Felipe Science museum. Southest facade details	8
Illustration 7. Southwest facade Blender model. Front view.....	8
Illustration 8. Southwest facade Blender model. Side view.....	8
Illustration 9. Southeast facade Blender model.....	8
Illustration 10. Northeast facade Blender model.....	8
Illustration 11. Counter blender model.....	9
Illustration 12. Mapping process in Blender	9
Illustration 13. Window's glass in Blender	10
Illustration 14. Final initial menu	10
Illustration 15. Characters for the virtual tour with Leap Motion	11
Illustration 16. Interaction with objects, Touched lamp.....	12
Illustration 17. Characters for the virtual tour with Leap Motion and Oculus Rift	13
Ilustración 18. Museo del Prado	27
Ilustración 19. Museo Nacional Centro de Arte Reina Sofía	28
Ilustración 20. Museo del traje de Madrid.....	28
Ilustración 21. Museo Sorolla.....	28
Ilustración 22. Juego "Nought & crosses" de Alexander S. Douglas	29
Ilustración 23. Magnavox Odyssey, primera videoconsola de sobremesa	29
Ilustración 24. Videjuego "Computer Space" de Nolan Bushnell.....	29
Ilustración 25. Videojuego "Pong" de Al Alcom	30
Ilustración 26. Videojuego "Shinobi"	30
Ilustración 27. Videojuego "Doom".....	30
Ilustración 28. Izquierda, videoconsola"Game & Watch". Derecha, videoconsola"Nintedo DS"31	
Ilustración 29. Logo de Blender.....	33
Ilustración 30. Interfaz Blender	33
Ilustración 31. Logo Autodesk.....	34
Ilustración 32. Interfaz 3ds Max.....	34
Ilustración 33. Interfaz Maya	35
Ilustración 34. Logo Unity 3D	38
Ilustración 35. Unreal Engine logo	39
Ilustración 36. CryEngine logo.....	40
Ilustración 37. Leap Motion	¡Error! Marcador no definido.
Ilustración 38. Intel Real Sense	43
Ilustración 39. Nimble VR.....	44

Ilustración 40. Usuario realizando pruebas con un sistema BCI	44
Ilustración 41. Oculus Rift	45
Ilustración 42. 13th Lab.....	46
Ilustración 43. Tecnologías utilizadas.....	48
Ilustración 44. Planificación estimada. 1 periodo equivale a 2 días	53
Ilustración 45. Arquitectura de la aplicación	60
Ilustración 46. Museo de las ciencias Príncipe Felipe. Esquema exterior visto desde un lateral	62
Ilustración 47. Museo de las ciencias Príncipe Felipe. Esquema exterior visto desde arriba	62
Ilustración 48. Museo de las ciencias Príncipe Felipe. Detalle de la fachada suroeste	62
Ilustración 49. Museo de las ciencias Príncipe Felipe. Detalle corredor de la fachada suroeste	63
Ilustración 50. Museo de las ciencias Príncipe Felipe. Detalle de la fachada noreste	63
Ilustración 51. Museo de las ciencias Príncipe Felipe. Esquema interior visto desde un lateral	64
Ilustración 52. Museo de las ciencias Príncipe Felipe. Plano interior	64
Ilustración 53. Diseño menú principal.....	65
Ilustración 54. Diseño menú visita virtual.....	65
Ilustración 55. Diseño menú visita virtual alternativa	66
Ilustración 56. Diseño menú derrota	66
Ilustración 57. Diseño HUD PC o mando	67
Ilustración 58. Diseño HUD Leap Motion.....	68
Ilustración 59. Museo de las ciencias Príncipe Felipe. Detalle de la fachada suroeste	69
Ilustración 60. Museo de las ciencias Príncipe Felipe. Detalle de la fachada suroeste	70
Ilustración 61. Modelado en Blender de la fachada suroeste, vista frontal	70
Ilustración 62. Modelado en Blender de la fachada suroeste, vista lateral.....	70
Ilustración 63. Museo de las ciencias Príncipe Felipe. Detalle de la fachada noreste	71
Ilustración 64. Museo de las ciencias Príncipe Felipe. Detalle de la fachada noreste	71
Ilustración 65. Modelado en Blender de la fachada noreste, vista frontal.....	71
Ilustración 66. Modelado en Blender de la fachada noreste, vista lateral	71
Ilustración 67. Museo de las ciencias Príncipe Felipe. Detalle de la fachada lateral	72
Ilustración 68. Modelado en Blender de la fachada lateral, vista frontal.....	72
Ilustración 69. Modelado en Blender de la fachada lateral, vista frontal simplificada.....	72
Ilustración 70. Modelado en Blender, fachada suroeste	73
Ilustración 71. Modelado en Blender, fachada noreste.....	73
Ilustración 72. Modelado en Blender del tejado.....	73
Ilustración 73. Modelado en Blender, cuatro plantas interiores del edificio	74
Ilustración 74. Modelado en Blender, fachada sureste finalizada	74
Ilustración 75. Modelado en Blender, fachada noreste finalizada	75
Ilustración 76. Modelado en Blender, logo.....	75
Ilustración 77. Modelado en Blender, caja fuerte	75
Ilustración 78. Modelado en Blender, lámpara mesa	76
Ilustración 79. Modelado en Blender, pc.....	76
Ilustración 80. Modelado en Blender, proyector.....	77
Ilustración 81. Modelado en Blender, escultura.....	78

Ilustración 82. Modelado en Blender, taza	78
Ilustración 83. Modelado en Blender, tetera	78
Ilustración 84. Modelado en Blender, lámpara techo	79
Ilustración 85. Modelado en Blender, mostrador.....	79
Ilustración 86. Modelado con Blender, pistola	80
Ilustración 87. Modelado con Blender, vida	80
Ilustración 88. Modelado con Blender, munición	80
Ilustración 89. Modelado en Blender, proceso de mapping	81
Ilustración 90. Modelado en Blender, cristales.....	82
Ilustración 91. Importación de modelos en Unity, aplicación de texturas	83
Ilustración 92. Proceso de generación de mapa de normales con CrazyBump	84
Ilustración 93. Unity creación de bosques	85
Ilustración 94. Árbol básico de Unity	85
Ilustración 95. Bosque.....	85
Ilustración 96. Enemigo para la visita virtual alternativa	86
Ilustración 97. Menú inicial final	87
Ilustración 98. Menú visita virtual final.....	87
Ilustración 99. Menú visita virtual alternativa final	88
Ilustración 100. Menú derrota final	88
Ilustración 101. Iluminación de ambiente.....	89
Ilustración 102. Iluminación exterior	89
Ilustración 103. Iluminación exterior propiedades	89
Ilustración 104. Iluminación interior, lámpara de techo.....	89
Ilustración 105. Iluminación interior, propiedades.....	89
Ilustración 106. Personaje para la visita virtual con PC o mando	91
Ilustración 107. Personaje para la visita virtual con Leap Motion	92
Ilustración 108. Interacción con objetos, silla.....	94
Ilustración 109. Interacción con objetos, lámpara antes de tocarla.....	94
Ilustración 110. Interacción con objetos, lámpara después de tocarla	94
Ilustración 111. Personaje para la visita virtual con Leap Motion y Oculus Rift	96
Ilustración 112. Código sistema PEGI	102
Ilustración 113. Regulación de la música utilizada.....	103
Ilustración 114. Distribución de las empresas de videojuegos en la geografía española	104
Ilustración 115. Gestión de datos	128
Ilustración 116. Interfaz Better DS3, selección tipo de archivo	130
Ilustración 117. Interfaz Better DS3, configuración de acciones	131
Ilustración 118. Interfaz Better DS3, asociar mando	132
Ilustración 119. Planificación real. 1 periodo equivale a 2 días	134

1. Introducción

La mayor parte de la información de hoy en día reside en las páginas web, permitiendo un acceso sencillo y rápido a la información. En ellas se representa un mundo bidimensional, a diferencia del mundo real que es tridimensional. De aquí nace el interés por integrar una nueva dimensión, que nos ofrezca una mayor inmersión en el mundo virtual. [1]

Todo ello se encuentra dentro de un importante campo de las Ciencias de la Computación, denominado Realidad Virtual que tiene importantes aplicaciones en la educación, para estimular el proceso de aprendizaje. Las aplicaciones de realidad virtual consiguen un efecto llamado inmersión, según el cual los usuarios pueden interactuar completamente con el ambiente artificial utilizando los sentidos del tacto, el oído, y la vista mediante dispositivos especiales que están conectados al ordenador, como el Leap Motion, Oculus Rift o Brain Computer Interface. Estos aparatos tienen sensores que detectan el movimiento de forma precisa, repercutiendo en el mundo virtual.

- Leap Motion es un pequeño dispositivo de control gestual que se colocó frente a la pantalla de nuestro ordenador, es capaz de capturar con una precisión enorme los movimientos de nuestras manos, dedos e incluso objetos.
- Oculus Rift gafas de realidad virtual que están siendo desarrolladas por Oculus VR.
- Brain Computer Interface constituye una tecnología que se basa en la adquisición de ondas cerebrales para luego ser procesadas e interpretadas por una máquina u ordenador. Establecen el camino para interactuar con el exterior mediante nuestro pensamiento ya que estas interfaces permiten transformarlos en acciones reales en nuestro entorno.

El término de realidad virtual suele asociarse a casi todo aquello que tiene que ver con imágenes en tres dimensiones generadas por ordenador y con la interacción de los usuarios con este ambiente gráfico. Ello supone la existencia de un complejo sistema electrónico para proyectar espacios virtuales en 3D y para enviar y recibir señales con información sobre la actuación del usuario, quien, con un sistema de este tipo, puede sentir que se encuentra inmerso en un mundo virtual.

La realidad virtual explota todas las técnicas de reproducción de imágenes y las extiende, usándolas dentro del entorno en el que el usuario puede examinar, manipular e interactuar con los objetos expuestos. Un mundo virtual es un modelo matemático que describe un “espacio tridimensional”, dentro de este espacio están contenidos objetos que pueden representar cualquier cosa, desde una simple entidad geométrica, por ejemplo un cubo o una esfera, hasta una forma compleja, como puede ser un desarrollo arquitectónico. Se trata, en definitiva, de un paso más allá de lo que sería la simulación

por ordenador, tratándose realmente de la simulación interactiva, dinámica y en tiempo real de un sistema.

En este trabajo se intenta destacar que el desarrollo de una visita virtual parte de la misma base que el de un videojuego, y que ambos pertenecen al mismo campo. Por ello, de aquí en adelante se hablará de videojuego para referirse a la parte del trabajo que se orienta a videojuegos. También, al referirse a visita virtual se habla de la parte del proyecto que está relacionada con la visita virtual del museo. El desarrollo de la última une diferentes aspectos, digitales y no digitales. Así, la parte no digital engloba:

- Selección de la tecnología.
- Bocetos de la creación de los personajes, carácter de esos personajes, los objetos que los identifican.
- Bocetos de los edificios, que compondrán el escenario.
- Diseño de los diferentes niveles de la visita virtual, creando para cada uno de ellos su escenario.
- Diseño del HUD.
- Selección de música y sonidos.

Por otra parte, encontramos los aspectos tecnológicos:

- Formas de interactuar con la visita virtual.
- Diseño de los elementos de la visita virtual (escenario, edificios, personajes, objetos, etc), así como la creación de texturas que se les aplicaran.
- Comportamiento de los personajes, mediante la aplicación de IA.
- Selección de la música y de los distintos sonidos que se reproducirán a lo largo de la visita.

Con todo ello se podría realizar una visita completa de cualquier edificio, desde cualquier parte del mundo. Por ejemplo de un museo, en el cual alumnos de un colegio podrían realizar una visita desde cualquier parte del mundo. Entre los museos más importantes de España, como son: Museo del Prado, Museo Nacional Centro de Arte Reina Sofía o el Museo de las Artes y las Ciencias Príncipe Felipe, no hay ninguno que cuente con una visita virtual. Algunos de ellos cuentan con recursos interactivos, pero que no tienen nada que ver, por ejemplo el museo del prado cuenta con videos interactivos, en los que puedes detener el video y ver la información de un cuadro, otro ejemplo es el Reina Sofía, en el que mediante Google Art Project se pueden ver algunas de sus obras.

Este proyecto nace precisamente de todo esto, para cubrir las carencias existentes en este ámbito y suplir esas carencias mediante la realización de una visita virtual.

Los objetivos que se persiguen con la elaboración de este proyecto son:

- Estudiar las etapas de desarrollo de una visita virtual, partiendo de las posibilidades existentes en la actualidad para la creación de un videojuego.
- Desarrollar una visita virtual de un museo desde cero.
- Desarrollo de una visita virtual desde cero, para demostrar que como se ha indicado ambos parten de la misma base.
- Incorporación de nuevas formas de interacción entre el usuario y el sistema, para ello se incorporará la interacción mediante:
 - Leap Motion → dispositivo de entrada para que el usuario se mueva por el escenario e interactúe con los objetos de la visita, para ello se asociarán acciones realizadas por el usuario con acciones en el escenario.
- Incorporación de nuevas formas de realidad virtual, que den al usuario la sensación de estar inmerso en el museo recreado.
 - Oculus Rift → dispositivo de salida para que el usuario vea el escenario como si estuviera allí mismo.

Todos los objetivos mencionados se han ido realizando paso a paso, y están documentados detalladamente más adelante. Durante su realización se han ido encontrando algunas dificultades debido al uso de diferentes programas y lenguajes de programación, aunque todo ello se ha resuelto satisfactoriamente. El resultado final se puede ver en el siguiente video: <https://www.youtube.com/watch?v=P8t9yj6OXfl>

Tras su desarrollo se han realizado algunas pruebas con usuarios, para comprobar que todo funcionase correctamente y además obtener información sobre qué otras cosas les gustaría encontrarse en futuras versiones de la aplicación. De este pequeño estudio se han obtenido interesantes resultados: como la propuesta del renderizado por partes o el giro analógico. Además tras la finalización del proyecto se han visto que se pueden realizar diversos trabajos futuros partiendo del proyecto actual, no ha sido posible la implementación de ninguno de ellos debido al tiempo que hay asignado a este proyecto: aumentar el abanico de formas de interacción, añadir cinemáticas o sistema multijugador.

Estructura del documento

El resto del documento está estructurado de la siguiente manera:

Estado del arte. Introducción a la historia de los videojuegos, a las formas de interacción entre el usuario y el sistema, tanto actuales como nuevas, estado actual de la tecnología.

Definición formal del problema. Explicación de forma más concreta de cómo se va a desarrollar el proyecto.

Planificación y presupuesto. Detalle de las distintas fases de las que consta el proyecto, identificando las tareas y el tiempo estimado, al igual que una estimación del presupuesto.

Desarrollo. Engloba tres fases, el análisis, el diseño y la implementación, en todas se ha descrito detalladamente todos los pasos que se han realizado.

Pruebas. Descripción de las pruebas a las que se someterá el producto y análisis de los resultados.

Marco regulador. Explicación de las normativas para la creación del proyecto.

Entorno socio-económico. Impacto social de los videojuegos en la actualidad.

Conclusiones. Exposición del éxito o fracaso de los objetivos planteados, incluyendo el análisis de la planificación y presupuesto, y los posibles trabajos futuros.

Bibliografía

Anexos. Existirán varios anexos al documento, que será información adicional incluida, estos anexos son: acrónimos, glosario de términos, metodología de desarrollo Scrum, requisitos del sistema, casos de uso, gestión de datos, resultados de las pruebas realizadas, manual de usuario, y por último análisis de la planificación y coste del proyecto tras su finalización.

2. Estado del arte

En este apartado se realiza un estudio de la situación actual de las visitas virtuales en los museos, un resumen de la industria de los videojuegos así como un análisis de su situación actual, todo ello para ayudar al lector a valorar el trabajo realizado y que se puedan apreciar claramente los aspectos más destacables del proyecto.

Visitas virtuales en los museos de España

La gran mayoría de los museos de España no cuentan con una visita virtual, y los que cuentan con ella no lo hacen con una visita libre, en la mayoría te mueves con imágenes, pero el usuario no interactúa de forma directa con el entorno, ya que en todas ellas estas muy limitado. [2] A continuación se muestran algunos de estos museos:

Museo del Prado: cuenta con videos interactivos, en los que situando el ratón encima del video, el video se detiene, mostrándote puntos de información, sobre los cuadros que se ven en ese momento en la imagen, si pulsas sobre alguna opción, puedes ver la información de ese cuadro. En la Ilustración 18 se puede ver una captura de este video. [3]

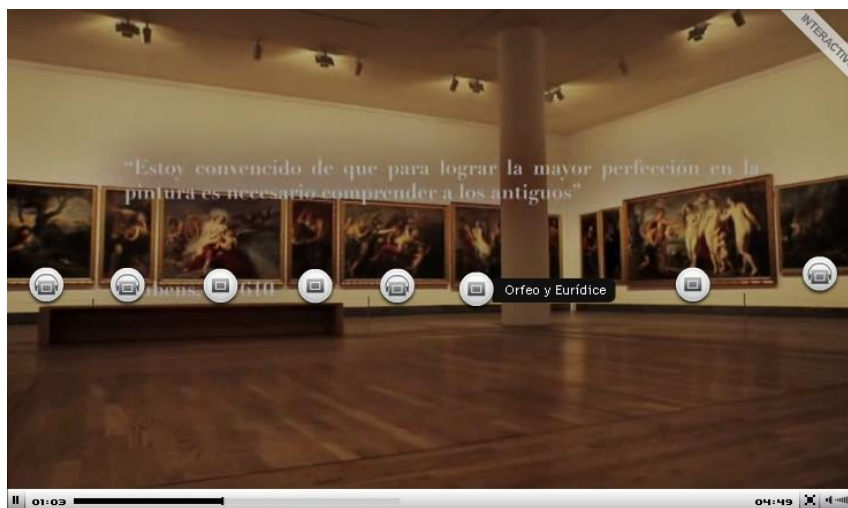


Ilustración 18. Museo del Prado

Museo Nacional Centro de Arte Reina Sofía: en el que mediante Google Art Project (web que presenta una recopilación de imágenes en alta resolución de obras de arte expuestas en varios museos del mundo) se pueden visualizar con gran detalle algunas de sus obras expuestas. En la Ilustración 19 se pueden ver algunas de las obras disponibles. [4]



Ilustración 19. Museo Nacional Centro de Arte Reina Sofia

Museo del traje de Madrid: este museo nos permite visitarlo mediante imágenes que podemos ir pasando una a una o mediante un mapa en el que podemos seleccionar el punto en el que nos queremos situar. En la Ilustración 20 se puede ver el mapa del museo, y la vista de uno de ellos. [5]

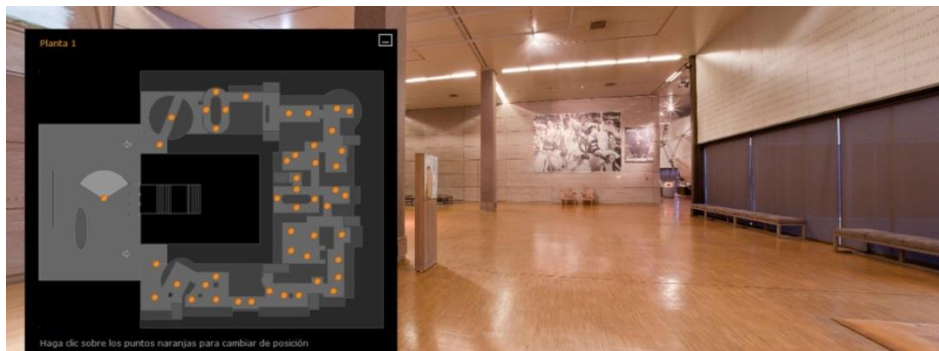


Ilustración 20. Museo del traje de Madrid

Museo Sorolla de Madrid: este museo es el que cuenta con una visita virtual algo más realista, ya que mediante una barra con opciones podemos ir moviendo la cámara hacia donde nosotros queramos, cuenta con unas flechas en el suelo de la imagen que si se pulsaran permiten irse moviendo por el museo. En la Ilustración 21 se puede ver una imagen de este museo, y la barra de flechas anteriormente indicada. [6]

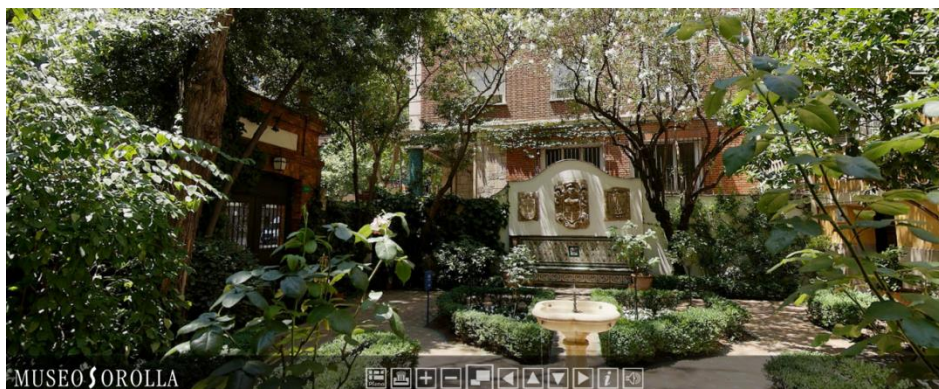


Ilustración 21. Museo Sorolla

En este apartado se han mostrado diferentes tipos de visitas que existen en la actualidad, el resto de museos si cuentan con visitas virtuales son similares a las anteriormente descritas.

Industria de los videojuegos

No se ha realizado un resumen de la industria de las visitas virtuales, ya que no existe información, debido a que además de ser un campo muy reciente, no deja de ser un videojuego con diferente tipo de acciones, por lo que ambos pertenecen a la misma industria, por este motivo a continuación se ha realizado un breve estudio de cómo ha sido la evolución de esta industria. [7]

La creación de una visita virtual, lleva asociado el mismo trabajo que la creación de un videojuego, ya que en su esencia es lo mismo, aunque en el caso de una visita virtual la parte de IA no existe. En este punto vamos a hacer un pequeño recorrido por la industria de los videojuegos desde sus inicios hasta la actualidad.

Los videojuegos, como todo lo ligado a la tecnología han tenido una gran evolución en muy pocos años, donde se ha pasado de apenas unos rudimentarios medios a disponer de una tecnología capaz de recrear realidades virtuales asombrosas. Para llegar hasta este punto ha sido necesaria una gran cantidad de dinero, que es lo que han gastado las grandes editoras en crear esta apasionante industria, al igual que una gran creatividad para crear dispositivos que pudieran ejecutar aplicaciones gráficas. A continuación se realiza un repaso de los grandes hitos por los que han pasado las videoconsolas:

1952: El primer videojuego de la historia se atribuye a Alexander S. Douglas, que creó una versión computarizada de las 3 en raya “Nought & crosses”, Ilustración 22.



Ilustración 22. Juego "Nought & crosses" de Alexander S. Douglas

1966: Ralph Baer comenzó con “Fox & Hounds” lo que acabaría siendo Odyssey, el primer sistema domestico de videojuegos lanzado en 1972, Ilustración 23.



Ilustración 23. Magnavox Odyssey, primera videoconsola de sobremesa

1971: Nolan Bushnell, comercializa el proyecto “Computer Space”, el primer juego arcade de la historia, Ilustración 24.



Ilustración 24. Videjuego "Computer Space" de Nolan Bushnell

1972: Al Alcom crea “Pong”, ilustración 25, la piedra angular del videojuego como industria. Apareciendo detrás “Space Invaders” (Taito), “Asteroirds” (Atari) y sistemas de entretenimiento como Atari 2600.

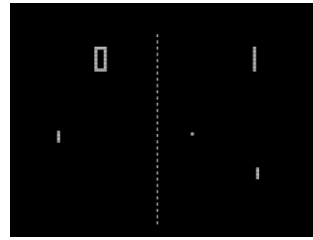


Ilustración 25. Videojuego "Pong" de Al Alcom

1980-89: “La década de los 8 bits”

Es la época del crecimiento de los soportes donde aparecen las primeras videoconsolas.

En 1983 se lanza la videoconsola NES (Nintendo Entertainment System). Y a finales de los 80 comienzan a aparecer las primeras consolas de 16 bits como Megadrive de Sega.

En esta época triunfan los videojuegos arcade como Pacman (Namco), Battle Zone (Atari), Pole Position (Namco), Tron (Mindway), Zaxxon (Sega). También aparece uno de los grandes clásicos, Tetris de Alexey Pajitnov.

Un poco más adelante en 1985 aparece “Super Mario Bros”, cambiando el curso de la historia. Este es el primer videojuego que rompe con la estructura de bucle de pantalla tan sólo basada en obtener una buena puntuación. Creando un nuevo juego al margen de los puzles, los juegos de plataformas.



Ilustración 26. Videojuego "Shinobi"

Otro videojuego de esta época que a destacar es “Shinobi”, ilustración 26, lanzado en 1987 fue un juego que introdujo varias novedades en el sistema clásico de plataformas, como una IA mucho más sofisticada para los enemigos.

1990-99: “La revolución de las 3D”

Empiezan a aparecer nuevos géneros de videojuegos, entre ellos los primeros desarrollo que buscaban un entorno tridimensional. Se publican “Wolfestein” y “Doom” (Ilustración 27) para PC, los primeros videojuegos que intentan emular un entorno “tridimensional”. Apareciendo con ellos un nuevo género, FPS (First Person Shooter).



Ilustración 27. Videojuego "Doom"

También para PC, aparecen los primeros videojuegos íntegros en 3D como “4D Boxing”, y mezcla de 3D en tiempo real y prerenderizado como “Alone In The Dark”.

La industria de inmediato adopta los videojuegos 3D creando la “generación de 32 bits” con las consolas Play Station de Sony y Saturn de Sega. Y la “generación de 64 bits” con Nintendo 64 y Atari Jaguar.

La popularidad de Internet y su accesibilidad, facilitaron los videojuegos multijugador vía online, creándose importantes comunidades denominadas a menudo “Clanes”. Siendo los responsables de la creación de un nuevo género, MMORPG (Massive Multiplayer Online Role-Playing Games).

En 1998 aparece una nueva consola, Dreamcast de Sega, iniciándose con ella la “generación de 128 bits”.

2000: “Nuevo siglo”

Esta década marca el nacimiento de los videojuegos como industria, donde se supera a la industria del cine en cuanto a facturación en punto de venta, aunque el cine consigue más beneficios por los negocios asociados como son el DVD o el merchandising.

En el año 2000 Sony lanza la PlayStation 2, la consola más importante hasta la fecha en la industria de los videojuegos con más de 150 millones de unidades vendidas.

En 2001 Microsoft lanza XBOX entrando en la industria de las consolas. Además, se centra en el desarrollo de videojuegos para su consola, como Halo, una de las mejores adaptaciones de FPS a consola.

Seguidamente, Nintendo lanza GameCube y GameBoy Advance.

En 2004 Nintendo lanza Nintendo DS, sucesora de la exitosa consola GameBoy Advance con retro-compatibilidad y aportando novedades importantes al sistema de control, mediante una pantalla táctil y una segunda pantalla normal. La segunda pantalla no suponía ninguna novedad, ya que fue utilizada por Nintendo en 1980 con la Game & Watch, aunque para el público sí que lo supuso. En la Ilustración 28, se ve la diferencia entre las dos videoconsolas con dos pantallas anteriormente mencionadas.



Ilustración 28. Izquierda, videoconsola “Game & Watch”. Derecha, videoconsola “Nintendo DS”

En el mismo año, Sony lanza PSP (Play-Station-Portable), una consola portátil, con capacidad de reproducción de películas, música, fotos y navegación web como características principales.

En 2005 Microsoft lanza XBOX 360, que da comienzo a la séptima generación de consolas.

En el año 2006 Sony lanza la PlayStation3 compitiendo con XBOX 360.

En ese mismo año aparece Nintendo Wii que al igual que NDS, ofrece un sistema de control fuera de lo habitual, que mediante acelerómetros y sensores, detecta el movimiento que produce el usuario.

2011: “Internet”

Lo destacable de esta generación es el uso de internet como eje central de la funcionalidad de las consolas, juntando en un único aparato, las funciones de consola de juegos y venta de películas, series de TV y otros contenidos desde el propio aparato. Las consolas de sobremesa características de esta etapa son la Wii U, Play Station 4 y la XBOX One.

Diseño 3D

El diseño 3D es el campo de la informática dedicado a la creación de modelos tridimensionales generados por ordenador. Es uno de los campos con mayor evolución en los últimos años, a la par que evoluciona la tecnología informática, el diseño 3D alcanza mayor definición y capacidad de procesamiento, lo que da lugar a resultados cada vez más realistas.

El diseño tridimensional cada vez se aplica en más ámbitos, y ya no está únicamente restringido a videojuegos y cine. La medicina lo usa para recrear modelos de huesos que ayuden en las operaciones, en sistemas de vigilancia se utiliza para crear gráficos en tiempo real que analizan actividades forestales, marinas, o urbanas, en el campo del derecho, es una prueba valiosa para recreaciones de delitos etc.

Todos los videojuegos se componen de dos fases básicas: modelado y motor de juegos.

Modelado

El modelado en 3D es el proceso por el cual se crean personajes, objetos y escenas que se utilizan tanto para los videojuegos como para el cine. Representa una de las fases vitales del desarrollo de videojuegos y la que más tiempo y recursos requiere.

A continuación se muestra un listado de las herramientas de modelado 3D. Existen muchas herramientas de este tipo y que cumplirían los requisitos para realizar este proyecto, pero sólo se mostrarán las más populares, que poseen mayores recursos en la red, ejemplos, resultados y menor curva de aprendizaje.

Blender

Blender apareció como producto en agosto de 1994, integrando una serie de herramientas para la creación de una amplia gama de contenidos 2D y 3D. Blender ofrece un amplio espectro de funcionalidad para el modelado, texturizado, iluminación, animación y post-procesado de vídeo en un paquete. Por medio de su arquitectura abierta, ofrece interoperabilidad entre plataformas, extensibilidad, ocupa un espacio increíblemente pequeño en disco duro, y un flujo de trabajo altamente integrado. Blender es una de las aplicaciones de código abierto de gráficos 3D más populares del mundo. En la Ilustración 30 se puede ver como se distribuye la información por la pantalla de la aplicación. [8] [9]



Ilustración 29. Logo de Blender

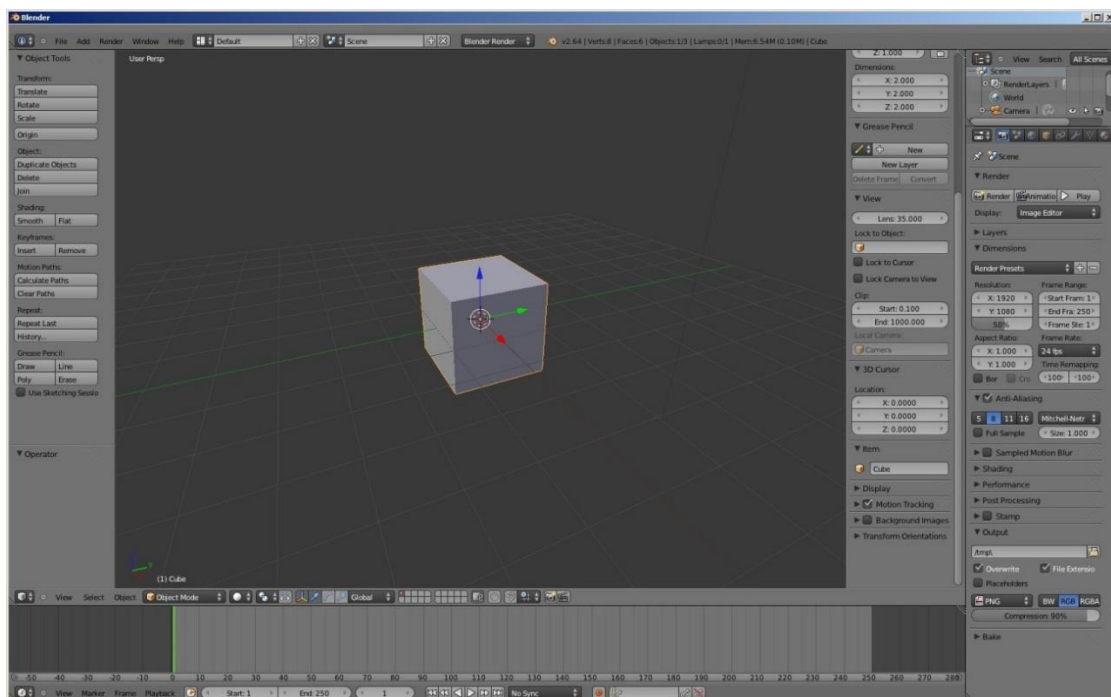


Ilustración 30. Interfaz Blender

3ds Max

Autodesk 3ds Max es un programa de creación de gráficos y animación 3D desarrollado por Autodesk, salió a la venta por primera vez en 1990. [10]



Ilustración 31. Logo Autodesk

3ds Max, con su arquitectura basada en plugins, es uno de los programas de animación 3D más utilizado, especialmente para la creación de videojuegos, anuncios de televisión, en arquitectura o en películas.

La curva de aprendizaje de este software es mucho más elevada que la del programa anterior, si bien sus resultados son muy superiores.

En cuanto a la licencia, 3ds tiene una versión de prueba de 30 días, una versión libre para estudiantes para uso no comercial y para el resto es necesario una licencia de pago de 3.900 €.

En la Ilustración 30 se puede ver como se distribuye la información por la pantalla de la aplicación.

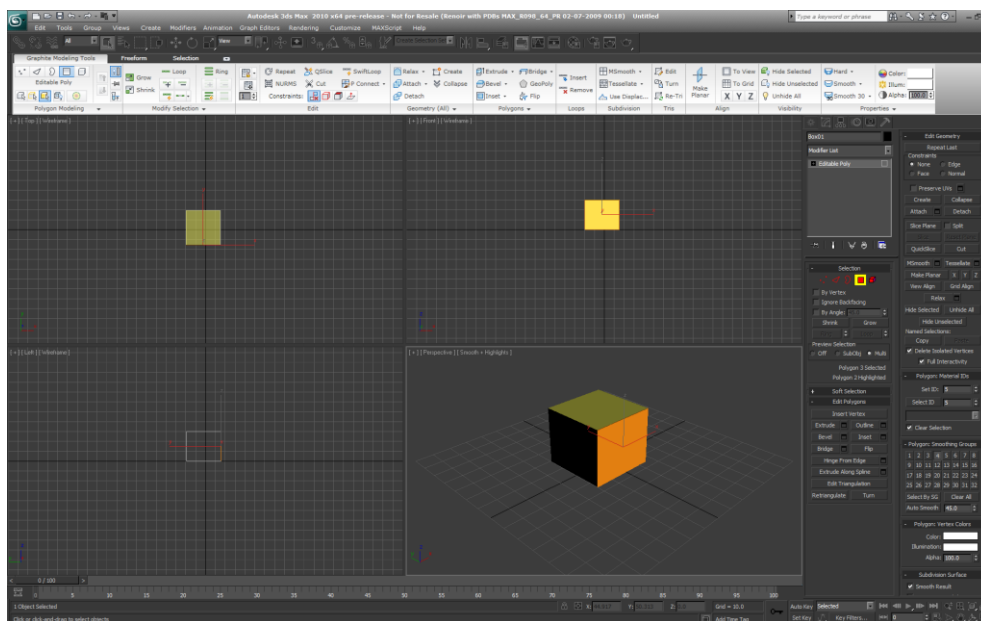


Ilustración 32. Interfaz 3ds Max

Maya

Maya creado también por Autodesk, es un software de animación en 3D que proporciona un conjunto completo de funciones creativas para realizar animaciones, modelados, simulaciones y renderizaciones en 3D por ordenador en una plataforma de producción sumamente ampliable. Maya proporciona conjuntos de herramientas de personajes y efectos de gran calidad así como un aumento de la productividad del modelado, la texturización y las tareas de creación de sombras. [11]

La licencia de Maya es la misma que de 3ds Max.

En la Ilustración 33 se puede ver como se distribuye la información por la pantalla de la aplicación.

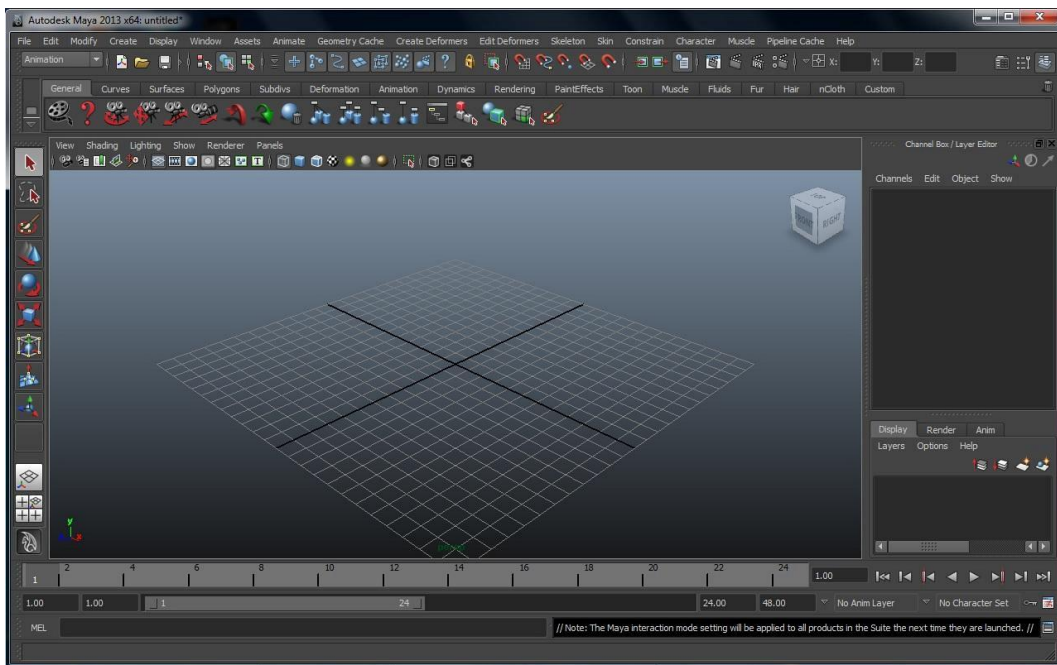


Ilustración 33. Interfaz Maya

En la Tabla 1 se puede ver una comparativa de las tres herramientas de modelado analizadas.

Comparación de las herramientas

<i>Herramienta</i>	<i>Blender</i>	<i>3ds Max</i>	<i>Maya</i>
<i>Desarrollador</i>	Fundación Blender	Autodesk	Autodesk
<i>Web</i>	Web blender	Web 3ds max	Web maya
<i>Licencia</i>	GPL	Software Propietario	Propietario
<i>Sistema operativo</i>	Multiplataforma	Windows 7 y 8	Windows, Mac OS X y Linux
<i>Tamaño</i>	224 MB	4,5 GB	4 GB
<i>RAM</i>	2 GB	4 GB	4 GB
<i>Precio</i>	Gratuita	Libre para estudiantes Prueba 30 días Pago 3.900 €	Libre para estudiantes Prueba 30 días Pago 3.900 €
<i>Experiencia personal previa</i>	Si	No	No

Tabla 1. Comparativa programas de modelado

Motores de juego

Un motor de videojuego es un término que hace referencia a una serie de rutinas de programación que permiten el diseño, la creación y la representación de un videojuego. Del mismo modo existen motores de juegos que operan tanto en consolas de videojuegos como en sistemas operativos. La funcionalidad básica de un motor es renderizar, mostrar gráficos en pantalla, tanto para 2D como 3D.

El problema de la mayoría de estos motores es que no soportan desarrollos para consola, lo que les deja fuera del mercado fuerte de videojuegos, pero como el proyecto que se va a realizar es para pc, esto no supone un problema. El proceso de desarrollo de un videojuego puede variar notablemente por reutilizar o adaptar un mismo motor de videojuego para crear diferentes juegos.

Dos ejemplos de estos *engines* 3D son Irrlicht Engine que es muy espectacular y el famoso Ogre 3D que tiene la gran ventaja de que mucha gente está desarrollando en él, por lo hay foros con mucha actividad.

Adicionalmente y a medida que tanto el hardware como el software ha ido evolucionando, se han ido incorporando componentes que ahora también resultan esenciales para desarrollar un videojuego, como puede ser el motor de físicas, detector de colisiones, sonidos, scripting, animación, el sistema de iluminación dinámica, la inteligencia artificial de los enemigos, redes, streaming, administración de memoria y un escenario gráfico.

Gracias a esto la tecnología de los videojuegos ha evolucionado enormemente y una de las grandes mejoras son los Game Engine, no son motores 3D para visualizar, sino que son un “motor” para crear un juego fácilmente, con muchos ajustes preestablecidos y con todo lo necesario para crear un juego integrado: físicas, IA, audio, etc. Los Game Engine hacen uso de librerías específicas para el tratamiento de gráficos. Las más

utilizadas por los desarrolladores son OpenGL y DirectX, que son las encargadas de comunicarse con la GPU del ordenador y ejecutar los shader para que compute polígonos, de esta forma ofrecen al desarrollador una serie de componentes para editar el juego a su gusto con interfaces intuitivas y sencillas. Además les evitan tener que utilizar otros programas para la realización de físicas o inteligencia artificial.

Los motores de juegos actuales normalmente tienen los siguientes componentes:

- **Motor gráfico:** Realiza funciones de renderizado 2D/3D. Mapea texturas, genera mallas 3D, gestiona el número de polígonos de la escena y los manipula.
- **Física,** existen dos medios para simular la física, ambos necesarios:
 - **Motor de física:** una de los componentes más importantes del motor de juegos, es la simulación de la física en la programación es solo una aproximación cercana a la física real (si bien se acerca mucho a la física que tendría un objeto en realidad, no es igual de realista que en la realidad). Aspectos como la gravedad, velocidad, fuerza o masa, no serían posibles sin este motor.
 - **Detector de colisiones:** el sistema está constantemente actualizando la posición del personaje, como el personaje y los objetos poseen un área de colisión, si el área del personaje se encuentra en la misma zona que el área de un objeto, el personaje se “choca”.
- **Motor de inteligencia artificial:** permite diseñar y programar toda la inteligencia de los personajes, es el encargado de que cuando nos acercamos a un enemigo este reaccione, y esto produzca que se dirija hacia nosotros, corra, ataque, o lo que se le haya indicado que haga.
- **Motor de sonido:** reproduce, gestiona y controla todos los sonidos, músicas y efectos que incluyan en el juego, el motor de sonido es el aspecto que menos ha evolucionado en la historia de los videojuegos, se programa de la misma manera que hace veinte años. La mayoría de los juegos actuales poseen una música de fondo, y efectos cuando se realizan determinadas acciones, como por ejemplo disparar y que se escucha un disparo.
- **Componente de redes:** encargado de la sincronización en línea.

Actualmente los tres motores de desarrollo de juegos más importantes son:

- Unity 3D
- Unreal Engine
- CryEngine

Unity 3D

Entre los Game Engine disponibles hay uno que se ha erigido por encima de todos, ese es Unity 3D, un motor tremendamente potente y flexible que permite desarrollar multiplataforma, PC, iPhone, Wii... e incluso el desarrollo de videojuegos para poder jugar en web, gracias a un pequeño plug-in para el navegador que permite jugar en 3D con buena calidad, sin necesidad de tener el juego instalado. [12]



Ilustración 34. Logo Unity 3D

El motor tiene una gran documentación y con una comunidad de desarrolladores enorme con la que poder compartir información. Además el programa es gratuito, también tiene una licencia comercial de pago, pero si nuestro proyecto lo regalamos el coste del Game Engine es 0 €.

Unity 3D es uno de los motores más versátiles que existe para iniciarse en el desarrollo de videojuegos.

Características que le diferencian de los otros dos motores de juego:

- Es la que posee mayor comunidad en internet y la interfaz más amigable e intuitiva.
- Soportar la publicación para muchas más plataformas que sus competidores, prácticamente todas las del mercado.
- Flexibilidad de lenguajes que es capaz de manejar (UDK, utiliza UnrealScript; CryEngine, LUA/C++).
- La versión gratuita no da acceso a todas las herramientas, para tenerlo todo es necesario adquirir la versión pro, o comprar módulos extra.
- Unity soporta 3 lenguajes:
 - *UnityScript*, variación de JavaScript y se encuentra como este último nombrado en el propio Unity.
 - *C#*.
 - *Boo*. derivado de Python.

Unreal Engine

Unreal Engine es un motor de juego de PC y consolas creados por la compañía Epic Games. Implementado inicialmente en el shooter en primera persona Unreal, siendo la base de juegos como BioShock, o Mass Effect. También se ha utilizado en otros géneros como el rol y juegos de perspectiva en tercera persona. Está escrito en Unreal Script (Lenguaje propio similar Java o C# modificado), siendo compatible con varias plataformas como PC (Microsoft Windows, GNU/Linux), Apple Macintosh (Mac OS, Mac OS X) y la mayoría de consolas (Dreamcast, Gamecube, Wii, Xbox, Xbox 360, PlayStation 2, PlayStation 3, Xbox One, PlayStation 4). [13]



Ilustración 35. Unreal Engine logo

Características que le diferencian de los otros dos motores de juego:

- Sistema de renderizado multihilo, con sistema de sombreado, iluminación, oclusión y gestor de sombras dinámicas.
- Unreal Kismet permite crear scripts de manera gráfica, a modo de elementos en un plano que se interrelacionan entre sí. Aplicado solo al nivel al que están asociados, permiten descargar parte del trabajo de código y hacerlo mucho más interactivo y fácil.
- Unreal Script, lenguaje de programación de alto nivel orientado a objetos propio similar Java o C# modificado.

CryEngine

CryEngine es un motor de juego creado por la empresa alemana desarrolladora de software Crytek, originalmente un motor de demostración para la empresa Nvidia, que al demostrar un gran potencial se implementa por primera vez en el videojuego Far Cry, desarrollado por la misma empresa creadora del motor. El 30 de marzo de 2006, la totalidad de los derechos de CryEngine son adquiridos por la distribuidora de videojuegos Ubisoft. [14]

Características que le diferencian de los otros motores de juego:

- Editor de animación facial.
- Mayor velocidad de renderizado y procesado.
- Motor de físicas más avanzado que el UDK.
- Lenguaje LUA, lenguaje de programación imperativo, estructurado y bastante ligero.



Ilustración 36. CryEngine logo

En la Tabla 2 se puede ver una comparativa de las tres herramientas de motores de juego analizadas.

Comparación de las herramientas

Herramienta	Unity 3D	Unreal Engine	CryEngine
Desarrollador	Unity Technologies	Epic Games	Crytek
Web	Web Unity 3D	Web Unreal Engine	Web CryEngine
Licencia	Propietario	Propietario	Propietario
Sistema operativo	Windows XP, 7 y 8 Mac OS X	Windows 7 64-bit Mac OS X	Windows
Tamaño RAM	4,5 GB	-	-
Precio	Versión gratuita Pago versión Pro o Prueba 30 días o 1500 \$	Libre para estudiantes Pago 19 \$ / mes	Reducida 9,90 \$ / mes Completa, consultar con la empresa
Experiencia personal previa	No	No	No
Lenguaje Script	Unity Script, C# y Boo	Propio	LUA
Compatible con Leap Motion	Si	Si	Si
Compatible con Oculus Rift	Si	Si	Si

Tabla 2. Comparativa motores de juegos

Texturización

El aspecto que más afecta a un videojuego en este terreno, es el de repetición y variedad de matices. Tanto si hacemos un escenario grande, como si se trata de algo más pequeño, nos solemos encontrar con el problema de que se repiten mucho las texturas y todo parece igual. En el caso de los videojuegos tenemos que convivir con el problema de memoria y rendimiento, que limita la cantidad de texturas que podemos utilizar. [15]

Hay que asegurarse no solo de hacer texturas atractivas, sino de preparar suficientes texturas que nos permitan ir intercambiando y conseguir que parezca un escenario creado con miles de ellas.

Tipos de texturas

Hay diferentes tipos de texturas en videojuegos con las que podemos conseguir un aspecto acorde a la iluminación y ambientación que se busca. La textura base sobre la que se consiguen y basan las demás es la de color o diffuse. A partir de esta se crean otras texturas, que combinadas todas entre sí, forman el modelo final. Estas texturas son:

- Diffuse o color: Es la textura básica. La que le va a dar el color al objeto. No tiene ningún efecto especial sobre el modelo.
- Specular o brillo: Es la textura que va a crear brillo sobre nuestro modelo. Se suele utilizar la textura de diffuse para crear el specular. Para que el motor gráfico detecte las partes que brillan, deberemos trabajar en blanco y negro, siendo blanco puro como brillo máximo y negro brillo cero.
- Normal: Esta textura en concreto se creó por la necesidad de optimizar rendimiento en los videojuegos, ya que para conseguir un detalle considerable en los modelos había que crear mucha geometría que hoy por hoy se suele evitar. Gracias a ellas, podemos generar relieves en los ejes X, Y y Z, pudiéndonos ahorrar cientos y cientos de polígonos.
- Bump: este tipo de textura existía antes que las normales, y se sigue usando, porque para ciertas imágenes funciona muy bien y consume muy poco. Solamente hace el efecto de relieve en un eje. También en blanco y negro, siendo blanco el punto más elevado y negro superficie plana.

Una herramienta que nos permite realizar todas estas tareas es CrazyBumps, aplicación para crear *Bumpmaps* (mapas de relieve) de manera más rápida y sencilla.

Tras el motor de juego, el siguiente aspecto más importante es la elección del programa para realizar el modelado, tenemos que tener en cuenta una cosa muy importante, elegir un programa de modelado compatible con el motor elegido, esto lo vamos a ver a continuación.

Interacción

Actuales formas de interactuar con los videojuegos

Actualmente existen diversas formas de interactuar con los videojuegos a continuación vamos a ver algunas de ellas. [16]

Teclado: Fue la primera forma de interacción y sigue siendo la favorita de muchos jugadores. La combinación de teclado y ratón es la mejor para muchos géneros, como por ejemplo para los juegos de estrategia y los FPS.

Mandos: Son bastante cómodos para controlar los juegos ya que los últimos modelos son todos inalámbricos. Además no sólo sirven para las consolas sino que también se pueden utilizar para jugar a mandos del ordenador..

Pantalla táctil (Nintendo DS): La pantalla inferior de la Nintendo DS emplea tecnología táctil, lo que ofrece una experiencia muy realista. Permite dibujar imágenes, controlar la acción en pantalla con gran precisión, o navegar por los menús con un ligero toque.

Detección de movimientos por infrarrojos (Wiimote): A diferencia de un mando que detecta la luz de una pantalla de televisión, el Wiimote detecta la luz de la Barra sensor de la consola, lo que permite el uso coherente, independientemente del tipo o tamaño de la televisión. Esta barra mide aproximadamente 20 cm de longitud y cuenta con diez LED infrarrojos, con cinco LED dispuestos en cada extremo de la barra. La posición y seguimiento del movimiento del Wii Remote permite al jugador imitar las acciones reales de juego, como blandir una espada o una pistola con objetivo, en lugar de simplemente pulsando los botones.

Nuevas formas de interactuar con los videojuegos que ya son una realidad

En la actualidad se están desarrollando nuevas formas de interacción con los videojuegos, algunas de ellas ya disponibles, aunque no en sus versiones definitivas. Estos nuevos sistemas nos permiten una mayor inmersión en los videojuegos, proporcionándonos una experiencia mucho más directa y real.

Leap Motion [17] [18]

Leap Motion, es un pequeño dispositivo de control gestual que se coloca frente a la pantalla de nuestro ordenador, es capaz de capturar con una precisión enorme los movimientos de nuestras manos, dedos e incluso objetos. Es como un Kinect pero con mayor funcionalidad,

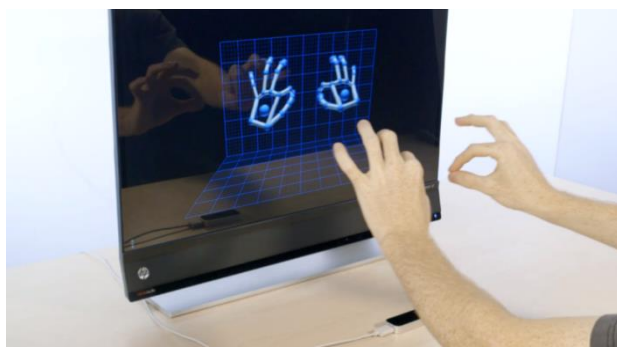


Ilustración 37. Leap Motion

ya que no solo podemos jugar a un shooter en primera persona, sino que podemos incluso escribir en el aire para detectar cada matiz de nuestra caligrafía, navegar por una web, dibujar, movernos por mapas o disfrutar de toda clase de videojuegos. Se distribuyó por primera vez en 2013.

Intel Real Sense [19]

Esta cámara, Ilustración 38, permite a los usuarios interactuar con sus dispositivos con una extraordinaria naturalidad. Se empezó a distribuir a partir del 2014.



Ilustración 38. Intel Real Sense

Funciones de la cámara:

- Interacción natural: reconocimiento facial, de gestos y de voz permitirá que la interacción con los sistemas sea más natural.
- Colaboración inmersiva: por ejemplo en las videoconferencias centrar la atención en el usuario mediante la sustracción de fondo para que la colaboración sea más auténtica.
- Juegos: sustitución de los movimientos W-A-S-D, permitiendo, por ejemplo, a los jugadores utilizar un golpe de karate en el aire para derribar a su adversario ninja de manera que el juego sea más intuitivo y divertido.

Nimble VR [20]

Manos y control por gestos en entornos de realidad virtual. Nace en 2012 por la falta de un sistema que ayude a aumentar la inmersión y la sensación de realismo al usar las Oculus Rift. Ponerte las gafas para entrar en un mundo virtual capaz de engañar a tu cerebro haciéndole creer que eso que está viendo es real y no verte las manos sigue siendo una sensación extraña y Nimble VR entra para trabajar en ese campo. El sistema que proponen es muy parecido al Leap Motion, visto anteriormente. En la Ilustración 39, se puede ver este dispositivo.



Ilustración 39. Nimble VR

Brain Computer Interface [21]

Las interfaces cerebro-computador, en inglés Brain Computer Interfaces (BCI) Ilustración 40, constituyen una tecnología que se basa en la adquisición de ondas cerebrales para luego ser procesadas e interpretadas por una máquina u ordenador. Establecen el camino para interactuar con el exterior mediante nuestro pensamiento ya que estas interfaces permiten transformarlos en acciones reales en nuestro entorno.



Ilustración 40. Usuario realizando pruebas con un sistema BCI

Esta tecnología ha atraído a muchos investigadores durante la última década con el motivo de desarrollar una interfaz cerebro-ordenador eficiente y fiable, lo cual ha supuesto que cada uno de los diferentes grupos de investigación creara su propio "modus operandi" respecto a la tecnología de sus dispositivos. A pesar de las diferencias entre los diferentes dispositivos todos siguen el mismo principio de funcionamiento básico: la medición de la actividad cerebral mediante sensores, procesado de la señal adquirida para obtener sus características de interés.

Las principales ventajas de esta tecnología son por una parte instaurar un canal natural de interacción entre el hombre y la máquina y, por otra, el acceso a toda la información cognitiva y emocional humana.

Realidad virtual

La realidad virtual es un entorno de escenas u objetos de apariencia real, generado mediante tecnología informática, que crea en el usuario la sensación de estar inmerso en él. Dicho entorno es contemplado por el usuario a través de un dispositivo conocido como gafas o casco de realidad virtual. Este puede ir acompañado de otros dispositivos, como guantes o trajes especiales, que permiten una mayor interacción con el entorno así como la percepción de diferentes estímulos que intensifican la sensación de realidad.

Oculus Rift [22] [23]

Oculus Rift, Ilustración 41, son unas gafas de realidad virtual que están siendo desarrolladas por Oculus VR. Durante su periodo como compañía independiente, Oculus VR ha invertido 91 millones de dólares para el desarrollo de Oculus Rift, realizando los primeros envíos del producto en 2013.

Es un sistema de inmersión audiovisual, es uno de los aspectos con mayor potencial de progresión para los próximos años. Más que la sustitución de las pantallas convencionales, parece que cada vez será más habitual el uso complementario de nuevas tecnologías, para percibir un entorno con profundidad y volumen. Cuentan con un creciente apoyo por parte de los desarrolladores y auguran una nueva experiencia de inmersión total. Lo que está claro es que la nueva era en los videojuegos está a la vuelta de la esquina y estamos en primera fila para vivirlo.

La versión para el consumidor está prevista para 2015.

Además se estudia el empleo de estas gafas para muchas otras aplicaciones, como su uso en terapia, permitiendo a personas mayores que no puedan salir de casa, evadirse de la realidad a través de ellas.



Ilustración 41. Oculus Rift

Esta compañía ha sido comprada recientemente por Facebook, además ha comprado dos empresas más relacionadas con la interacción y la realidad virtual:

- **Nimble VR**, nueva forma de interacción, explicada en el punto anterior.
- **13th Lab**: donde la cámara lo es todo. Tal y como ellos mismos explican, han desarrollado una tecnología basándose en la idea de que la cámara es el sensor más importante para entender nuestro entorno e interactuar con él. Han conseguido desde su fundación en 2010, obtener un entorno de reconstrucción 3D en tiempo real muy preciso capaz de captar modelos en tres dimensiones de objetos reales para procesarlos y modificarlos, se puede ver en la Ilustración 42. En Oculus aseguran que esto abre las puertas a todo tipo de nuevas aplicaciones y experiencias, como por ejemplo visitar las pirámides de Egipto o el coliseo Romano en un entorno de realidad virtual a escala 1:1. Esto suena ideal para Oculus al disponer de la posibilidad de obtener modelos en 3D de cualquier objeto real de forma rápida y precisa, lo cual elimina parte del proceso de tener que recrear esos mismos objetos de forma artificial.



Ilustración 42. 13th Lab

Estos tres proyectos se encuentran aún en desarrollo y no se pueden adquirir.

El último fichaje de Facebook en este ámbito es el experto en captura de movimientos Chris Bregler.

Conclusiones del análisis

Tras haber estudiado las distintas alternativas existentes a la hora de diseñar un videojuego, vamos a seleccionar las más adecuadas para la elaboración del presente proyecto.

Motor de juego

La elección del motor de juego, es una decisión importante, ya que es donde vamos a desarrollar la parte más importante.

De los tres motores propuestos, dado que todos cumplen con las necesidades indispensables, que son la compatibilidad con Leap Motion y Oculus rift, se ha elegido Unity 3D, ya que de los tres es el que posee una curva de aprendizaje menor, y dado que no se tiene ninguna experiencia anterior con ningún tipo de motor de juego. Además utiliza JavaScript y C#, de los que ya se tienen conocimientos previos.

Otra característica interesante es que es compatible con numerosas plataformas, y cuenta con una buena comunidad de usuarios.

Modelado

La elección entre un programa de modelado u otro en principio no es una decisión crítica, ya que todos poseen formatos de exportación e importación compatibles entre sí, además todos ellos permiten la aplicación de mapeado y texturizado de manera nativa. Las tres herramientas ofrecen, al nivel que vamos a darle uso, la misma funcionalidad.

De las tres herramientas propuestas se ha elegido Blender, porque es la única herramienta de la cual se tenían conocimientos previos. Además es compatible con los requisitos de importación del motor de juego Unity 3D.

Interacción

De las actuales formas de interacción a las que todos los usuarios tienen acceso, en este proyecto se van a utilizar dos controladores: dispositivo de entrada estandar PC , teclado y ratón (en adelante nombrado como PC), y un mando de videoconsola, en este caso se va a probar con el de la PS3.

De las nuevas formas de interacción, se va a utilizar Leap Motion, ya que es el controlador del que se dispone en la universidad, además de que ofrece una interacción muy realista con el entorno, debido a que se va a comportar igual que los movimientos de las manos del usuario.

Realidad virtual

De las herramientas existentes para la visualización de entornos de apariencia real, se va a utilizar Oculus Rift, ya que se dispone de unas en la universidad, además de que es un dispositivo que nos ofrece las características buscadas para que la simulación al usuario le parezca la más real posible.

Como alternativa, se va a permitir también el uso de la pantalla del PC.

En la Ilustración 43 se muestra a modo de esquema la relación existente entre las tecnologías seleccionadas.

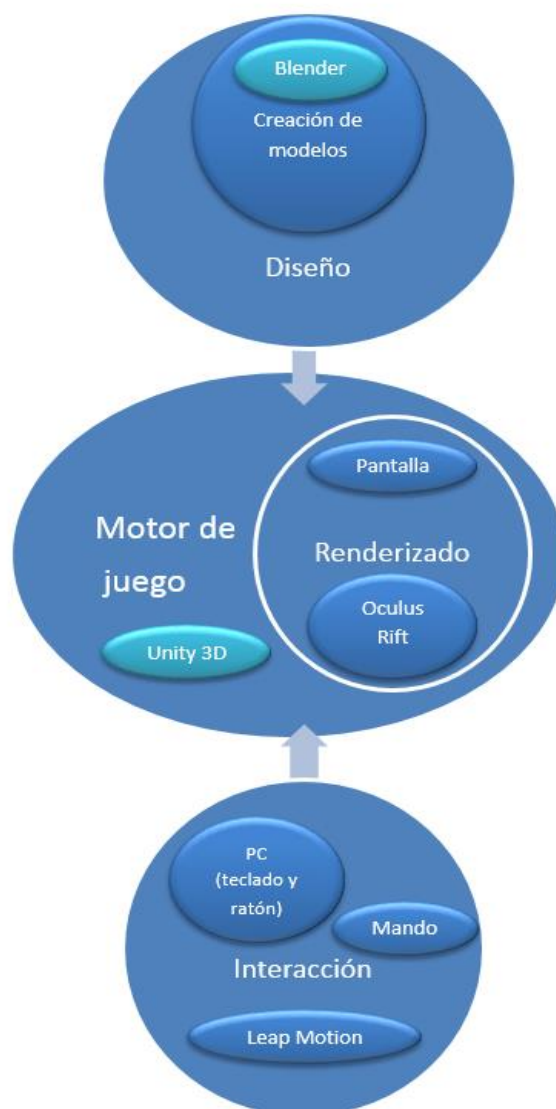


Ilustración 43. Tecnologías utilizadas

Cada una de estas herramientas se utilizará para las siguientes funciones:

Blender: herramienta de diseño gráfico, usada para la creación de los distintos modelos utilizados, del edificio principal y todos los objetos: arma, decoración, etc.

Unity3D: motor de juego, usado para desarrollar todo el juego:

- Creación del escenario utilizando todos los diseños creados con Blender, así como el diseño del interior del edificio.
- Integración con las distintas formas de interacción.
- Creación de menús.
- IA.

PC (teclado y ratón): Dispositivos básicos para la interacción con el sistema.

Mando: Dispositivo más avanzado para la interacción con el sistema.

Leap Motion: Dispositivo de última generación para la interacción de control gestual.

Pantalla: Dispositivo básico de salida para el renderizado.

Oculus Rift: Dispositivo de salida de última generación para el renderizado.

3. Definición formal del problema

Tras el análisis de las diferentes tecnologías con las que contamos para la elaboración del proyecto, a continuación se va a explicar de forma más concreta como se va a desarrollar:

Debido a que el proceso de realizar una visita virtual, como ya se ha visto es muy parecido al de realizar un videojuego, se va a utilizar como base la visita virtual, para la creación de un videojuego del tipo FPS. De esta forma el proyecto tendrá una doble finalidad:

- Creación de una visita virtual. Que tendrá dos formas de interacción diferentes:
 - Leap Motion.
 - Leap Motion y Oculus Rift.
- Creación de una visita virtual alternativa, videojuego tipo FPS, con esta creación se pretende demostrar que la creación de una visita virtual y un videojuego parten de la misma base. Que también tendrá dos formas de interacción:
 - PC (teclado y ratón) o mando.
 - Leap Motion.

Modelado

Lo primero que se hará es el modelado de los diseños en 3D, para ello se utilizará la herramienta de Blender. Se van a modelar los siguientes diseños:

- Edificio museo de las ciencias Príncipe Felipe, de este edificio se modelará únicamente la parte exterior. Se ha elegido modelar este edificio ya que su estructura es bastante particular y compleja, debido a que contiene grandes cristalerías repartidas por toda la fachada. Aunque como ventaja presenta partes con la misma estructura.
- Interior del edificio. Se va a diseñar un interior simple, para que no suponga una gran carga a la hora de renderizar el juego, dando la sensación de que nos encontramos en una oficina, para ello se diseñaran mesas, sillas, lámparas, pantallas de proyección, proyectores, ordenadores, etc.
- Exterior del edificio. Al igual que el interior, se creará un exterior también simple, se incluirá un bosque que rodeará al edificio principal.
- Pistola para disparar, para el modo de visita virtual alternativa, cuando se maneje el juego con el PC (teclado y ratón) o el mando.

Motor de juego

El siguiente paso, es incluir todos los modelos creados en el motor de juego, para ello se utilizará Unity 3D. Donde se van a realizar las siguientes tareas:

- Incluir los modelos creados con Blender.
- Texturización de esos modelos, para darles el realismo que requieren.
- Creación de scripts para controlar la interacción con los distintos objetos, creando una visita virtual. Además en la versión alternativa, se añadirán enemigos creando un juego FPS, donde se utilizarán scripts para controlar a los enemigos y la interacción con ellos.
- Creación de menús que permitan al usuario elegir la opción de visita que el desee.
- Se añadirá música a las distintas escenas del juego.

Interacción

Se incorporarán dos formas de interacción actualmente utilizadas:

- PC (teclado y ratón).
- Mando de consola, se utilizará un mando de PS3.

Además se incorporará una forma de interacción muy novedosa:

- LeapMotion: con este controlador, el usuario realiza movimientos con las manos que ve reflejados en el dispositivo de salida, como si de sus propias manos se tratase. Los movimientos realizados por el usuario se asociarán a acciones, que le permitan moverse por el entorno en el que se encuentra, así como interactuar con los objetos que se encuentran a su alrededor.

Realidad virtual

Se incorporará una forma visualización de entornos de apariencia real:

- Oculus Rift: estas gafas permiten que el usuario se sitúe directamente en el entorno que se ha creado, teniendo la sensación de encontrarse allí mismo. Las gafas se utilizarán para sustituir a la pantalla del ordenador, ofreciendo una visita mucho más realista.

Como alternativa a este sistema se podrá utilizar también la pantalla del PC, aunque con este sistema de salida, se pierda la sensación de que el usuario se encuentre allí realmente.

4. Planificación y presupuesto

Planificación

Para el desarrollo del proyecto, se ha desglosado el trabajo en cuatro partes: estudio preliminar, desarrollo, pruebas y documentación. Las tareas a realizar dentro de cada fase son las siguientes:

- Estudio preliminar:
 - Estudio de la industria de los videojuegos.
 - Estudio de las actuales y nuevas formas de interacción.
 - Estudio de herramientas de diseño 3D: motores de juego y modelado.
 - Estudio de actuales y nuevas formas de interacción.
 - Estudio de sistemas de realidad virtual.
- Desarrollo:
 - Análisis:
 - Definición de requisitos.
 - Casos de uso.
 - Diseño
 - Diseño de la aplicación.
 - Diseño del escenario.
 - Diseño de menús.
 - Diseño del HUD.
 - Implementación
 - Modelado con Blender. Museo de las ciencias Príncipe Felipe, creación de objetos para la decoración interior del museo, y otros específicos de la visita virtual alternativa, también se realizarán materiales con sus texturas para todos los modelos.
 - Jugabilidad con Unity. Decoración exterior del museo, interacción de enemigos con el personaje principal, creación de menús, creación de la lógica para la integración de nuevas formas de interacción y realidad virtual, utilizando Leap Motion y Oculus Rift.
- Pruebas realizadas con usuarios.
- Documentación.

Planificación estimada:

Para el proyecto se ha estimado una duración total de 100 días, en la Ilustración 44 se muestra el tiempo que se ha estimado necesario para cada una de las tareas, cada periodo representa 2 días.

Planificación estimada

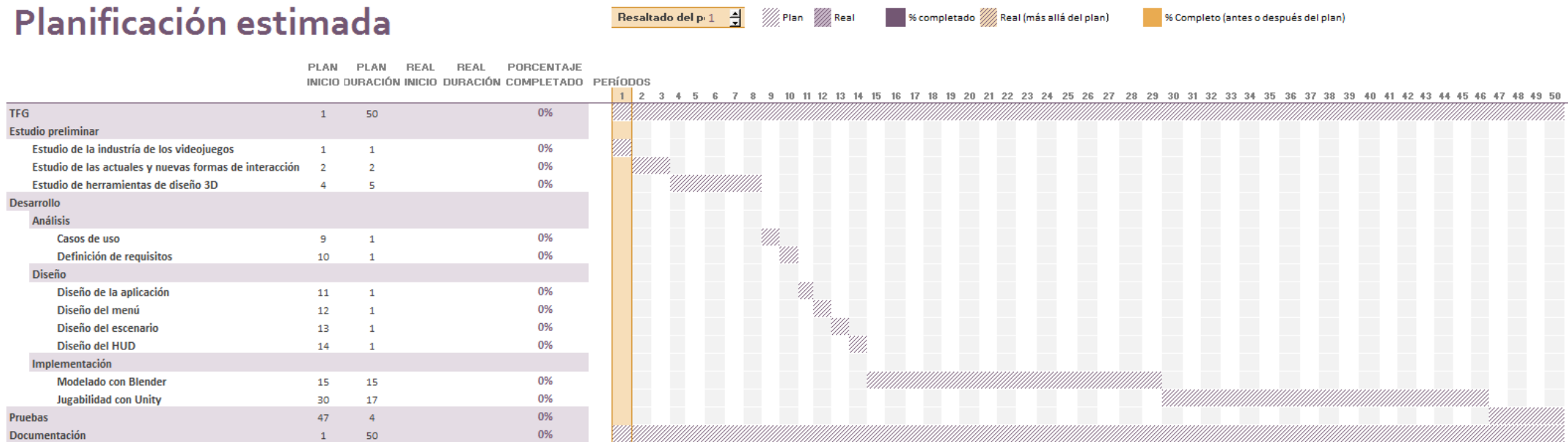


Ilustración 44. Planificación estimada. 1 periodo equivale a 2 días

Metodología de desarrollo

Utilizaremos una metodología de desarrollo ágil llamada Scrum, muy utilizada en empresas de desarrollo de software en general y de videojuegos en particular.

En el [Anexo III. Metodología de desarrollo Scrum](#) se detalla en que consiste esta metodología.

Esta metodología permite que todo el equipo conozca perfectamente en qué estado va el proyecto, qué falta por desarrollar y mostrar lo que ya ha realizado cada uno. Esto se consigue a través de reuniones de control en las que participa todo el equipo del proyecto. Tras una reunión de control, todo el mundo debe tener una idea muy clara del estado del proyecto.

Scrum lo forman diversos conceptos que debemos desarrollar y trabajar:

- Product Backlog: detallado en el apartado [Requisitos software](#).
- Sprint: detallado en el punto anterior [Planificación](#).
- ProductOwner: en nuestro proyecto este es el jefe de proyecto.
- ScrumMaster: el encargado de esta tarea también será el jefe de proyecto.
- Team: nuestro equipo para desarrollar el proyecto está formado por el jefe de proyecto, un analista, un diseñador, un programador y un tester.
- Reunión de planificación: se desarrollan al comienzo de un Sprint.
- Reunión diaria: se realizará una reunión cada mañana, de esta forma todo el equipo se mantiene al tanto del estado y evolución del proyecto.
- Reunión de revisión: se realizará una reunión al finalizar cada Sprint.

Presupuesto

Para el desarrollo del proyecto es necesario disponer de un equipo de trabajo especializado (recursos humanos) que se encargarán de garantizar el correcto seguimiento de las tareas y que la versión definitiva del proyecto se adecua a los requerimientos establecidos. Además, para ello, cada miembro necesitará ciertos equipos físicos (recursos materiales).

Recursos humanos

Para el cálculo del coste por hora, se ha tomado como referencia la media de la mayor y menor tarifa de dichos puestos en el portal infojobs.

Jefe de proyecto: 3298€/mes

Programador: 2500€/mes

Analista: 2160€/mes

Tester: 100€/mes

Diseñador: 2200€/mes

Para realizar una estimación correcta, para cada puesto se ha calculado el coste por día, tomando como base que en un mes se trabajaría 20 días, obteniendo como resultado los costes indicados en la Tabla 3.

<i>Puesto</i>	<i>Dedicación (nº días)</i>	<i>Coste/día</i>	<i>Total</i>
<i>Jefe de proyecto</i>	16	164,45 €	2.631,20 €
<i>Analista</i>	4	108,00 €	432,00 €
<i>Diseñador</i>	8	110,00 €	880,00 €
<i>Programador</i>	64	125,00 €	8.000,00 €
<i>Tester</i>	8	50,00 €	400,00 €
TOTAL	100		12.343,20 €

Tabla 3. Estimación presupuesto recursos humanos

Recursos materiales

Para la realización del proyecto se ha realizado una estimación de los costes materiales, se han dividido en dos categorías hardware y software, Tabla 4 y Tabla 5.

Hardware

<i>Descripción</i>	<i>Coste</i>	<i>% de uso</i>	<i>Dedicación</i>	<i>Depreciación</i>	<i>Coste imputable</i>
<i>PC Intel I7</i>	1.299,00 €	100%	5	60	108,25 €
<i>Mando PS3</i>	57,69 €	100%	1	60	0,96 €
<i>Leap Motion con montura para Oculus Rift</i>	96,99 €	100%	1	60	1,62 €
<i>Oculus Rift</i>	350,00 €	100%	1	60	5,83 €
TOTAL					116,66 €

Tabla 4. Estimación presupuesto recursos materiales hardware

Software

<i>Descripción</i>	<i>Coste</i>	<i>% de uso</i>	<i>Dedicación</i>	<i>Depreciación</i>	<i>Coste imputable</i>
<i>Windows 7</i>	79,00 €	100%	3	60	3,95 €
<i>Microsoft Office</i>	79,00 €	100%	3	60	3,95 €
TOTAL					7,90 €

Tabla 5. Estimación presupuesto recursos materiales software

Habiendo estimado los recursos necesarios para el correcto desarrollo del proyecto, el presupuesto final se desglosa en la Tabla 6.

<i>Descripción</i>	<i>Coste</i>
<i>Recursos humanos</i>	12.343,20 €
<i>Recursos materiales hardware</i>	116,66 €
<i>Recursos materiales software</i>	7,90 €
TOTAL	12.467,76 €
<i>Riesgo (25%)</i>	3.116,94 €
<i>Beneficio (15%)</i>	1.870,16 €
TOTAL sin IVA	17.454,87 €
<i>IVA (21%)</i>	3.665,52 €
TOTAL con IVA	21.120,39 €

Tabla 6. Estimación total del presupuesto

El presupuesto total del proyecto asciende a la cantidad de:

21.120,39 €

Veintiún mil ciento veinte euros con treinta y nueve céntimos.

Leganés, a 3 de Octubre de 2014



Fdo. Jennifer García de la Calle

5. Desarrollo

Análisis

Este apartado tiene como objetivo especificar las características y opciones que deberán tener el proyecto propuesto.

Requisitos del sistema

Los requisitos se van a dividir en dos grupos:

- Requisitos software funcionales: especifica el propósito del software.
- Requisitos software no funcionales: especifica el modo de realizar las tareas y sus requisitos.

Para la definición de los distintos requisitos se han tenido en cuenta los siguientes atributos:

- Identificador: Siglas del tipo de requisito seguido del número de requisito, por orden cronológico de creación.
- Prioridad: Importancia de implantación (baja, media, alta).
- Descripción: Explicación detallada del requisito.

Así, los requisitos recogidos son los que aparecen en el siguiente catálogo, Tabla 7 y Tabla 8. Los detalles se encuentran en el [Anexo IV. Requisitos del sistema](#).

Requisitos software funcionales

<i>Identificador</i>	<i>Nombre</i>	<i>Descripción</i>
<i>RSF-01</i>	Alta	La aplicación deberá tener un menú principal
<i>RSF-02</i>	Alta	La aplicación deberá tener un menú visita virtual
<i>RSF-03</i>	Alta	La aplicación deberá tener un menú de visita virtual alternativa
<i>RSF-04</i>	Alta	El tipo de juego a desarrollar tiene que ser un FPS (First Person Shooter)
<i>RSF-05</i>	Alta	El tipo de juego a desarrollar tiene que ser un FP (First Person)
<i>RSF-06</i>	Alta	Se debe permitir iniciar una nueva visita virtual, con el Leap Motion
<i>RSF-07</i>	Alta	Se debe permitir iniciar una nueva visita virtual, con el Leap Motion y las Oculus Rift, en un ejecutable diferente, específico para Oculus Rift
<i>RSF-08</i>	Alta	Se debe permitir iniciar una nueva visita virtual alternativa, con teclado y ratón o mando de consola
<i>RSF-09</i>	Alta	Se debe permitir iniciar una nueva visita virtual alternativa, con el Leap Motion
<i>RSF-10</i>	Alta	El mapa principal de la visita virtual debe estar

<i>Identificador</i>	<i>Nombre</i>	<i>Descripción</i>
		inspirado en el Museo de las ciencias Príncipe Felipe, de Valencia
<i>RSF-11</i>	Alta	Desplazamiento personaje básico
<i>RSF-12</i>	Alta	Desplazamiento personaje avanzado
<i>RSF-13</i>	Alta	Al pulsar ESC, desde la partida de juego, exceptuando la de Leap Motion y Oculus Rift, se vuelve al menú principal
<i>RSF-14</i>	Alta	Al pulsar ESC, desde la partida de juego de Leap Motion y Oculus Rift, se sale del juego

Tabla 7. Catálogo de requisitos funcionales

Requisitos software no funcionales

<i>Identificador</i>	<i>Prioridad</i>	<i>Descripción</i>
<i>RSNF-01</i>	Alta	La aplicación deberá ser compatible con distintas resoluciones de pantalla
<i>RSNF-02</i>	Alta	La aplicación deberá tener dos ejecutables, uno para la interacción con el Leap Motion y las Oculus Rift, y otro para el resto de interacciones: Leap Motion y PC
<i>RSNF-03</i>	Alta	El sistema deberá utilizar música y efectos de sonido.

Tabla 8. Catálogo de requisitos no funcionales

Casos de uso

Los casos de uso es el modo de identificar las distintas interacciones que un usuario tendrá con la aplicación.

Actores

Este proyecto consta de un único actor, el jugador. Para desempeñar este rol vale cualquier tipo de persona, genero, sexo, edad ya que el juego no presenta ningún tipo de contenido que restrinja su uso.

Para la definición de los distintos requisitos se han tenido en cuenta los siguientes atributos:

- Identificador: Siglas del caso de uso seguido del número de requisito, por orden cronológico de creación.
- Nombre: nombre descriptivo del caso de uso.
- Requisitos: con los que está relacionado el caso de uso.

Para la especificación de los casos de uso se ha creado el siguiente catálogo, Tabla 9, que cubrirán los requisitos acordados en el punto anterior. Los detalles de cada uno de los casos de uso se encuentran en el [Anexo V. Casos de uso](#).

<i>Identificador</i>	<i>Nombre</i>	<i>Requisitos</i>
CU-01	Visita virtual	RSF-01 RSF-02
CU-02	Visita virtual alternativa	RSF-01 RSF-03
CU-03	Salir de la aplicación	RSF-01
CU-04	Visita virtual con Leap Motion	RSF-02 RSF-05 RSF-06
CU-05	Visita virtual con Leap Motion y Oculus Rift	RSF-05 RSF-07
CU-06	Cargar menú principal desde el menú visita virtual	RSF-01 RSF-02
CU-07	Visita alternativa con PC y mando	RSF-01 RSF-03 RSF-08
CU-08	Visita virtual alternativa con Leap Motion	RSF-01 RSF-03 RSF-09
CU-9	Cargar menú principal desde el menú visita virtual alternativa	RSF-01 RSF-03
CU-10	Mover cámara	RSF-04 RSF-05 RSF-06 RSF-07 RSF-08 RSF-09 RSF-11 RSF-12
CU-11	Mover personaje	RSF-04 RSF-05 RSF-06 RSF-07 RSF-08 RSF-09 RSF-11 RSF-12
CU-12	Disparar	RSF-04 RSF-08 RSF-09 RSF-12
CU-13	Salir de la pantalla de juego, al menú principal con ESC	RSF-01 RSF-06 RSF-08 RSF-09
CU-14	Salir del juego con ESC para las Oculus Rift	RSF-7 RSF-11

Identificador	Nombre	Requisitos
		RSF-12
		RSF-13
		RSF-14

Tabla 9. Catálogo de casos de uso

Diseño

En este capítulo se explicara de manera concisa y detallada el diseño del proyecto a desarrollar.

El proyecto consiste en una visita virtual en primera persona con un mundo abierto, donde el jugador puede recorrer un mapa sin seguir un camino. A medida que recorra el mapa, el jugador puede encontrarse con objetos, armas o enemigos a batir.

Arquitectura de la aplicación

La arquitectura de la aplicación permite obtener un diseño a alto nivel del sistema, pudiendo identificar claramente las partes que lo forman y la relación que existente entre ellos. Esta relación se puede ver de forma esquemática en la Ilustración 45.

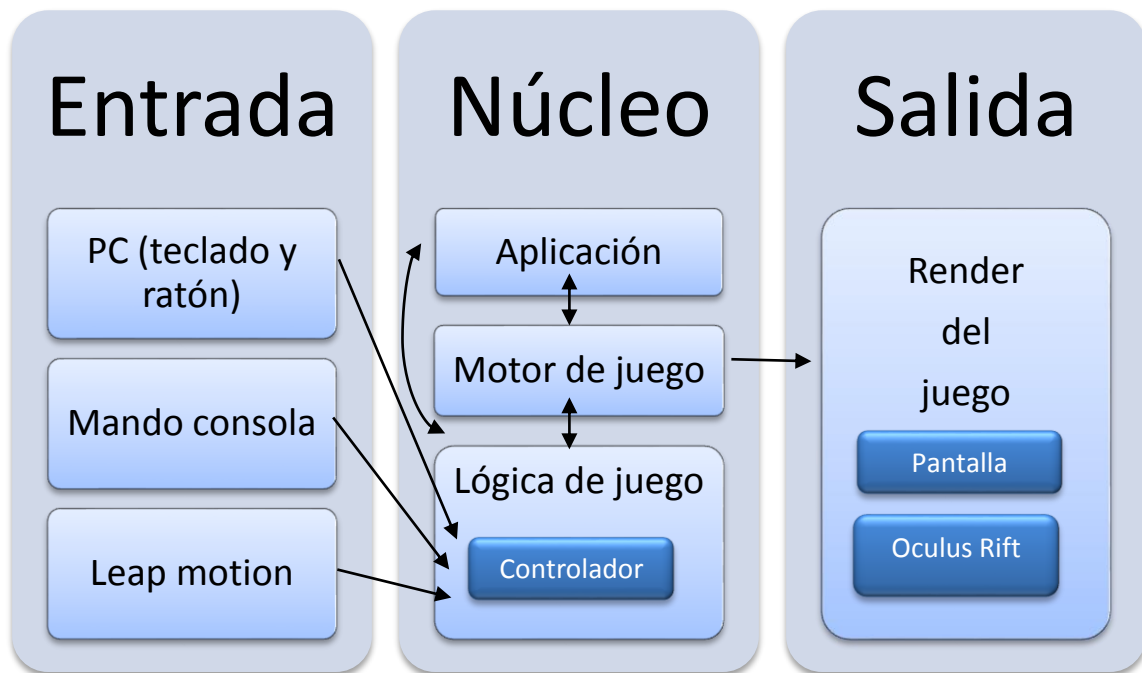


Ilustración 45. Arquitectura de la aplicación

Entrada: encargada de capturar las acciones.

- PC: captura las acciones del jugador.
- Mando consola: captura las acciones del jugador.
- Leap Motion: captura los gestos del jugador

Núcleo: parte más importante de la visita virtual, ya que engloba la aplicación, el motor y la lógica.

- Aplicación: componente principal del proyecto, encargado relacionar los principales componentes y ejecutar la aplicación, recibir los datos de entrada, controlar el motor de juego y la lógica.
- Motor de juego: controla los distintos aspectos del juego, renderización del juego, movimiento de personajes, físicas, colisiones, iluminación, sonidos, control de datos, y de realizar la salida por pantalla (render).
- Lógica de juego: programación que establece las relaciones entre los elementos del juego y controla el comportamiento de los personajes.
 - Controlador: código necesario para interpretar los datos que capturan los dispositivos de entrada y los traduce en movimientos del juego.

Salida: se encarga de mostrar al jugador el renderizado del juego.

- Render de juego: Encargado de la salida, proporciona al usuario una interfaz visual de la aplicación.
 - Pantalla: salida por defecto del ordenador.
 - Oculus Rift: salida, mediante realidad virtual.

La modificación del núcleo de la aplicación es la única que tiene efectos directos sobre el producto final. La modificación de la entrada y la salida afecta a las distintas formas de interactuar y mostrar la información del juego. En caso de querer añadir más formas de interacción sólo habría que enlazarlas con el juego añadiendo nuevos controladores.

Modelado. Escenario

Toda la creación del escenario ha sido realizada desde cero para este proyecto, para lo que se ha utilizado la herramienta Blender.

Se ha elegido el diseño del escenario del Museo de las ciencias Príncipe Felipe de Valencia. Se ha seleccionado este edificio debido a su complejidad, ya que presenta un gran reto:

El Museo de las Ciencias cuenta con más de 42.000 metros cuadrados de superficie construidos, de los cuales 26.000 metros cuadrados son expositivos. El edificio se distribuye en tres plantas. Además de contar con una impresionante superficie acristalada del Museo, con más de 4.000 cristales, y las vistas al jardín del Turia, desde las terrazas exteriores.

A continuación se han recopilados distintas imágenes, para poder modelar el edificio:

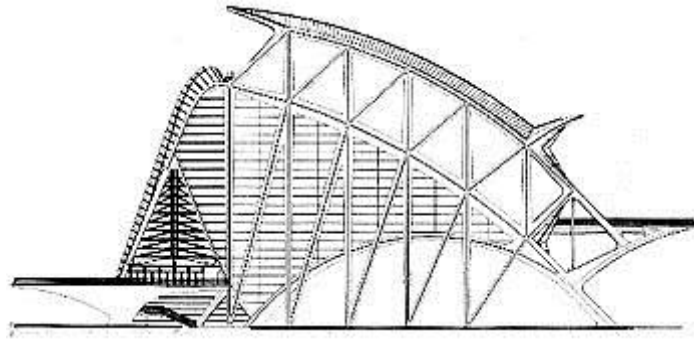


Ilustración 46. Museo de las ciencias Príncipe Felipe. Esquema exterior visto desde un lateral

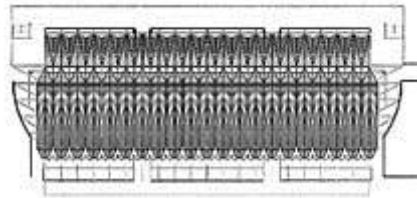


Ilustración 47. Museo de las ciencias Príncipe Felipe. Esquema exterior visto desde arriba



Ilustración 48. Museo de las ciencias Príncipe Felipe. Detalle de la fachada suroeste



Ilustración 49. Museo de las ciencias Príncipe Felipe. Detalle corredor de la fachada suroeste



Ilustración 50. Museo de las ciencias Príncipe Felipe. Detalle de la fachada noreste

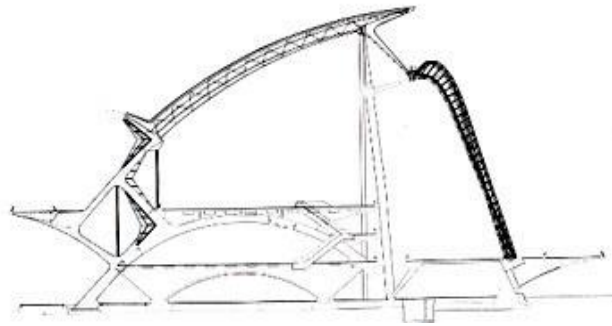


Ilustración 51. Museo de las ciencias Príncipe Felipe. Esquema interior visto desde un lateral

PLANO PLANTA MUSEO DE LAS CIENCIAS PRÍNCIPE FELIPE / FLOOR PLAN OF THE SCIENCE MUSEUM PRÍNCIPE FELIPE

PLANTA TERCERA / THIRD FLOOR

- 1 Sala de talleres / Workshop Room *
Es necesaria la entrada al Museo. / A general entry ticket to the Museum is required.
- 2 Ciencia y deporte con el Valencia CF / Science and sport with Valencia CF
- 3 Memoria helada / Frozen memory
- 4 Cambio climático / Climatic change
- 5 Gravedad cero / Zero gravity
- 6 Academia del Espacio / Space Academy *
Es necesaria la entrada al Museo. / A general entry ticket to the Museum is required.
- 7 Bosque de Cromosomas / Forest of Chromosomes

*Venta de entradas en taquillas. / Tickets on sale in ticket offices.



PLANTA TERCERA
THIRD FLOOR

PLANTA SEGUNDA / SECOND FLOOR

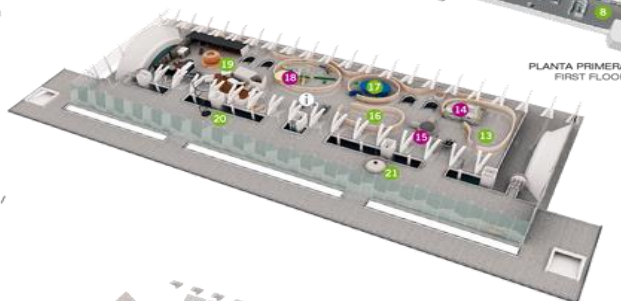
- 8 Un siglo de premios Nobel / A century of Nobel prizes
- 9 Jean Dausset
- 10 Severo Ochoa
- 11 Ramón y Cajal
- 12 Santiago Grisolia



PLANTA SEGUNDA
SECOND FLOOR

PLANTA PRIMERA / FIRST FLOOR

- 13 ¡A toda vela! / Top sail!
- 14 Teatro de la Electricidad / Theatre of Electricity
Consulta de horarios y retirada de entradas en el punto de información de la planta primera. / Consult times and ticket pick-up arrangements at the information point on the first floor.
- 15 Escuela del Cadete del Espacio / Space Cadet School
Público infantil. / Children's audience.
- 16 Cuida, cuidamos tu calidad de vida / Cuida, we care for your quality of life
- 17 Exploratorio / Exploratorium
- 18 L'Espai dels Xiquets
De 4 a 7 años. / For children from 4 to 7.
- 19 Amueblando el hábitat de la mano con la naturaleza / Furnishing the world hand in hand with nature
- 20 Representación artística de la molécula de ADN / Artistic representation of the DNA molecule
- 21 Péndulo de Foucault / Foucault's Pendulum



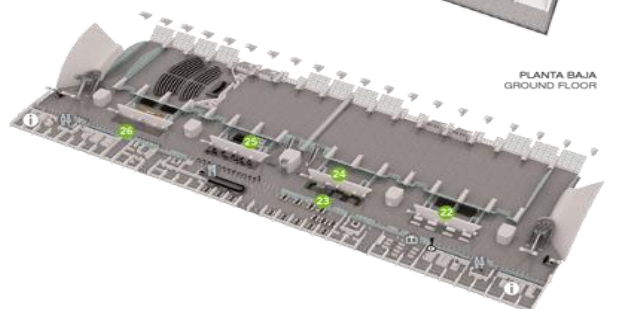
PLANTA PRIMERA
FIRST FLOOR

PLANTA BAJA / GROUND FLOOR

- 22 La ciencia a escena. Salas 1 y 2 *
La ciencia a escena. Sala 3 / Science on stage. Rooms 1 and 2
- 23 Ventana Valencia / Window on Valencia
- 24 La ciencia a escena. Estudio de TV / Science on stage. TV Studio *
- 25 La ciencia a escena. Sala 3 / Science on stage. Room 3 *
- 26 Exposición temporal / Temporary exhibition

*Venta de entradas en taquillas. / Tickets on sale in ticket offices.

Programación sujeta a cambios. / Programme subject to changes.



PLANTA BAJA
GROUND FLOOR

SERVICIOS / SERVICES

- Tienda / Shop
- Aseos / Toilets
- Cafetería-Restaurante / Cafeteria-Restaurant
- Información / Information
- Taquilla / Ticket office

Consulta en los puntos de información y disfruta al máximo de tu visita! / Enquire at the information points and enjoy your visit to the full!

Saborea los menús diarios del restaurante. / Try the restaurant's daily menus.

Ilustración 52. Museo de las ciencias Príncipe Felipe. Plano interior

Motor de juego

Menús

Debido a que la visita virtual tiene distintas formas de jugar, se van a crear unos menús para que el usuario seleccione con que opción de las disponibles desea jugar, exceptuando la interacción con Leap Motion y Oculus Rift, que se encuentran en un ejecutable diferente. A continuación se muestran los bocetos de estos menús.

Cada menú tiene un título para que el usuario sepa donde se encuentra, y botones para seleccionar la opción que desee.

Menú principal

Será el primer menú que se cargue al lanzar el juego, Ilustración 53.

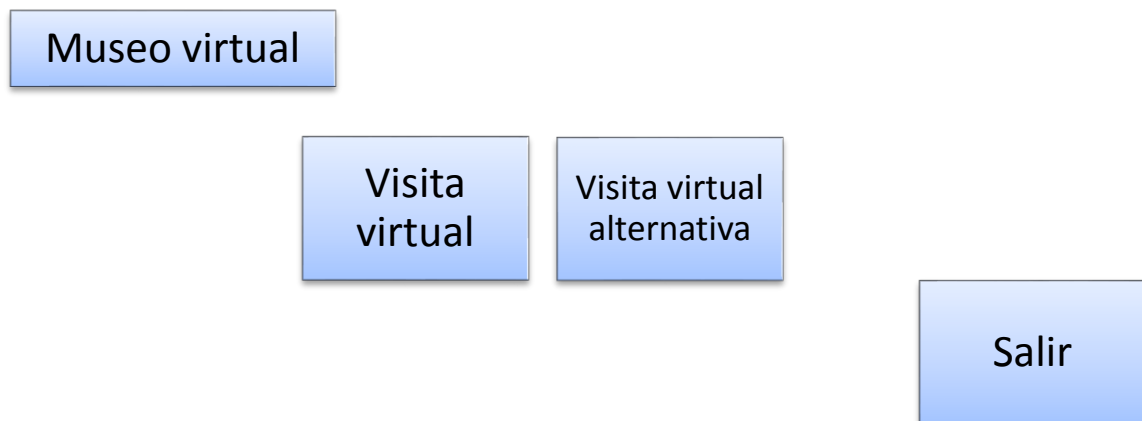


Ilustración 53. Diseño menú principal

Menú visita virtual

Tras seleccionar la opción visita virtual en el menú inicial, se mostrará el menú representado en la Ilustración 54.

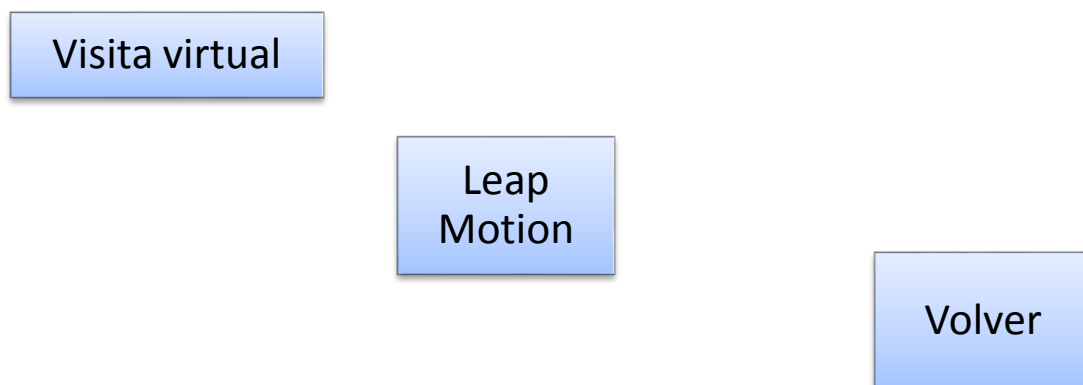


Ilustración 54. Diseño menú visita virtual

Menú visita virtual alternativa

Tras seleccionar el menú visita virtual alternativa, se mostrará el menú representado en la Ilustración 55.

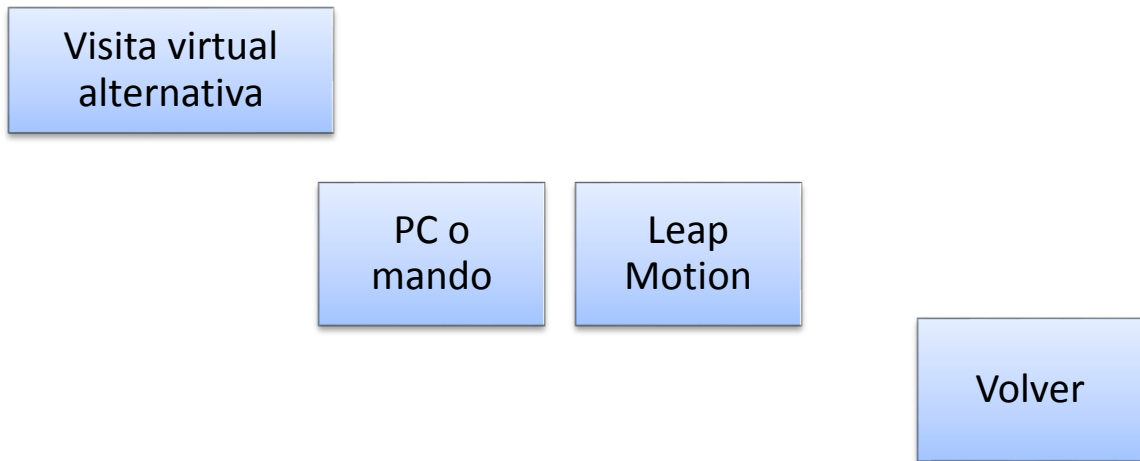


Ilustración 55. Diseño menú visita virtual alternativa

Menú derrota

El menú de la Ilustración 56 se mostrará cuando el jugador haya sido derrotado por los enemigos, tras seleccionar la opción de visita virtual alternativa y cualquiera de sus opciones de interacción.

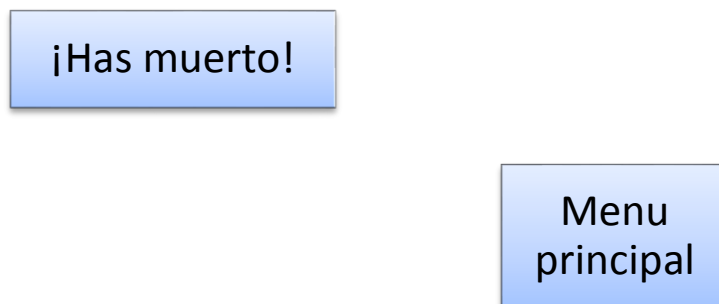


Ilustración 56. Diseño menú derrota

Diseño del HUD (Visor frontal de datos) para la visita virtual alternativa

El HUD (Head Up Display) es la interfaz de comunicación con el jugador durante el juego, todo lo que vemos en pantalla de información: vida, munición, etc. Se denomina HUD a todo lo que se “pinta” por encima del render y que le da información al jugador sobre lo que está pasando. [24]

Es uno de los aspectos más críticos en la creación de un videojuego. El diseño de un buen HUD reside en tener un buen equilibrio entre no molestar al jugador, y ofrecer toda la información necesaria para avanzar en el juego.

Aspectos a tener en cuenta:

- Información: que se quiere mostrar en pantalla.
- Localización de los elementos: como se sitúa esa información, hay que tener en cuenta que partes de la pantalla son más visibles para el ojo humano.
- Estética: aspecto de esa información en pantalla.

En el caso de los shooter, el que nos ocupa, es fundamental que la información no moleste, ya que son juegos muy rápidos.

Teniendo en cuenta todo lo anterior. La información que se va a mostrar es la vida y la munición, Ilustración 57, en el caso de la interacción con PC, y en el caso del uso de Leap Motion se mostrará la vida, la energía y la fuerza de disparo, Ilustración 58. Esta información se va a mostrar en la parte superior izquierda de la pantalla, debido a que el ojo humano suele mirar primero aquí, ya que las personas comienzan a leer por ese sitio, esto cambia en culturas donde se lee de derecha a izquierda.



Ilustración 57. Diseño HUD PC o mando



Ilustración 58. Diseño HUD Leap Motion

Implementación

Introducción

A continuación se va a proceder a describir las tecnologías, los formatos, y la gestión de datos que se ha realizado en la aplicación.

Formatos de importación utilizados

Se han utilizado los siguientes formatos de importación/exportación:

BLEND: formato que utiliza el software Blender para guardar proyectos realizados con la herramienta.

FBX: es un formato de archivo 3D que proporciona acceso al contenido creado en cualquier paquete de software. Las creaciones de Blender se exportan en este formato, para después ser importadas por Unity3D.

PNG: Portable Network Graphics, es un formato gráfico basado en un algoritmo de compresión sin pérdida para bitmaps. Formato para texturas reconocido por Unity3D.

JPG o JPEG: Joint Photographic Experts Group, es un formato gráfico que utiliza habitualmente un algoritmo de compresión con pérdida para reducir el tamaño de los archivos de imágenes. Formato para texturas reconocido por Unity3D.

JS: lenguaje de programación interpretado. Se define como orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico.

C#: lenguaje de programación orientado a objetos desarrollado y estandarizado por Microsoft como parte de su plataforma .net.

Gestión de datos

La gestión de los datos y recursos, se ha realizado organizándolos en carpetas por categorías, se ha detallado en el [AnexoVI. Gestión de datos](#).

Modelado

El modelado del proyecto se puede dividir en tres partes:

- Edificio del Museo de las ciencias príncipe Felipe.
- Decoración del interior del edificio.
- Visita virtual alternativa: Arma, vida y munición.

También hay que tener en cuenta que, tras el modelado de los objetos, hay que exportarlos de tal forma que puedan ser correctamente importados por Unity 3D, y también hay que crear los materiales y texturas correspondientes para cada parte de los objetos.

Museo de las ciencias Príncipe Felipe

El Museo de las ciencias Príncipe Felipe, es la pieza principal del proyecto, se ha intentado que se asemeje lo máximo posible al edificio real, por lo que se han ido reproduciendo cada una de las partes del edificio:

- Fachada suroeste
- Fachada noroeste
- Fachadas laterales
- Tejado

Fachada suroeste

La primera fachada que se ha creado es la fachada suroeste, debido a que es la más compleja de todas, ya que posee unas formas más complejas de diseñar.

Lo primero que se aprecia de este edificio es que posee formas que se repiten todo el tiempo, Ilustración 59, esto facilita mucho el trabajo de diseño, ya que sólo tendremos que crearlo una vez y luego replicarlo. En la Ilustración 60, está marcado claramente en azul una de las secciones del edificio, esta se corresponde a la sección creada.



*Ilustración 59. Museo de las ciencias Príncipe Felipe.
Detalle de la fachada suroeste*

El modelado de esta sección se ve perfectamente en la Ilustración 61 e Ilustración 62. A la vez que se iba creando el edificio se le iban añadiendo materiales a los diferentes elementos que lo componen, por la sencillez en este aspecto del edificio, sólo cuenta con dos el blanco que representa la estructura del edificio, y el azul que representa los cristales. Esta es la misma gama de colores que se va a mostrar a lo largo de toda la creación del edificio.



Ilustración 60. Museo de las ciencias Príncipe Felipe. Detalle de la fachada suroeste

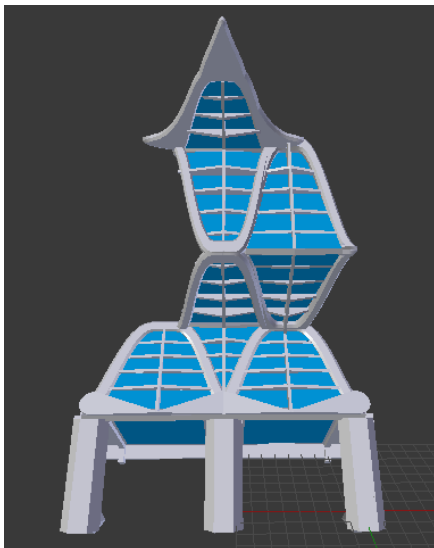


Ilustración 61. Modelado en Blender de la fachada suroeste, vista frontal



Ilustración 62. Modelado en Blender de la fachada suroeste, vista lateral

Fachada noreste

A continuación se ha realizado el otro lado de la fachada. Al igual que en el caso anterior esta fachada también posee formas que se repiten, Ilustración 63, por lo que sólo se debe modelar un fragmento, Ilustración 54. En este caso se ha reducido el número de ventanas que posee la fachada, ya que se consigue el mismo efecto visual de la fachada y se ve reducida la dificultad de su diseño. En la Ilustración 65 e Ilustración 66, se ve cómo ha quedado el modelado final.



Ilustración 63. Museo de las ciencias Príncipe Felipe.
Detalle de la fachada noreste



Ilustración 64. Museo de las ciencias Príncipe Felipe.
Detalle de la fachada noreste

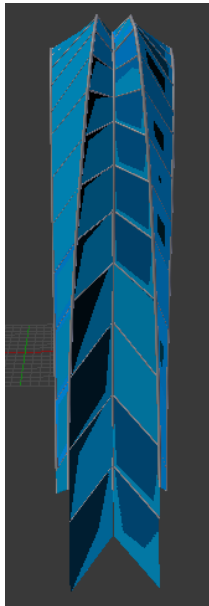


Ilustración 65. Modelado en Blender de la fachada
noreste, vista frontal

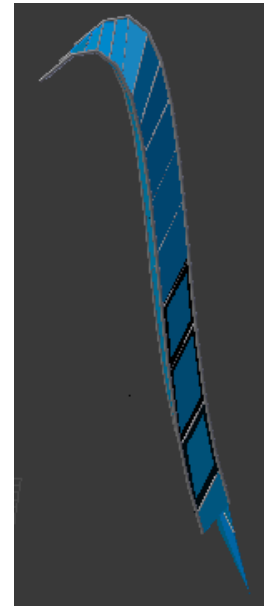


Ilustración 66. Modelado en Blender de la fachada
noreste, vista lateral

Fachadas laterales

A diferencia de las dos fachadas anteriores, la fachada lateral no tiene ningún patrón que se repite, Ilustración 67, pero sí que tiene otra particularidad, las dos fachadas laterales son simétricas, con lo que sólo hay que diseñar una de ellas y mediante una función espejo, se obtiene la otra fachada, Ilustración 68.



*Ilustración 67. Museo de las ciencias Príncipe Felipe.
Detalle de la fachada lateral*

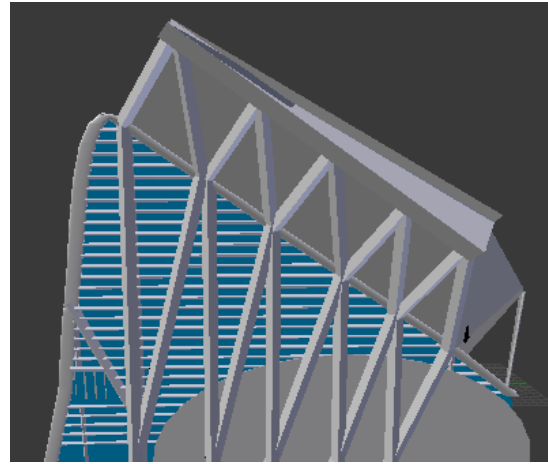


Ilustración 68. Modelado en Blender de la fachada lateral, vista frontal

Esta fachada supuso un problema cuando se añadió a Unity 3D, debido a que contiene varios vértices en un mismo punto, por este motivo se ha simplificado la fachada. El resultado final se muestra en la Ilustración 69.

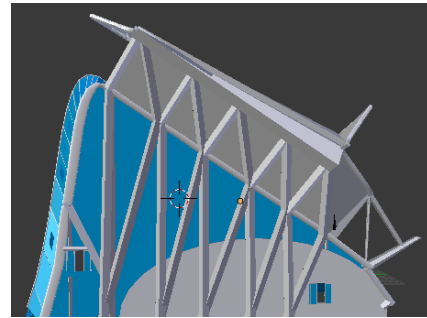


Ilustración 69. Modelado en Blender de la fachada lateral, vista frontal simplificada

Completado de fachadas

El siguiente paso consistió en la replicación de las dos fachadas principales y la unión de éstas con las fachadas laterales. En la Ilustración 70 e Ilustración 71 se puede ver cómo ha sido el resultado una vez realizados estos pasos.

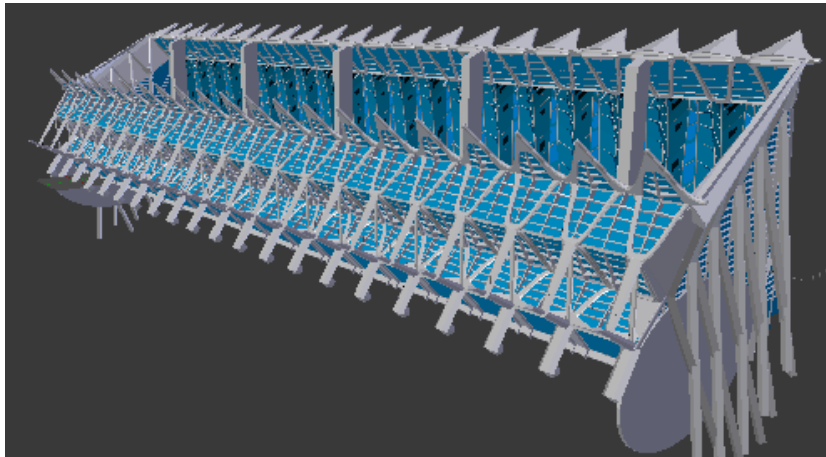


Ilustración 70. Modelado en Blender, fachada suroeste

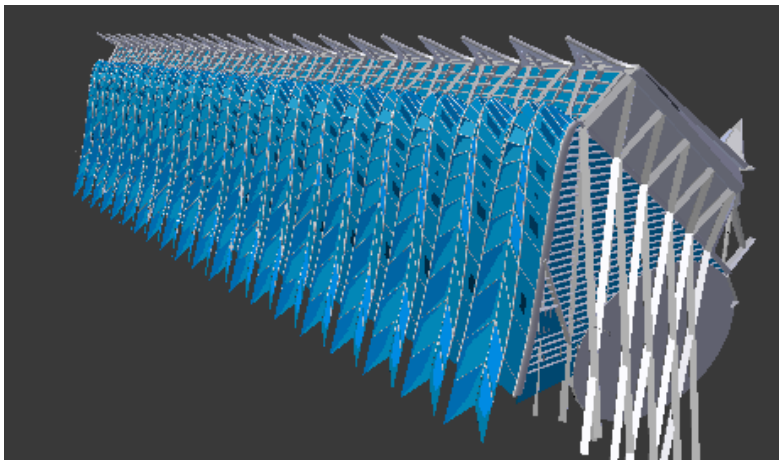


Ilustración 71. Modelado en Blender, fachada noreste

Tejado

El tejado es la parte más sencilla de todo el edificio, ya que son piezas rectangulares en forma de zig-zag, Ilustración 72.

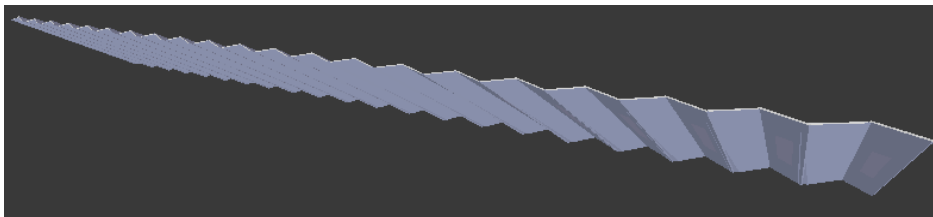


Ilustración 72. Modelado en Blender del tejado

Detalles finales

Por último se ha añadido los detalles que completan el edificio.

- Las cuatro plantas interiores que lo componen, Ilustración 73.
- Terraza de la fachada sureste, con escalares para acceder a la terraza, Ilustración 74.
- Terraza de la fachada noroeste, a esta terraza se accede desde dentro del edificio, Ilustración 75.

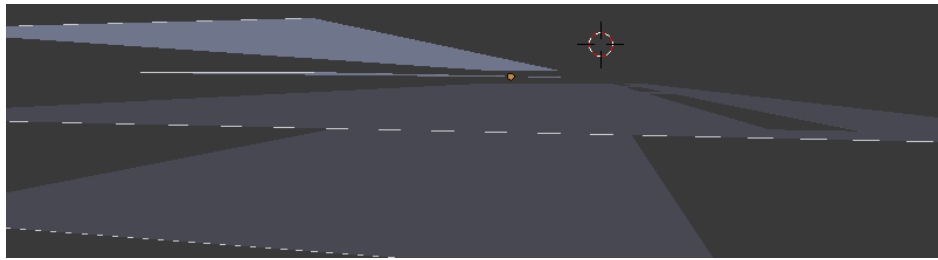


Ilustración 73. Modelado en Blender, cuatro plantas interiores del edificio

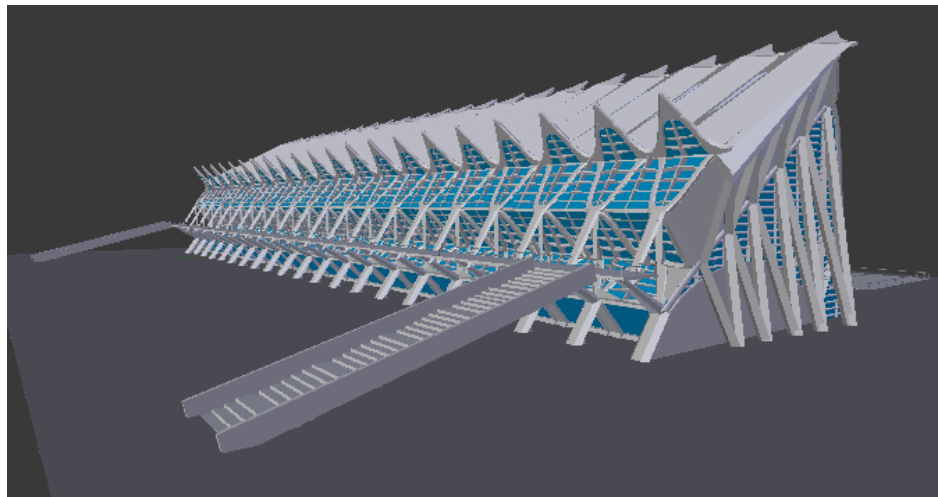


Ilustración 74. Modelado en Blender, fachada sureste finalizada

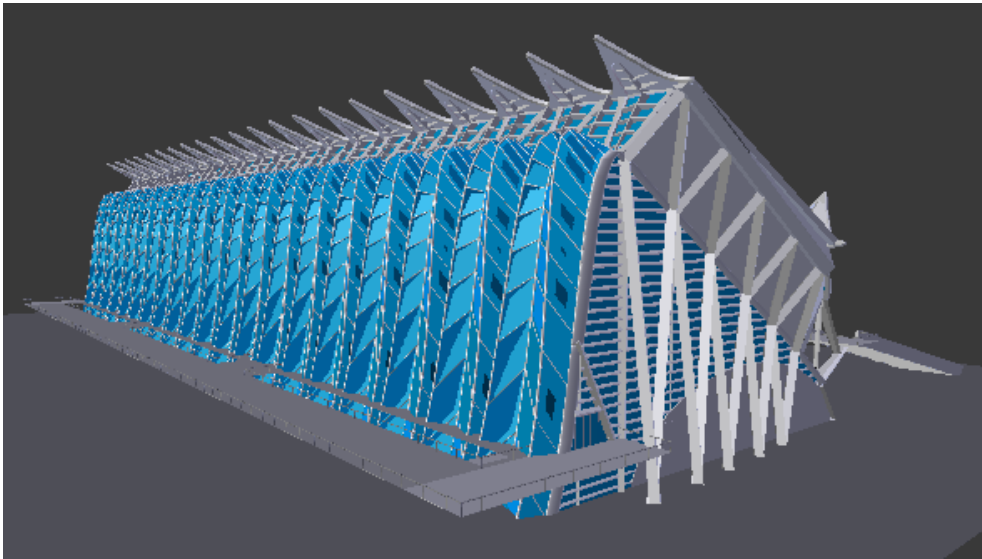


Ilustración 75. Modelado en Blender, fachada noreste finalizada

Decoración interior edificio

Para decorar el edificio por dentro se ha diseñado un interior sencillo, para que no suponga una gran carga a la hora de renderizar el juego, dando la sensación de que nos encontramos en una oficina, para ello se diseñaran mesas, sillas, lámparas, pantallas de proyección, proyectores, ordenadores, etc. Algunos de estos objetos serán los que se utilizaran para interactuar con ellos en la visita virtual.



Ilustración 76. Modelado en Blender, logo

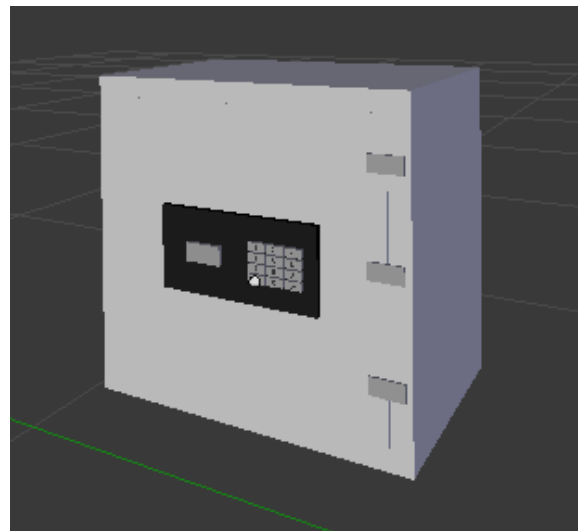


Ilustración 77. Modelado en Blender, caja fuerte

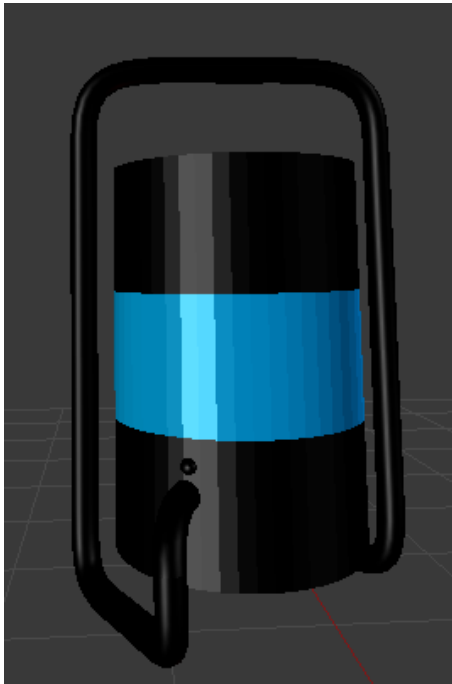


Ilustración 78. Modelado en Blender, lámpara mesa

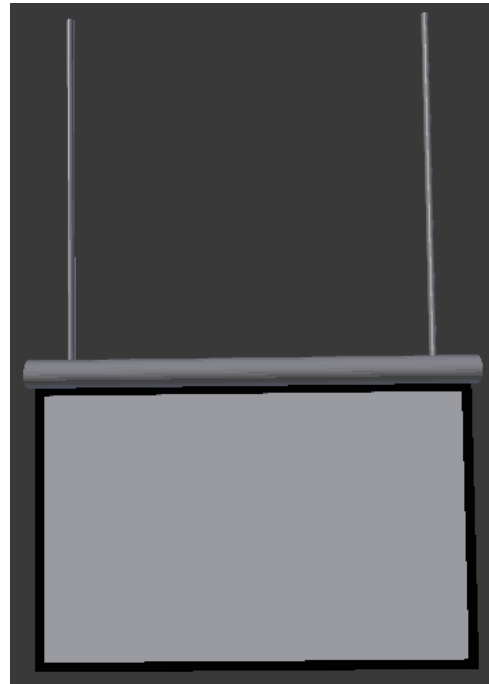


Ilustración 54. Modelado en Blender, pantalla de proyección

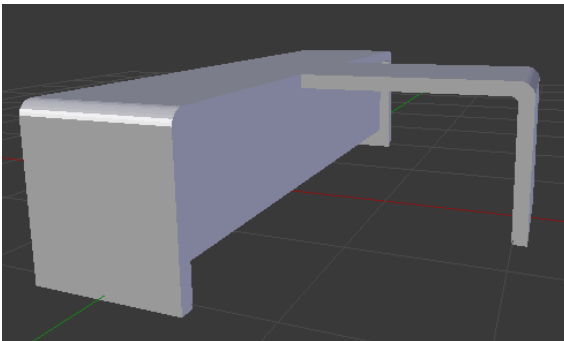


Ilustración 55. Modelado en Blender, mesa 1

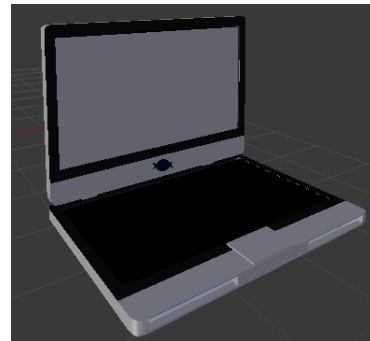


Ilustración 79. Modelado en Blender, pc

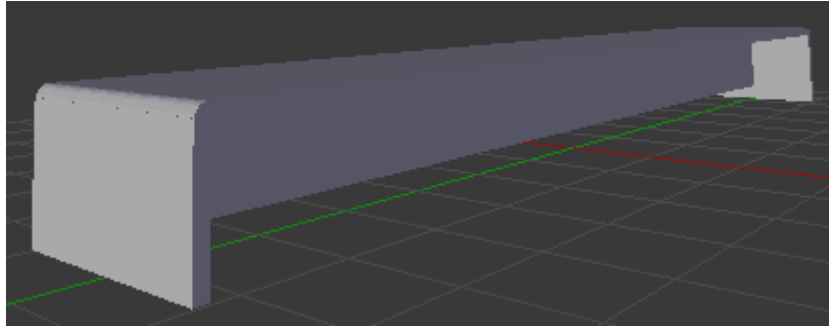


Ilustración 57. Modelado en Blender, mesa 2

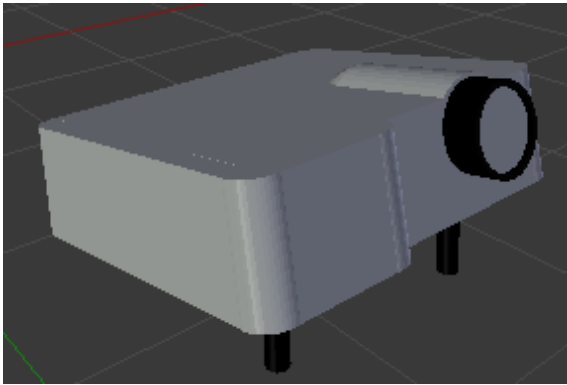


Ilustración 80. Modelado en Blender, proyector

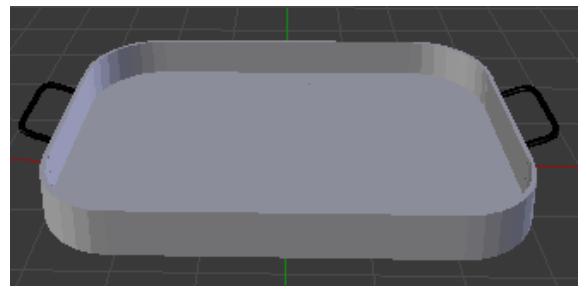


Ilustración 59. Modelado en Blender, bandeja

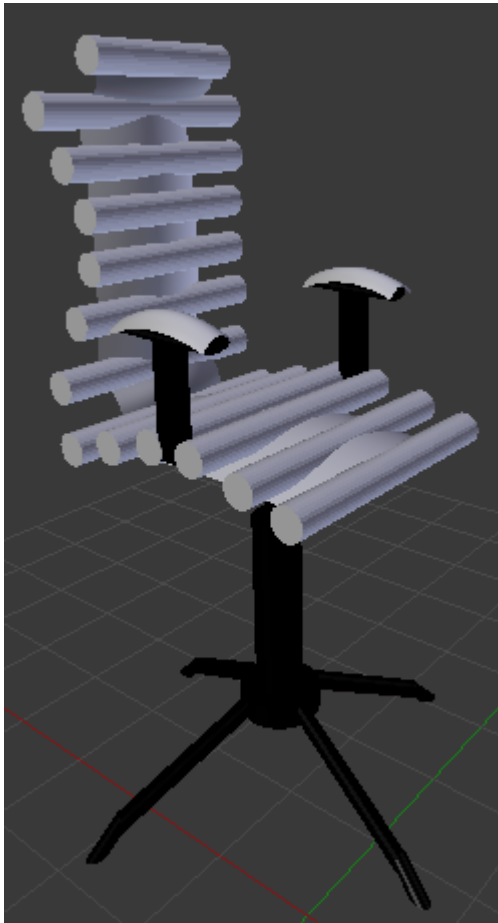


Ilustración 60. Modelado en Blender, silla

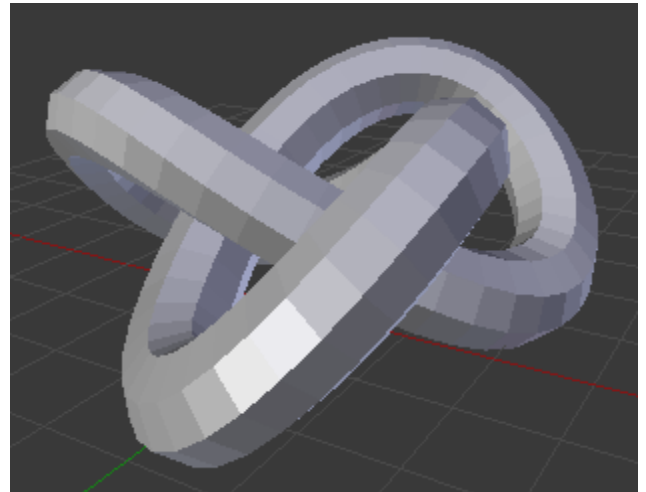


Ilustración 81. Modelado en Blender, escultura

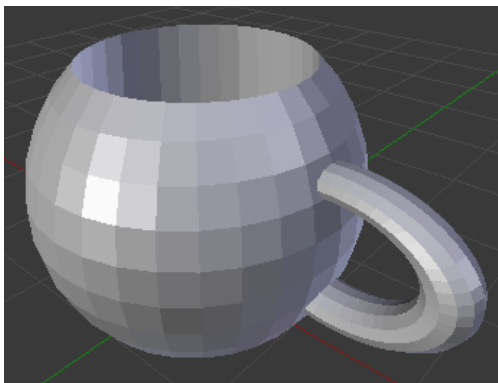


Ilustración 82. Modelado en Blender, taza



Ilustración 83. Modelado en Blender, tetera

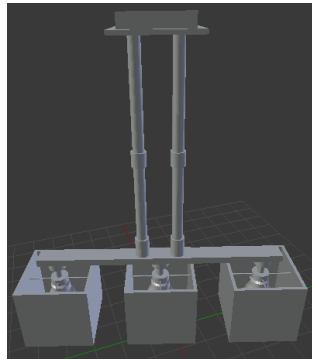


Ilustración 84. Modelado en Blender, lámpara techo

Por último se ha creado un mostrador, Ilustración 85, en el cual se han incluido algunos de los elementos anteriores, como el logo y las sillas.

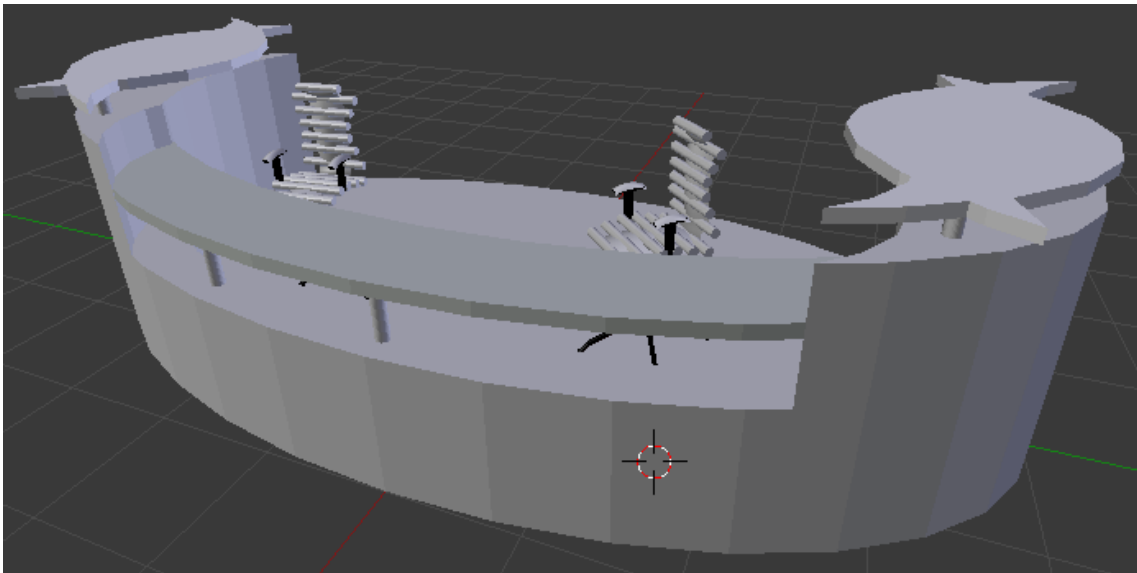


Ilustración 85. Modelado en Blender, mostrador

Visita virtual alternativa

En la visita virtual alternativa, se han necesitado diseñar algunos objetos más, se muestran a continuación.

Arma

En la opción que se juega con el PC o con el mando, se necesita de un objeto para poder disparar, por ello se ha creado un arma, Ilustración 86, para poder disparar a los enemigos que aparecen por el juego.



Ilustración 86. Modelado con Blender, pistola

Con las opciones de Leap Motion y Leap Motion con Oculus Rift, se utilizarán las manos para disparar.

Vida y munición

Cuando los enemigos son derrotados, en ocasiones dejan objetos, esos objetos son un botiquín de vida y una caja de munición, que se muestran en la Ilustración 87 y en la Ilustración 88.

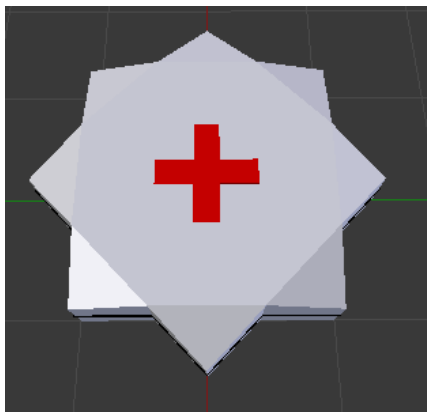


Ilustración 87. Modelado con Blender, vida

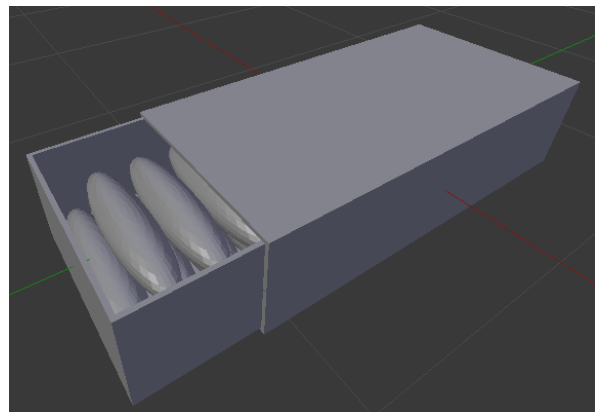


Ilustración 88. Modelado con Blender, munición

Exportación de modelos

Blender guarda sus proyectos en ficheros .blend, estos ficheros no son legibles por Unity, por lo que han de ser exportados a .fbx, formato que sí que admite Unity.

Materiales y texturas

Para que las texturas se vean correctamente en Unity se ha seguido el mismo proceso para todos los modelos de Blender. El proceso se detalla a continuación.

Materiales opacos

El material de la fachada, representado de color blanco, está pensado para después aplicar una imagen que le de textura, por lo que para su aplicación se utiliza el método UV unwrap el cual permite aplicar un imagen de dos dimensiones sobre una superficie 2D del modelo tridimensional. Las imágenes serán aplicadas después en Unity3D.

En la parte derecha de la Ilustración 89, se ve el proceso de mapping, ya que es la representación bidimensional del modelo tridimensional situado a la izquierda.

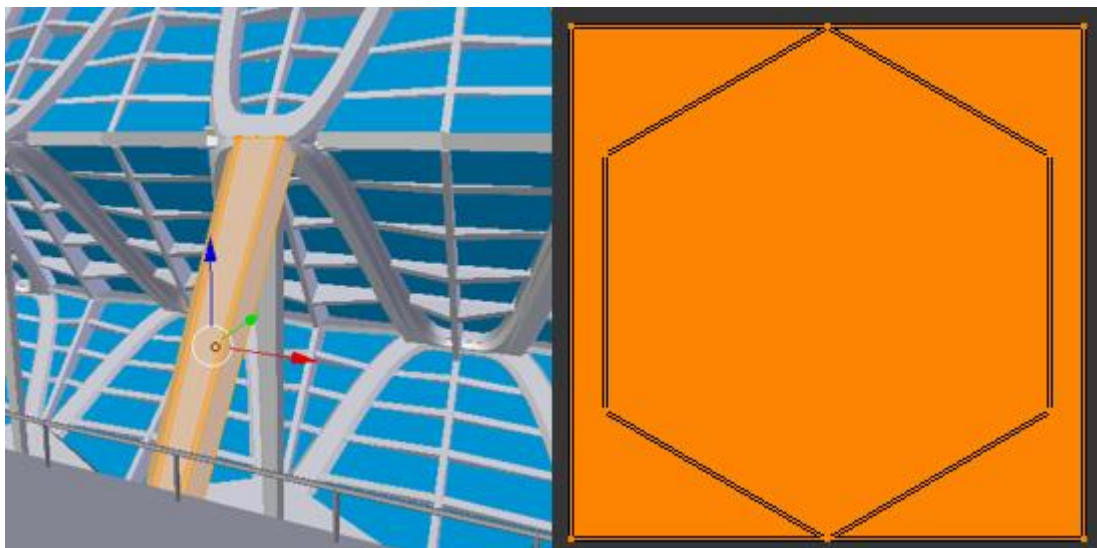


Ilustración 89. Modelado en Blender, proceso de mapping

La aplicación de los materiales al principio supuso un problema, ya que no se podía dejar el modelo perfectamente acabado con imágenes en Blender, debido a que al exportar en .fbx, ese formato no guardaba las texturas. Y tampoco se pudo mapear e insertar las imágenes con Unity3D, ya que el programa trataba cada cara de un objeto como caras separadas y mapeaba cada una de esas caras, repitiéndose la imagen n veces (nº de caras) en un mismo objeto, por lo que el resultado obtenido no era nada realista.

Llegar a la solución anteriormente mencionada no fue tarea fácil, pero el resultado obtenido es el que se buscaba desde un principio.

Cristales

El material azul de los cristales, también tiene sus particularidades, aunque no es tan complejo. En este caso no se aplica ningún tipo de imagen, sino que se ha creado un material transparente y que refleja la luz, de esta forma se ha conseguido un aspecto muy realista a los cristales. A la hora de exportarlo con .fbx y cargarlo en Unity3D no da ningún tipo de problema. En la Ilustración 90 se ve perfectamente la claridad del cristal, se ha colocado un rectángulo verde detrás de la fachada, para que se pueda apreciar claramente el efecto.

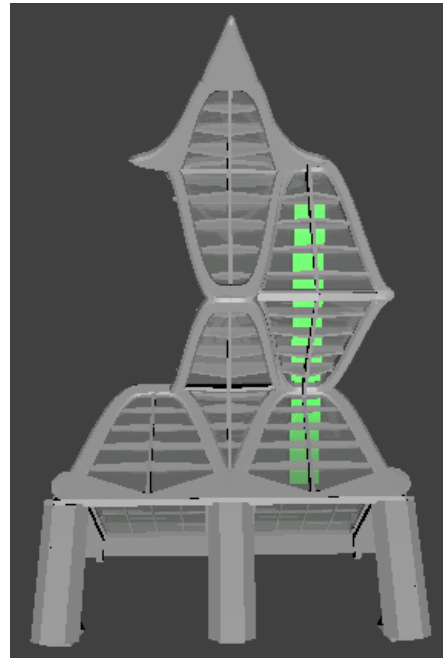


Ilustración 90. Modelado en Blender, cristales

Motor de juego

En este apartado se va a explicar todo el desarrollo que se ha llevado a cabo con la herramienta Unity3D.

Importación modelos 3D

Los modelos de Blender después de ser creados se exportaron a .fbx, los modelos en este formato serán los que se incluirán en el programa.

Modelos de Blender, aplicación de texturas

Tras el mapeado que se llevó a cabo en Blender, ahora solamente queda asociar una imagen a cada material, de esta forma cada material tendrá una textura diferente. Esto lo hemos hecho para todos los materiales opacos, la textura del cristal, su opacidad, reflejo y otras propiedades, que se adecuaron con Blender, por lo que no se han modificado. Para el resto de los materiales (opacos), se ha seleccionado una de estas dos opciones:

- Diffuse: Es la textura básica. La que le da el color a los objetos. No tiene ningún efecto especial sobre el modelo. Esta opción sólo admite una imagen, que será la del color que queramos para ese material.
- Bumped Diffuse: Además de tener una imagen que le da el color, tiene una segunda imagen correspondiente al mapa de normales. La imagen de normales ha sido creada a partir de la imagen del color, con el programa CrazyBump.

En la Ilustración 91, se ve donde se modifican los datos sobre el motor de juego.

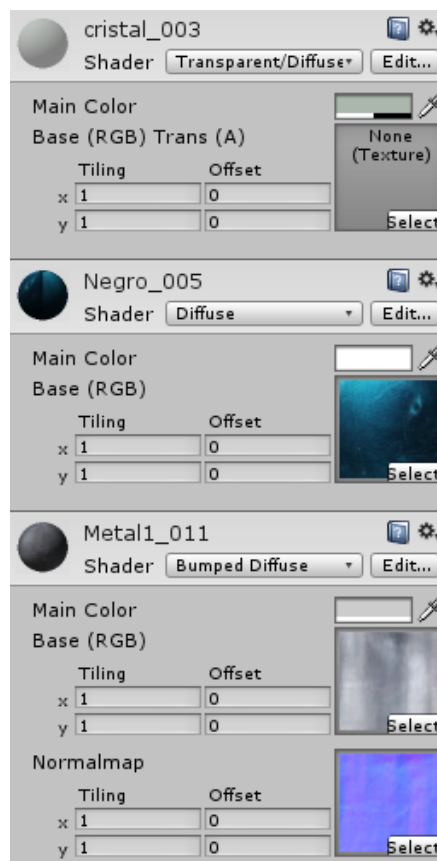


Ilustración 91. Importación de modelos en Unity, aplicación de texturas

En la Ilustración 92, se muestra en la imagen superior la imagen inicial, en la imagen del centro la imagen del proceso y en la imagen inferior la imagen de mapa de normales generada.

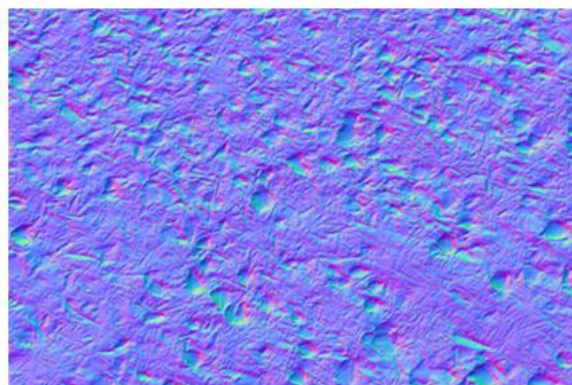
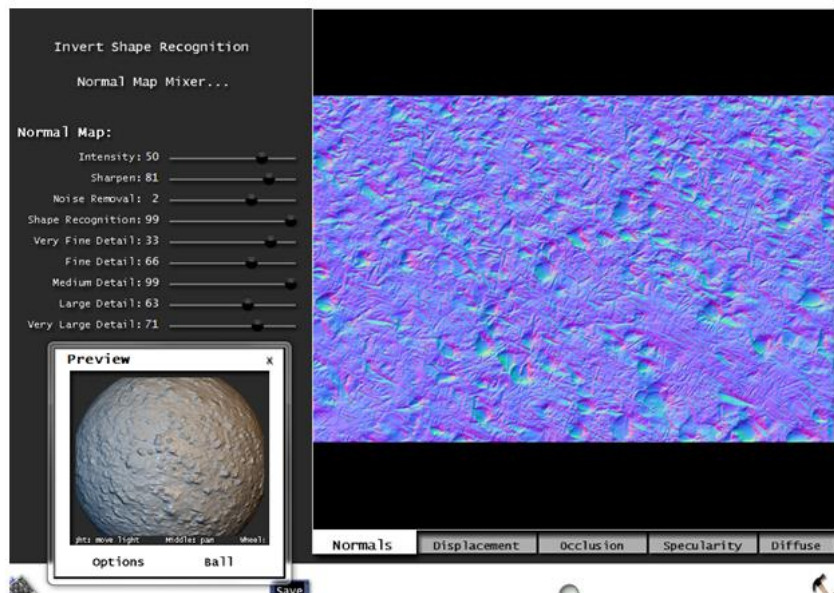


Ilustración 92. Proceso de generación de mapa de normales con CrazyBump

Decoración exterior

Para el exterior del edificio se ha creado una decoración sencilla, para que el peso de esta parte en el proyecto sea muy pequeño y permitir un renderizado rápido del proyecto.

Se ha creado un bosque en frente del edificio, para lo que se ha utilizado una herramienta de Unity que permite añadir muchos objetos de uno o más tipos rápidamente, Ilustración 93. En esta ocasión el bosque está compuesto por el mismo tipo de árbol, variando su color, tamaño..., el árbol utilizado pertenece a los objetos básicos que trae Unity por defecto, Ilustración 94. El resultado final se puede ver en la Ilustración 95.



Ilustración 93. Unity creación de bosques



Ilustración 94. Árbol básico de Unity

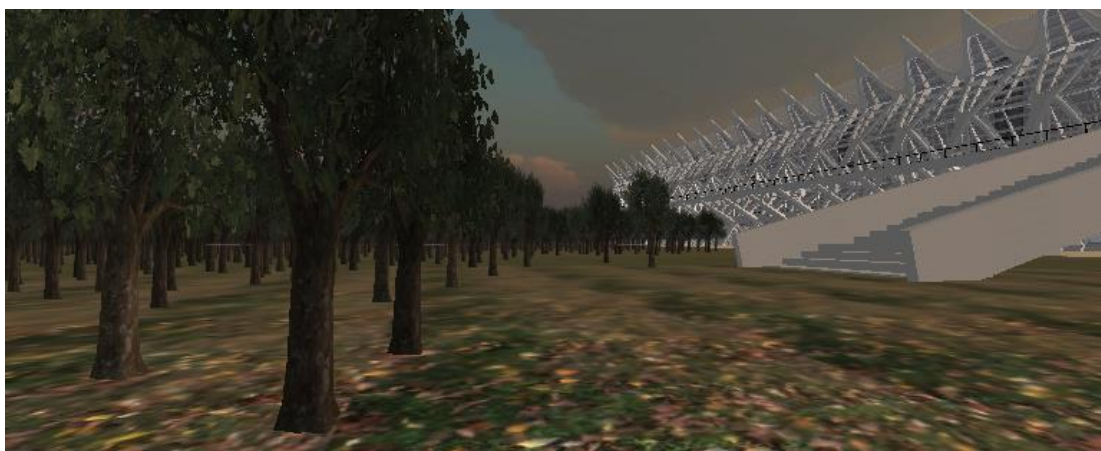


Ilustración 95. Bosque

Personajes

A continuación se va a explicar cómo se han integrado los enemigos en el proyecto, la integración de los personajes principales, los que son controlados por el usuario, se encuentran dentro de los apartados [Interacción](#) y [Realidad virtual](#), en función de la tecnología utilizada.

Enemigos

La creación de enemigos es un desarrollo que no se ha llevado a cabo para este proyecto. El enemigo que se ha utilizado pertenece al Asset Store de Unity3D, Ilustración 96.



Ilustración 96. Enemigo para la visita virtual alternativa

Dado que la finalidad del proyecto no era realizar una inteligencia avanzada para los enemigos, se les han asociado diferentes scripts, que realizan las siguientes acciones:

- Perseguir al jugador → los enemigos persiguen al jugador si este no se encuentra muy alejado, hasta acercarse a él.
- Atacar al jugador → cuando el enemigo se encuentra muy cerca del jugador, le ataca quitándole vida.
- Muerte → cuando el jugador dispara al enemigo y le alcanza, el enemigo muere, por lo que realiza una pequeña animación de que le han disparado y cae al suelo. En este momento el enemigo puede dejar un objeto al jugador:
 - Vida, 20% de probabilidades.
 - Munición, 20% de probabilidades.
 - Nada, 60% de probabilidades.

Existe otro script que se encarga de generar enemigos de forma aleatoria. Este script contiene diferentes puntos del juego en los cuales pueden salir enemigos, y cada cierto tiempo, genera un nuevo enemigo en uno de estos puntos al azar. De esta forma el jugador no se puede aprender el patrón en el que salen los enemigos, ya que va cambiando cada vez.

Menús

A continuación se muestra como ha quedado implementado el diseño de los menús. Los menús se han diseñado solo para la interacción sin Oculus Rift, ya que el escenario con ellas es otro ejecutable diferente. Cada menú tiene un título para que el usuario sepa donde se encuentra, y botones para seleccionar la opción que desee, como se indicó en el diseño.

Menú inicial

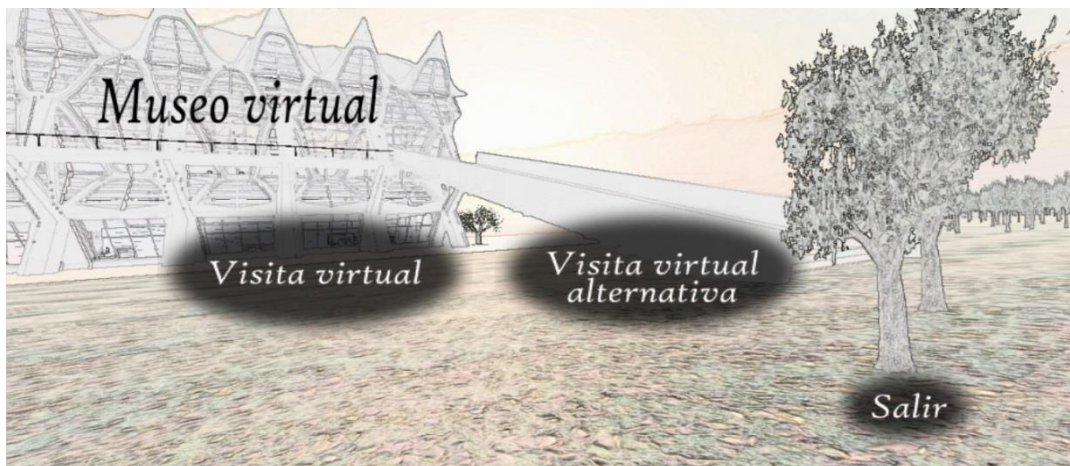


Ilustración 97. Menú inicial final

Menú visita virtual

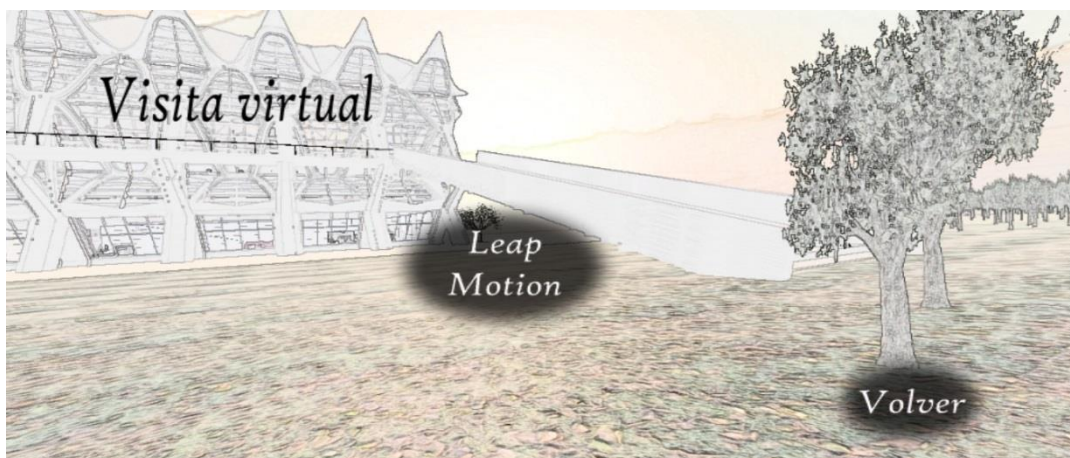


Ilustración 98. Menú visita virtual final

Menú visita virtual alternativa

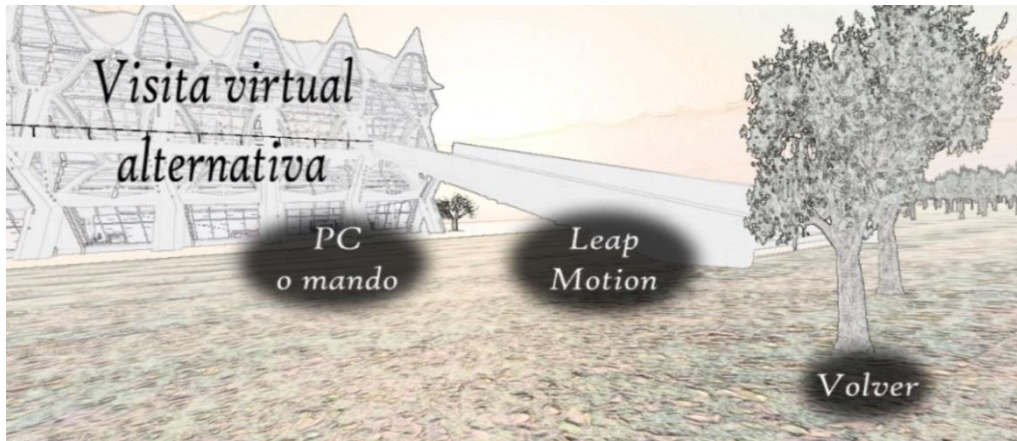


Ilustración 99. Menú visita virtual alternativa final

Menú derrota



Ilustración 100. Menú derrota final

Iluminación

Un aspecto también muy importante de la visita virtual es la iluminación, ya que en ella reside el conseguir el máximo realismo de las texturas.

Luz exterior

La luz de ambiente se ha disminuido a la mitad, de la máxima iluminación posible, para mantener que haya luz en toda la escena, pero sin tapar a la luz direccionada.

La luz direccionada es colocada infinitamente lejos y afecta todo de la escena, en este caso se ha colocado estratégicamente para que dé la sensación de que se trata del sol, Ilustración 102, de esta forma el escenario estará iluminado de forma diferente según hacia donde mire el personaje principal, además para que sea más realista el atardecer, se le ha dado un color anaranjado así como una baja intensidad, Ilustración 101 e Ilustración 103.



Ilustración 101. Iluminación de ambiente

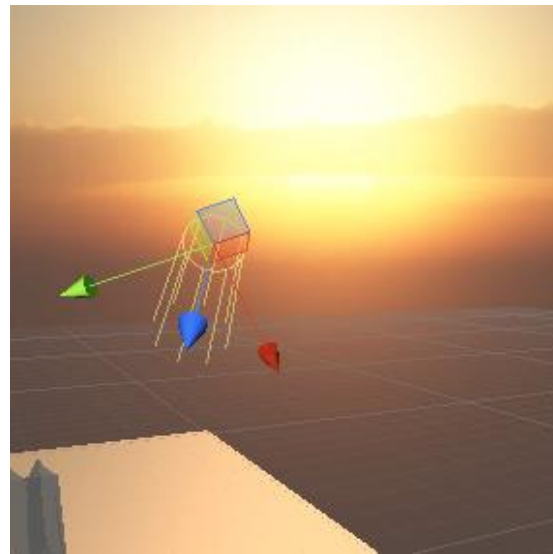


Ilustración 102. Iluminación exterior

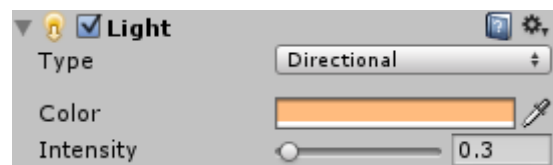


Ilustración 103. Iluminación exterior propiedades

Luz interior

En el interior del edificio se han añadido lámpara de techo y de mesa, las lámparas de techo se pueden ver en todas las plantas del edificio, Ilustración 104, cada una de ellas cuenta con tres puntos de iluminación, este tipo de iluminación alumbrada desde una ubicación igual en todas las direcciones, simulando perfectamente el resultado de una bombillo encendida. En este caso la luz emitida es de color blanco, y la iluminación también es baja, Ilustración 105. Las lámparas de mesa, cuentan con un único foco de luz y de tonalidad más amarilla.

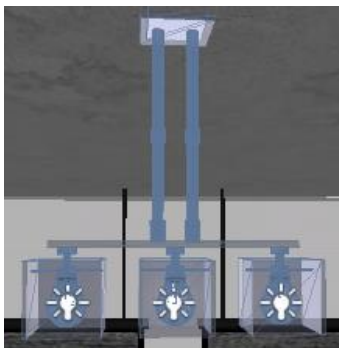


Ilustración 104. Iluminación interior, lámpara de techo

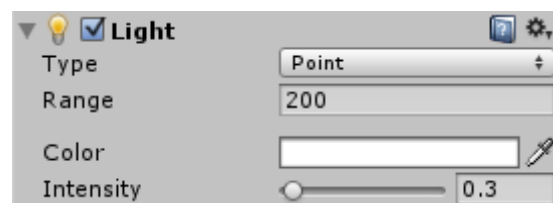


Ilustración 105. Iluminación interior, propiedades

Música

La música utilizada para acompañar al jugador durante la visita virtual, se ha obtenido de la página web [jamendo](http://www.jamendo.com). Y las canciones utilizadas son:

Para el menú principal y la visita virtual, autor de la música Chill Carrier:

www.jamendo.com/es/track/221334/beautiful-flow

Para el menú de derrota, autor de la música Chill Carrier:

www.jamendo.com/es/track/1067669/sparks-back-to-the-start-version

Interacción

En este apartado se va a explicar cómo está creada toda la lógica que existe detrás de los controladores y su interacción con los objetos.

Personajes principales

El personaje principal es un objeto específico creado con Unity, que contiene los objetos y scripts necesarios que cada personaje principal necesita. El personaje no tiene ningún tipo de diseño ya que al tratarse de un juego en primera no es necesario, ya que no se vería en ningún momento.

En el proyecto existen tres formas distintas de controlar el juego, creando tres personajes principales distintos, para cada una de ellas se asocian una tecla o movimiento a una acción determinada, se detallan a continuación.

PC o mando

En el motor de juego se comienza creando un First Person Controller, que contiene:

- Main Camera → cámara del juego, al tratarse de un juego en primera persona, el campo visual de esta cámara es lo que verá el jugador por pantalla.
 - Arma → se utiliza para disparar a los enemigos en la visita virtual alternativa.
 - Disparo → en la visita virtual alternativa, es el objeto que se dispara a los enemigos, se encuentra dentro del arma ya que al accionar el botón izquierdo del ratón el arma es el objeto encargado de disparar.

Además cuenta con dos altavoces:

- El que se encuentra en la zona delantera de la pistola, emitirá sonidos al disparar.
- El que se encuentra en el cuerpo del personaje, emitirá la música que acompañará al jugador durante la visita virtual.

En la Ilustración 106 se pueden ver algunas de esas partes.

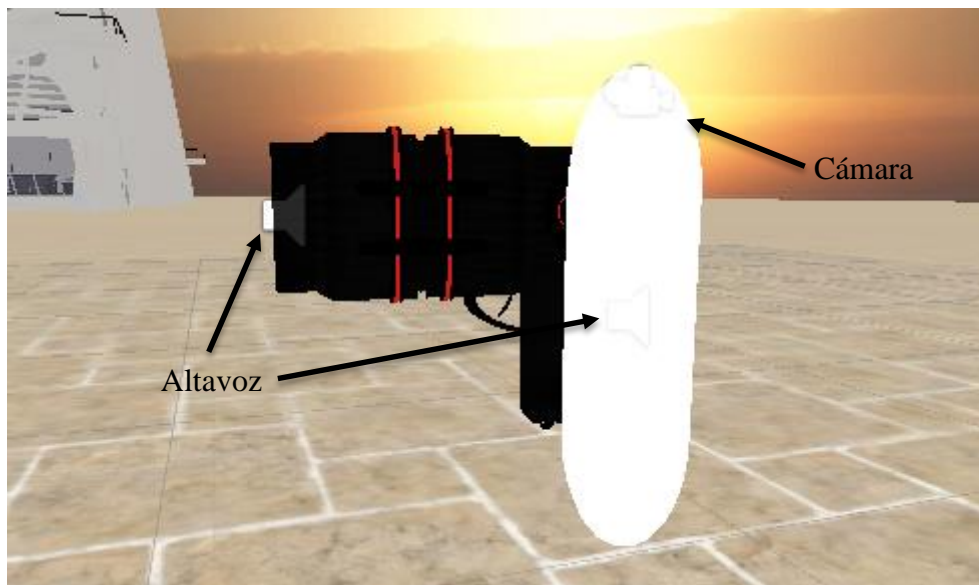


Ilustración 106. Personaje para la visita virtual con PC o mando

PC

Si se selecciona la opción de realizar la visita virtual con PC, se utilizará el teclado y el ratón, por lo que el personaje se controlaría mediante scripts de la siguiente forma:

- Mover cámara: movimientos de ratón.
- Mover Personaje:
 - Delante: W y flecha arriba.
 - Atrás: S y flecha abajo.
 - Izquierda: D y flecha derecha.
 - Derecha: A y flecha izquierda.
- Disparar: botón izquierdo ratón. Esta opción sólo está disponible para la visita virtual alternativa.

Mando

Desde el punto de vista del proyecto, el uso de un mando de consola no supone ninguna variación respecto a controlarlo con PC. Pero requiere un programa y nos pasos adicionales para asociar los botones del mando a los del PC, esto queda detallado en el apartado [Anexo VII. Manual de usuario](#).

Leap Motion

Para el uso de Leap Motion es necesario importar un plugin para su uso, lo encontramos en su página web y es específico para Unity. Gracias a este plugin tendremos un controlador en el proyecto que nos facilitará su uso.

Al igual que para el caso anterior se comienza creando un First Person Controller, que contiene:

- Main Camera → cámara del juego, al tratarse de un juego en primera persona, el campo visual de esta cámara es lo que verá el jugador por pantalla.
 - HandController → controlador que nos facilita el uso de Leap Motion en el juego, lo obtenemos al importar el plugin de Leap Motion.
 - Disparo → en la visita virtual alternativa, es el objeto que se dispara a los enemigos, se encuentra dentro del HandController ya que el disparo se lanza al abrir la mano derecha, y esto se controla con el HandController.

Además cuenta con dos altavoces:

- El que se encuentra más cerca del HandController, emitirá sonidos al disparar, este sonido sólo está habilitado en la visita virtual alternativa.
- El que se encuentra en el cuerpo del personaje, emitirá la música que acompañará al jugador durante la visita virtual

En la Ilustración 107 se pueden ver algunas de esas partes.

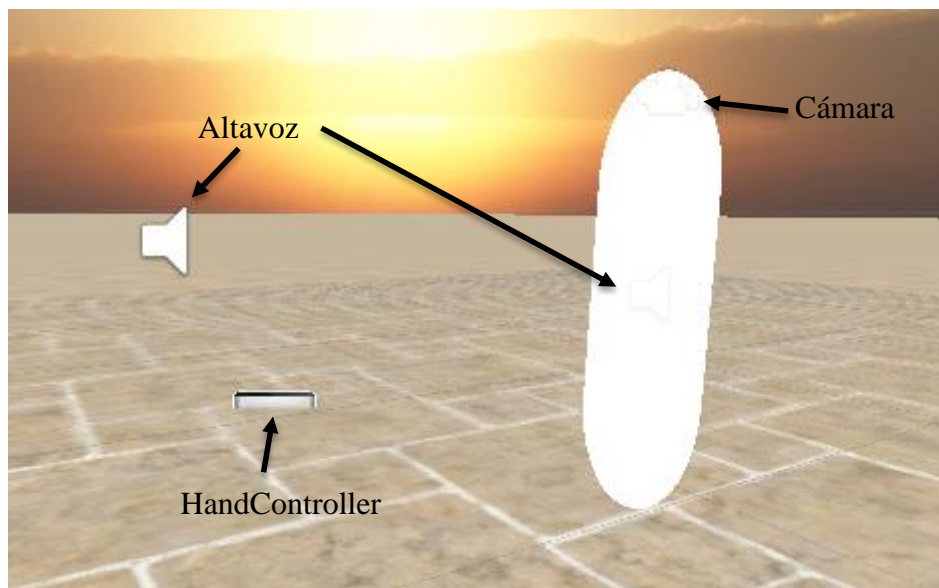


Ilustración 107. Personaje para la visita virtual con Leap Motion

En el motor de juego si se selecciona la opción de realizar la visita virtual con Leap Motion, el personaje se controlaría mediante scripts de la siguiente forma:

- Mover cámara: girar a la izquierda o derecha, la muñeca izquierda.
- Mover Personaje:
 - Delante: cerrar mano izquierda.
 - Atrás: no tiene esta opción, en su lugar deberá girar la cámara hacia uno de los lados y andar hacia delante.
 - Izquierda: no tiene esta opción, en su lugar deberá girar la cámara a la izquierda y andar hacia delante.
 - Derecha: no tiene opción, en su lugar deberá girar la cámara a la derecha y andar hacia delante.
- Disparar: abrir mano derecha. Esta opción sólo está disponible para la visita virtual alternativa.
- Recargar energía de disparo: cerrar mano derecha. Esta opción sólo está disponible para la visita virtual alternativa.

Interacción con objetos

Nuestra visita virtual se caracteriza porque además de que el jugador se encuentra en el edificio que desea visitar, puede interactuar con algunos objetos que se encuentran en él, ilustración 108. Para realizar esta interacción los objetos deben poseer un atributo llamado collider, este atributo nos permite saber cuándo otro objeto o personaje a colisionado con él, cuando esto sucede se muestra en pantalla una descripción de ese objeto en concreto, dependiendo del objeto con el que el jugador interactúe se le mostrará una información u otra.

Cuando se implementó esta opción para ser utilizada con Oculus Rift se encontraron algunos problemas, ya que el texto que se veía en pantalla cuando se utilizaban las Oculus Rift no era visible. La solución es asociar a la cámara de las gafas un script que mostrará la información del objeto con el que se interactúa.

La diferencia entre estas dos formas de mostrar la información es que en la primera la información la muestran directamente los scripts asociados a los objetos, mientras que en la segunda los scripts de los objetos mandan su información al script asociado a la cámara de las Oculus Rift, para que se vea por pantalla.

A continuación se muestran algunos ejemplos:

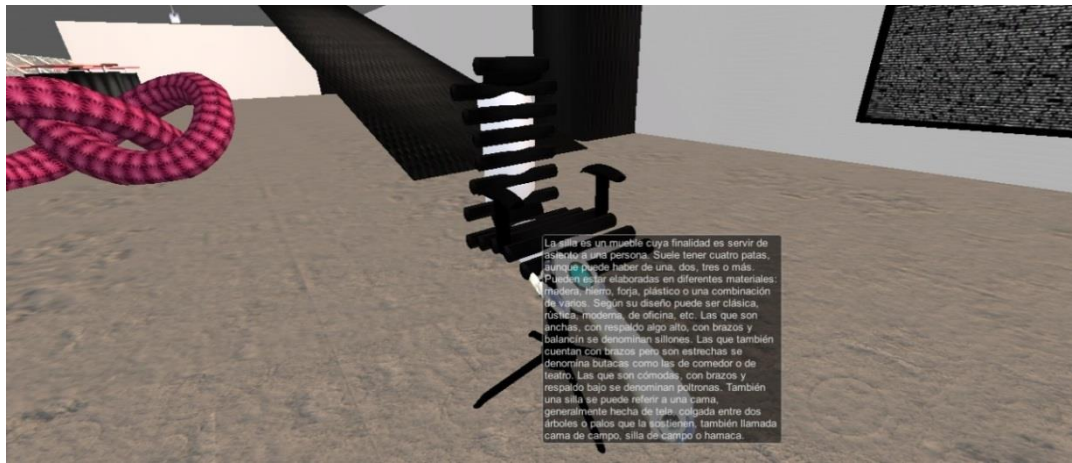


Ilustración 108. Interacción con objetos, silla

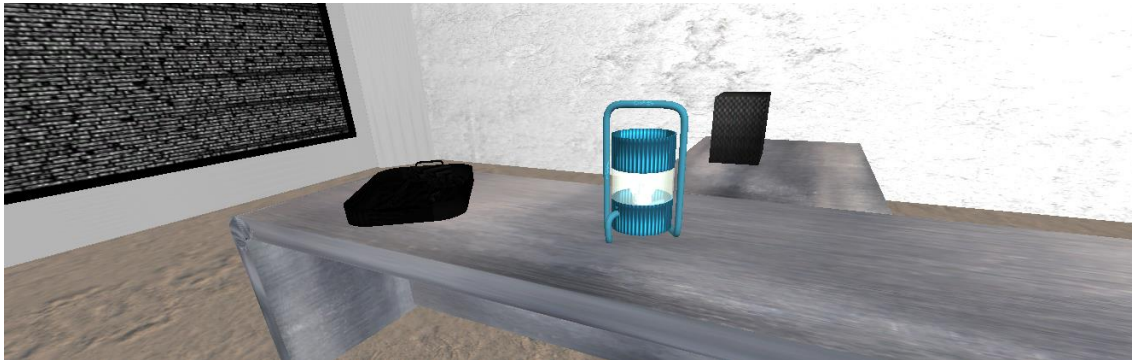


Ilustración 109. Interacción con objetos, lámpara antes de tocarla

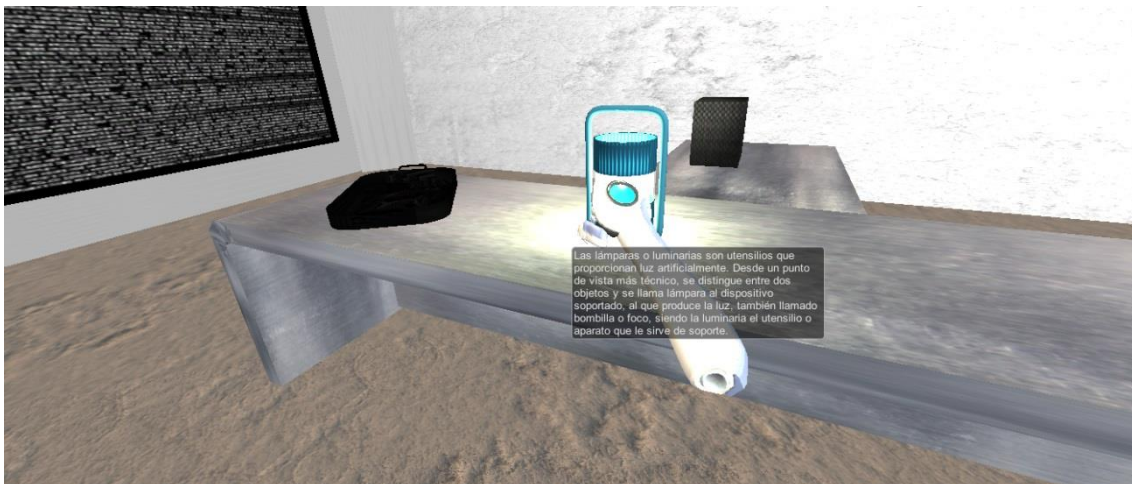


Ilustración 110. Interacción con objetos, lámpara después de tocarla

En el ejemplo de la lámpara, además de mostrar la información acerca del objeto, se ha encendido la luz.

En la Ilustración 109 se ve la lámpara antes de interactuar con ella, y en la Ilustración 110 después de interactuar con ella.

Botiquín y munición

El botiquín y la munición son dos objetos, que el jugador puede recoger al colisionar con ellos. El botiquín aumenta la vida del personaje, mientras que la munición restaura por completo la munición del personaje.

Realidad virtual

Para el uso de Oculus Rift es necesario importar un plugin para su uso, lo encontramos en su página web y es específico para Unity. Gracias a este plugin tendremos un controlador en el proyecto que nos facilitará su uso.

En este caso se comienza creando un OVR Player Controller, lo obtenemos al importar el paquete de Oculus Rift, que contiene:

- OVRCameraRig → esta cámara es especial y diferente a las otras dos, ya que emite dos imágenes, las contenidas en LeftEyeAnchor y RightEyeAnchor, para que con las Oculus Rift cada imagen sea visualizada por un ojo.
 - HandController → controlador que nos facilita el uso de Leap Motion en el juego, lo obtenemos al importar el plugin de Leap Motion.
 - Disparo → en la visita virtual alternativa, es el objeto que se dispara a los enemigos, se encuentra dentro del HandController ya que el disparo se lanza al abrir la mano derecha, y esto se controla con el HandController.
 - LeftEyeAnchor → nos permite ver la imagen que se vería por el ojo izquierdo.
 - CenterEyeAnchor → centro de ambas imágenes.
 - RightEyeAnchor → nos permite ver la imagen que se vería por el ojo derecho.

Además cuenta con dos altavoces:

- El que se encuentra más cerca del HandController, emitirá sonidos al disparar, este sonido sólo está habilitado en la visita virtual alternativa.
- El que se encuentra en el mismo punto que la cámara, emitirá la música que acompañará al jugador durante la visita virtual

En la Ilustración 111 se pueden ver algunas de esas partes.

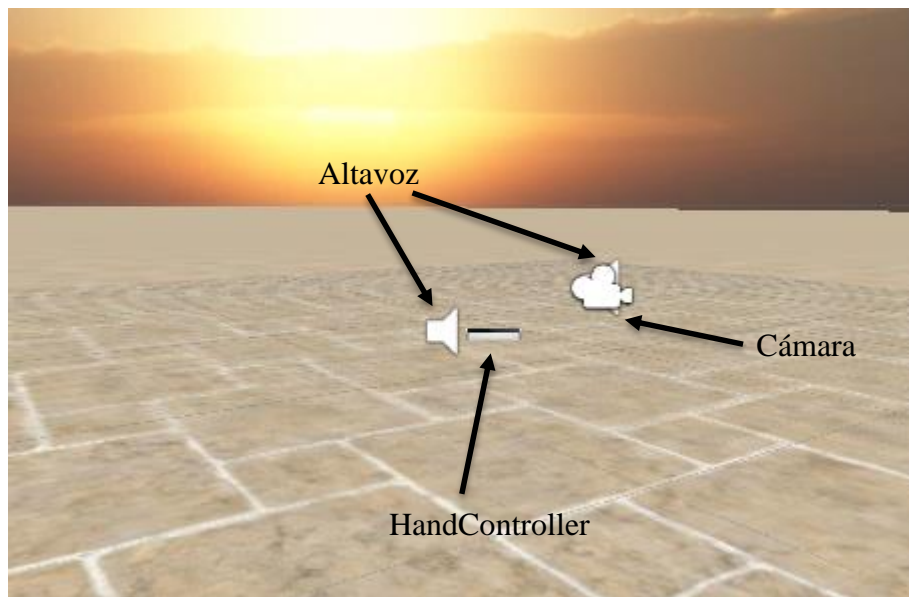


Ilustración 111. Personaje para la visita virtual con Leap Motion y Oculus Rift

En el motor de juego si se selecciona la opción de realizar la visita virtual con Leap Motion y Oculus Rift, el personaje se controlaría exactamente igual que en el caso en el que sólo se utiliza Leap Motion, ya que el controlador es el mismo. Cuando se implementó esta opción se encontraron algunos problemas, debido a que el Leap Motion ya no se encuentra paralelo al suelo, sino perpendicular. La solución fue girar la ubicación del HandController, girándolo 300° en el eje X y 180° en el eje Z. De esta forma el controlador del proyecto se encuentra en la misma posición que en la realidad.

6. Pruebas

Una vez finalizado el desarrollo, es necesario verificar si el resultado funciona correctamente y cumple con los requisitos establecidos.

Para ello, a continuación describiremos las pruebas necesarias para determinar que el sistema cumple las especificaciones requeridas.

La aplicación se va a probar con personas externas al proyecto, para su realización se les entregará el manual de usuario, así como las metas que tienen que realizar y un cuestionario para comprobar cómo les ha respondido la aplicación. Las pruebas tienen dos fases, la primera en la que tienen que realizar una serie de metas, y la segunda en la que deberán contestar unas preguntas sobre el funcionamiento de la aplicación.

Plantilla de pruebas

Primera fase, evaluación por metas. En ella se deberán realizar las metas marcadas en la Tabla 10, y se marcará si el usuario ha sido capaz de completarlas.

<i>Metas</i>	<i>Completado</i>	
	<i>SI</i>	<i>NO</i>
<i>Inicia la aplicación Museo</i>	<input type="checkbox"/>	<input type="checkbox"/>
<i>Selecciona la opción Visita virtual</i>	<input type="checkbox"/>	<input type="checkbox"/>
<i>Selecciona la opción Leap Motion</i>	<input type="checkbox"/>	<input type="checkbox"/>
<i>Interactúa con los objetos</i>	<input type="checkbox"/>	<input type="checkbox"/>
<i>Sal de la partida actual</i>	<input type="checkbox"/>	<input type="checkbox"/>
<i>Selecciona la opción visita virtual alternativa</i>	<input type="checkbox"/>	<input type="checkbox"/>
<i>Selecciona la opción PC o mando</i>	<input type="checkbox"/>	<input type="checkbox"/>
<i>Deja que los enemigos te derroten</i>	<input type="checkbox"/>	<input type="checkbox"/>
<i>Interactúa con el entorno</i>	<input type="checkbox"/>	<input type="checkbox"/>
<i>Vuelve al Menú principal</i>	<input type="checkbox"/>	<input type="checkbox"/>
<i>Selecciona la opción Visita virtual alternativa</i>	<input type="checkbox"/>	<input type="checkbox"/>
<i>Selecciona la opción Leap Motion</i>	<input type="checkbox"/>	<input type="checkbox"/>
<i>Interactúa con el entorno</i>	<input type="checkbox"/>	<input type="checkbox"/>
<i>Sal de la partida actual</i>	<input type="checkbox"/>	<input type="checkbox"/>
<i>Cierra la aplicación</i>	<input type="checkbox"/>	<input type="checkbox"/>
<i>Inicia la aplicación MuseoVisitaOculus</i>	<input type="checkbox"/>	<input type="checkbox"/>
<i>Interactúa con los objetos</i>	<input type="checkbox"/>	<input type="checkbox"/>
<i>Sal de la partida actual</i>	<input type="checkbox"/>	<input type="checkbox"/>

Tabla 10. Modelo de pruebas, metas

Segunda fase, evaluación con preguntas. Tras probar la aplicación con la fase uno, deberán contestar a las preguntas de la Tabla 11.

	<i>Preguntas</i>	<i>Respuesta</i>	
		<i>SI</i>	<i>NO</i>
	¿El ejecutable Museo funciona correctamente?	<input type="checkbox"/>	<input type="checkbox"/>
	¿Es necesaria alguna instalación adicional?	<input type="checkbox"/>	<input type="checkbox"/>
	¿El juego inicia correctamente?	<input type="checkbox"/>	<input type="checkbox"/>
	¿El menú inicial contiene las opciones visita virtual, visita virtual alternativa y salir?	<input type="checkbox"/>	<input type="checkbox"/>
	¿Si pulsas una de las opciones: visita virtual o visita virtual alternativa, se carga el menú correspondiente?	<input type="checkbox"/>	<input type="checkbox"/>
<i>Visita virtual Leap Motion</i>	¿El menú de visita virtual contiene la opción Leap Motion?	<input type="checkbox"/>	<input type="checkbox"/>
	¿Si pulsas en la opción: Leap Motion, se carga la partida correctamente?	<input type="checkbox"/>	<input type="checkbox"/>
	¿Has podido ver los mensajes que se muestran al interactuar con algunos de los objetos?	<input type="checkbox"/>	<input type="checkbox"/>
	¿Si pulsas ESC desde el escenario, sales al menú principal?	<input type="checkbox"/>	<input type="checkbox"/>
<i>Visita virtual alternativa PC o mando</i>	¿El menú de visita virtual alternativa contiene la opción PC o mando?	<input type="checkbox"/>	<input type="checkbox"/>
	¿Has visto a los enemigos?	<input type="checkbox"/>	<input type="checkbox"/>
	¿Has conseguido disparar alguno?	<input type="checkbox"/>	<input type="checkbox"/>
	¿Cuándo los enemigos te derrotan, se carga el menú correspondiente, con la opción de Menú inicial?	<input type="checkbox"/>	<input type="checkbox"/>
	¿Has sabido volver al menú principal?	<input type="checkbox"/>	<input type="checkbox"/>
<i>Visita virtual alternativa Leap Motion</i>	¿El menú de visita virtual alternativa contiene la opción Leap Motion?	<input type="checkbox"/>	<input type="checkbox"/>
	¿Has visto a los enemigos?	<input type="checkbox"/>	<input type="checkbox"/>
	¿Has conseguido disparar alguno?	<input type="checkbox"/>	<input type="checkbox"/>
	¿Si pulsas ESC desde uno de estos escenarios, sales al menú principal?	<input type="checkbox"/>	<input type="checkbox"/>
	¿El ejecutable MuseoVisitaOculus funciona correctamente?	<input type="checkbox"/>	<input type="checkbox"/>
	¿Es necesaria alguna instalación adicional?	<input type="checkbox"/>	<input type="checkbox"/>
	¿El juego inicia correctamente?	<input type="checkbox"/>	<input type="checkbox"/>
<i>Visita virtual Oculus Rift</i>	¿Has podido ver los mensajes que se muestran al interactuar con algunos de los objetos?	<input type="checkbox"/>	<input type="checkbox"/>
	¿Si pulsas ESC desde el escenario, sales al menú principal?	<input type="checkbox"/>	<input type="checkbox"/>
Observaciones:			

Tabla 11. Modelo de pruebas, preguntas

Resultados de las pruebas

Los resultados recogidos de las pruebas son los que aparecen en los catálogos Tabla 12 y Tabla 13.

<i>Metas</i>	<i>Completado</i>	
	<i>SI</i>	<i>NO</i>
<i>Inicia la aplicación Museo</i>	100 %	0 %
<i>Selecciona la opción Visita virtual</i>	100 %	0 %
<i>Selecciona la opción Leap Motion</i>	100 %	0 %
<i>Interactua con los objetos</i>	100 %	0 %
<i>Sal de la partida actual</i>	100 %	0 %
<i>Selecciona la opción visita virtual alternativa</i>	100 %	0 %
<i>Selecciona la opción PC o mando</i>	100 %	0 %
<i>Deja que los enemigos te derroten</i>	100 %	0 %
<i>Interactua con el entorno</i>	100 %	0 %
<i>Vuelve al Menú principal</i>	100 %	0 %
<i>Selecciona la opción Visita virtual alternativa</i>	100 %	0 %
<i>Selecciona la opción Leap Motion</i>	100 %	0 %
<i>Interactua con el entorno</i>	100 %	0 %
<i>Sal de la partida actual</i>	100 %	0 %
<i>Cierra la aplicación</i>	100 %	0 %
<i>Inicia la aplicación MuseoVisitaOculus</i>	100 %	0 %
<i>Interactua con los objetos</i>	100 %	0 %
<i>Sal de la partida actual</i>	100 %	0 %

Tabla 12. Catálogo de resultados de las pruebas, metas

	<i>Preguntas</i>	<i>Respuesta</i>	
		<i>SI</i>	<i>NO</i>
	¿El ejecutable Museo funciona correctamente?	100 %	0 %
	¿Es necesaria alguna instalación adicional?	0 %	100 %
	¿El juego inicia correctamente?	100 %	0 %
	¿El menú inicial contiene las opciones visita virtual, visita virtual alternativa y salir?	100 %	0 %
	¿Si pulsas una de las opciones: visita virtual o visita virtual alternativa, se carga el menú correspondiente?	100 %	0 %
<i>Visita virtual Leap Motion</i>	¿El menú de visita virtual contiene la opción Leap Motion?	100 %	0 %
	¿Si pulsas en la opción: Leap Motion, se carga la partida correctamente?	100 %	0 %
	¿Has podido ver los mensajes que se muestran al interactuar con algunos de los objetos?	83,34 %	16,66 %
	¿Si pulsas ESC desde el escenario, sales al menú principal?	100 %	0 %
<i>Visita virtual</i>	¿El menú de visita virtual alternativa contiene la opción PC o mando?	100 %	0 %

	<i>Preguntas</i>	<i>Respuesta</i>	
		<i>SI</i>	<i>NO</i>
<i>alternativa PC o mando</i>	¿Has visto a los enemigos?	100 %	0 %
	¿Has conseguido disparar alguno?	66,66 %	33,33 %
	¿Cuándo los enemigos te derrotan, se carga el menú correspondiente, con la opción de Menú inicial?	83,34 %	16,66 %
	¿Has sabido volver al menú principal?	100 %	0 %
<i>Visita virtual alternativa Leap Motion</i>	¿El menú de visita virtual alternativa contiene la opción Leap Motion?	100 %	0 %
	¿Has visto a los enemigos?	100 %	0 %
	¿Has conseguido disparar alguno?	66,66 %	33,33 %
	¿Si pulsas ESC desde uno de estos escenarios, sales al menú principal?	100 %	0 %
	¿El ejecutable MuseoVisitaOculus funciona correctamente?	100 %	0 %
	¿Es necesaria alguna instalación adicional?	0 %	100 %
	¿El juego inicia correctamente?	100 %	0 %
<i>Visita virtual Oculus Rift</i>	¿Has podido ver los mensajes que se muestran al interactuar con algunos de los objetos?	83,34 %	16,66 %
	¿Si pulsas ESC desde el escenario, sales al menú principal?	0 %	100 %
Observaciones: <ul style="list-style-type: none"> • Añadir giro analógico, de forma que gire más rápido cuanto más se dobla la muñeca izquierda. • En algunas zonas donde existen un número elevado de objetos el proceso de renderizado es un poquitín más lento. • Sensación de inmersión con las Oculus Rift está muy conseguida. • Dificultad inicial para adaptarse a las nuevas formas de interacción, aunque de fácil adaptación en poco tiempo. • Los mensajes no se ven del todo nitidos con las Oculus Rift. 			

Tabla 13. Catálogo de resultados de las pruebas, preguntas

Conclusiones de las pruebas

Las pruebas han dado un buen resultado, tal y como se esperaba. De las metas propuestas, todos los usuarios han sido capaces de realizarlas sin ningún tipo de problema. En cuanto a la evaluación del programa, en general ha respondido bastante bien, aunque cabe destacar que donde en el renderizado a veces se ralentizaba un poquitín, como han indicado los usuarios. Esto es debido a la gran carga de objetos y materiales que posee el edificio, se podría solucionar, como indica uno de los usuarios que la ha probado, mediante la carga parcial del escenario, de tal forma que la cantidad de información que debe renderizar el programa en cada instante se reduciría, y así se evitaría esa ralentización.

Otra sugerencia realizada consiste en realizar giros de forma analógica, esta forma consistiría en que a mayor giro de la muñeca más rápido sería el giro del personaje, a diferencia de la forma actual que independientemente del ángulo de giro de la muñeca,

el personaje realiza el giro a la misma velocidad. Esto se podría implementar adicionalmente al anterior, dado que no a todo el mundo le ha costado realizar los giros con el sistema actual, por lo que se podría crear un menú de opciones donde el usuario pueda seleccionar que tipo de opción de giro prefiere.

En cuanto a los mensajes que no se ven del todo nítidos cuando se utilizan las Oculus Rift, esto es algo que desde el presente proyecto no tiene solución, ya que es debido a una limitación de la tecnología utilizada, las gafas se encuentran actualmente en desarrollo y aún les quedan cosas que perfeccionar, como aumentar la calidad visual que ofrecen.

Estos son datos importantes a tener en cuenta para futuras mejoras.

7. Marco regulador

Como toda industria en alza que mueve grandes cantidades de dinero, la industria del videojuego y la administración central han establecido un sistema de criterios y normas que permiten regular el mercado de los videojuegos en España. [25] [26]

- Código nacional de autorregulación: Elaborado por la aDeSe en el año 2001, establece las pautas de actuación en el desarrollo de software interactivo en aspectos como la clasificación, etiquetado, promoción y publicidad
- Código europeo de autorregulación: ISFE (Interactive Software Federation of Europe) representa los intereses del sector del software interactivo ante la Unión Europea y otras instituciones internacionales. Entre sus objetivos destacan la promoción y protección de la industria del software a nivel europeo y mundial, la participación de sus miembros en áreas como la lucha contra la piratería, la protección de los menores y el medioambiente, la obtención de datos sobre mercados como fuente de información o la defensa de la industria europea del videojuego como elemento clave de la economía y nuevo vector de la cultura.

El Sistema PEGI (Pan European Game Information), Ilustración 112, es el mecanismo de autorregulación diseñado por la industria para dotar a sus productos de información orientativa sobre la edad adecuada para su consumo. Fue desarrollado por la ISFE y entró en práctica en abril del 2003. El sistema PEGI se aplica en 25 países sin tener relación alguna con la Unión Europea, entre ellos España.

Existen dos formas de clasificación para cualquier software; una de edad sugerida y otra sobre seis descripciones de contenido, tales como el uso de lenguaje indecente, violencia, etc.



Ilustración 112. Código sistema PEGI

En cuanto a los elementos que se utilizan en el proyecto, todos han sido diseñados y creados desde cero, exceptuando los árboles utilizados para la decoración exterior que pertenece a los objetos básicos que trae Unity por defecto, y el enemigo perteneciente al

Asset Store de Unity3D. Ninguno de los elementos posee ningún tipo de restricción de uso.

También se ha añadido al videojuego música, que se ha obtenido de la página web [jamendo](#). En concreto se han utilizado dos canciones pertenecientes al mismo artista llamado Chill Carrier, que son de libre distribución siempre que se mencione el nombre del artista. En la Ilustración 113 se muestran los requisitos que se indican en la página citada para el uso de las canciones seleccionadas.



Puedes copiar, distribuir, promocionar y poner esta pista siempre que tú:



Menciones el nombre del artista

Ilustración 113. Regulación de la música utilizada

Otra regulación a tener en cuenta es la LOPD (Ley Organiza de Protección de Datos), es una Ley Orgánica española que tiene por objeto garantizar y proteger, en lo que concierne al tratamiento de los datos personales. En nuestro caso no guardamos ningún tipo de información del usuario. [27]

8. Entorno socio-económico

Sin duda, la industria de los videojuegos ha tenido gran impacto a nivel mundial, tanto en lo económico, como en su aspecto tecnológico ya que con la aparición de los nuevos juegos con altas resoluciones y complejidad y sus respectivas consolas, se ha disparado el nivel de tecnología a puntos nunca antes vistos. Por otra parte también es importante lo social, debido a la gran importancia que tienen los mismos en los humanos y la constante lucha entre sus “pros y contras”. [28] [29]

El impacto que han tenido los videojuegos en nuestro país, especialmente en los últimos años, es notorio. Nuestra sociedad se encuentra fuertemente afectada por la industria de los videojuegos. De las 330 empresas desarrolladoras de videojuegos censadas en España facturaron 313,7 millones de euros (mayo de 2014), Ilustración 114, cifra que sitúa al sector entre las principales industrias nacionales de contenidos digitales. Además se prevé que aumentará su facturación un 130% en los próximos tres años.

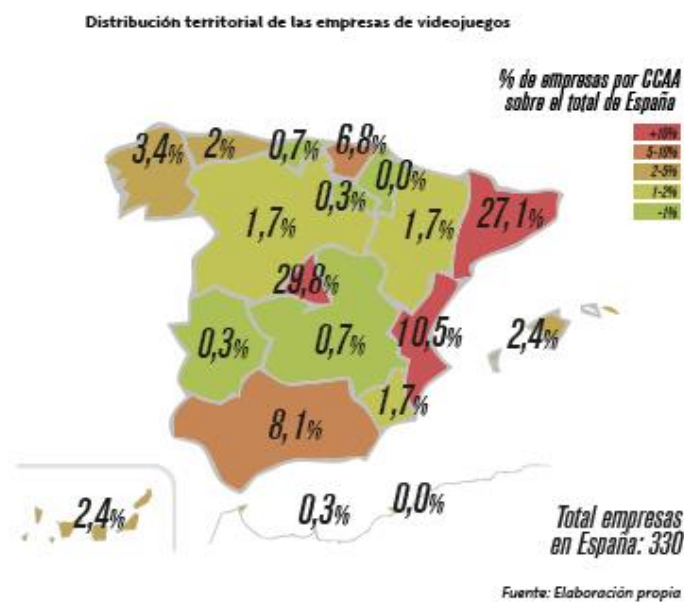


Ilustración 114. Distribución de las empresas de videojuegos en la geografía española

Esto nos indica que desde el punto de vista de social la creación de visitas virtuales inmersivas sería muy positiva, ya que nuestro país es puntero en el uso de estas tecnologías, fomentando la visita de gente a los museos, pudiéndose realizar desde cualquier lugar, como escuelas o desde los propios hogares, permitiendo que los usuarios puedan interactuar completamente con el ambiente artificial utilizando los sentidos del tacto, el oído, y la vista mediante dispositivos especiales que están conectados al ordenador.

A pesar de todo ello, este proyecto no se podría poner en funcionamiento en estos momentos, debido a que las formas de interacción utilizadas: Leap Motion y Oculus Rift, están aún en desarrollo. Una vez que se finalice el desarrollo de ambos dispositivos, el proyecto se podría implantar en cualquier ordenador.

9. Conclusiones

En esta sección, y para terminar el documento, se presenta una conclusión sobre el trabajo realizado.

Aportaciones realizadas

Los objetivos iniciales que se perseguían con la elaboración de este proyecto eran:

- Estudiar las etapas de desarrollo de una visita virtual, partiendo de las posibilidades existentes en la actualidad para la creación de un videojuego.
- Desarrollar una visita virtual de un museo desde cero.
- Desarrollo de una visita virtual desde cero, para demostrar que como se ha indicado ambos parten de la misma base.
- Incorporación de nuevas formas de interacción entre el usuario y el sistema, para ello se incorporará la interacción mediante:
 - Leap Motion → dispositivo de entrada para que el usuario se mueva por el escenario e interactúe con los objetos de la visita, para ello se asociarán acciones realizadas por el usuario con acciones en el escenario.
- Incorporación de nuevas formas de realidad virtual, que den al usuario la sensación de estar inmerso en el museo recreado.
 - Oculus Rift → dispositivo de salida para que el usuario vea el escenario como si estuviera allí mismo.

Este proyecto ha cumplido con todos los objetivos que inicialmente se habían marcado, y se han realizado algunos más:

- Se ha realizado un estudio sobre las etapas de desarrollo de una visita virtual, partiendo de las posibilidades existentes en la actualidad, para la creación de un videojuego.
- Se ha realizado un estudio sobre las formas de interacción actualmente utilizadas por los usuarios, para los videojuegos, así como de las formas futuras que se están desarrollando ahora mismo.
- El desarrollo de una visita virtual, se ha creado desde cero:
 - Modelado del Museo de las Artes y las Ciencias de Valencia, así como sus texturas correspondientes.
 - Modelado de toda la decoración interior del edificio, sillas, mesas, lámparas, etc., así como sus texturas correspondientes.
 - Diseño del exterior del edificio, mediante la creación de un bosque.
- Se han creado dos formas de juego diferentes:
 - Visita virtual: en ella se puede interactuar con los objetos del escenario, cuando se toca un objeto se muestra en pantalla información sobre él.

- Visita virtual alternativa: además de mostrarse la información anterior, se han añadido enemigos, creando un juego en primera persona en el que hay que intentar sobrevivir.
- Se han incorporado formas de interacción actualmente utilizadas, asociando teclas a acciones en el juego:
 - PC o mando, se utilizan para realizar una visita virtual alternativa.
- Se han incorporado nuevas formas de interacción entre el usuario y el sistema, asociando acciones realizadas con las manos a acciones en el juego:
 - Leap Motion, se utiliza para realizar una visita virtual y una visita virtual alternativa.
 - Leap Motion y Oculus Rift, se utiliza para realizar una visita virtual.
- Se ha incorporado un menú que guía al usuario por la visita virtual, dependiendo de la forma de interacción que desee utilizar.
- Se ha añadido música a todo el juego, que ayuda a que el usuario se integre mejor en él.

Desde el punto de vista económico, tal y como se ve en el [Anexo VIII. Análisis de la planificación y coste del proyecto tras su finalización](#), el proyecto se ha resuelto satisfactoriamente.

Problemas encontrados y trabajos futuros

A continuación se presentan posibles mejoras que añadir al sistema desarrollado. Todas estas mejoras podrían afectar positivamente a su funcionamiento.

Solucionar los problemas encontrados:

- Renderizado por partes. Tras la realización de las pruebas, se ha visto que el programa donde más ha fallado es en el renderizado, como han indicado los usuarios que lo han probado, ya que al proyecto le costaba funcionar en algunos momentos y la imagen se ralentizaba demasiado. Esto es debido a la gran carga de objetos y materiales que posee el edificio, se podría solucionar, como indica uno de los usuarios que la ha probado, mediante la carga parcial del escenario, de tal forma que la cantidad de información que debe renderizar el programa en cada instante se reduciría, y así se evitaría esa ralentización, esto es un dato importante a tener en cuenta para futuras mejoras.
- Giro analógico. Otra sugerencia realizada en las pruebas, es que los giros se realicen de forma analógica, como se han explicado anteriormente. Esto se podría implementar adicionalmente al anterior, dado que no a todo el mundo le ha costado realizar los giros con el sistema actual, por lo que se podría crear un menú de opciones donde el usuario pueda seleccionar que tipo de opción de giro prefiere.

Sugerencias de nuevas implementaciones, que no se han podido realizar debido al tiempo asignado al proyecto actual:

- Aumentar el abanico de formas de interacción, según vayan apareciendo nuevas formas en el mercado, consiguiendo que la experiencia de usuario e inmersión cada vez sea mayor, debido a la mejora de las tecnologías actuales.
- Añadir audio descripciones, para que puedan utilizarlo personas ciegas. La aplicación iría indicando al usuario que debe hacer en cada momento, para que pueda alcanzar los objetos, una vez alcanzado el objeto se le describiría como es ese objeto y la descripción que tiene asociada.
- Añadir cinemáticas al juego, por ejemplo que las puertas se abran cuando vas a pasar.
- Implementación de un sistema multijugador. La visita virtual sería más interesante, si se puede visitar junto con amigos o familiares, ya que la mayoría de los visitantes de los museos lo hacen acompañados. De esta forma se podría comentar con otras personas las opiniones e impresiones sobre la visita que se está realizando.
- Creación de una IA más realista para la visita virtual alternativa. Para ello se podría crear un script desde cero o modificar el script ya existente, para que los enemigos por ejemplo tengan una estrategia de ataque cooperativa.
- La integración en un solo ejecutable. La parte del juego con Oculus Rift requiere ser compilados de forma independiente para que funcione. Integrándola en otra compilación enriquecería el proyecto completo.

10. Bibliografía

- [1] S. O. J. M. José R. Hilera. [En línea]. Disponible:
http://www.cursosdred.es/php/cursos/sc_formador_formadores/modulo2/unidad2/ampliar/Aplicacion_Realidad%20virtual_ensenanza_internet.pdf.
- [2] «MECD,» [En línea]. Disponible: <http://www.mecd.gob.es/cultura-mecd/areas-cultura/museos/visitas-virtuales.html>.
- [3] «Museo del Prado,» [En línea]. Disponible:
<https://www.museodelprado.es/exposiciones/info/en-el-museo/rubens/rubens-360/>.
- [4] «Museo Nacional Centro de Arte Reina Sofía,» [En línea]. Disponible:
<https://www.google.com/culturalinstitute/collection/museo-reina-sofia?hl=es>.
- [5] «Museo Sorolla,» [En línea]. Disponible:
http://museosorolla.mcu.es/visita_virtual/visita_virtual.html.
- [6] «Museo del traje de Madrid,» [En línea]. Disponible:
<http://www.mcu.es/visitavirtualmuseos/museo-del-traje>.
- [7] D. Gonzalez, «Diseño de videojuegos. Da forma a tus sueños,» Ra-Ma, 2011, pp. 15-19.
- [8] «Blender,» [En línea]. Disponible: <http://www.blender.org/>.
- [9] «WikiBlender,» [En línea]. Disponible:
<http://wiki.blender.org/index.php/Doc:ES/2.6/Manual/Introduction>.
- [10] «AutoDesk,» [En línea]. Disponible: <http://www.autodesk.es/products/3ds-max/overview>.
- [11] «AutoDesk,» [En línea]. Disponible: <http://www.autodesk.es/products/maya/overview>.
- [12] «Unity,» [En línea]. Disponible: <http://unity3d.com/>.
- [13] «Unreal Engine,» [En línea]. Disponible: <https://www.unrealengine.com/>.
- [14] «CryEngine,» [En línea]. Disponible: <http://cryengine.com/>.
- [15] D. Gonzalez, «Arte de videojuegos. Da forma a tus sueños,» Ra-Ma, 2014, pp. 130-140.
- [16] M. Carabantes, «Micromanía,» [En línea]. Disponible:

<http://www.micromania.es/blogs/pc-vs-consolas>.

- [17] M. Michán, «Applesfera,» [En línea]. Disponible: <http://www.applesfera.com/accesorios/leap-motion-el-sistema-de-control-gestual-mas-barato-y-preciso-que-kinect-que-dificilmente-triunfara>.
- [18] «LeapMotion,» [En línea]. Disponible: <https://www.leapmotion.com/>.
- [19] «Intel,» [En línea]. Disponible: <http://www.intel.es/content/www/es/es/architecture-and-technology/realsense-overview.html>.
- [20] «NimbleVR,» [En línea]. Disponible: <http://nimblevr.com/>.
- [21] R. C. y. D. Á. Roberto Hornero, «<http://www.fgcsic.es>,» [En línea]. Disponible: http://www.fgcsic.es/lychnos/es_es/articulos/Brain-Computer-Interface-aplicado-al-entrenamiento-cognitivo.
- [22] «OculusVR,» [En línea]. Disponible: <https://www.oculus.com/>.
- [23] A. C, «VidaExtra,» [En línea]. Disponible: <http://www.vidaextra.com/industria/oculus-sale-de-compras-control-por-gestos-vision-y-captura-de-movimientos>.
- [24] D. Gonzalez, «Diseño de videojuegos. Da forma a tus sueños,» Ra-Ma, 2011, pp. 73-78.
- [25] «PEGI,» [En línea]. Disponible: <http://www.pegi.info/es/index/>.
- [26] «AEVI,» [En línea]. Disponible: <http://www.aevi.org.es/>.
- [27] «Agencia Estatal Boletín Oficial del Estado,» [En línea]. Disponible: <https://www.boe.es/buscar/doc.php?id=BOE-A-1999-23750>.
- [28] O. E. y. otros, «Universidad de Palermo,» [En línea]. Disponible: http://fido.palermo.edu/servicios_dyc/publicacionesdc/archivos/272_libro.pdf.
- [29] «Audiovisual451,» [En línea]. Disponible: <http://www.audiovisual451.com/las-330-empresas-desarrolladoras-de-videojuegos-censadas-en-espana-facturaron-3137-millones-de-euros/>.

Anexo I. Acrónimos

4D: O cuatro dimensiones, además de las cualidades del espacio, que son largo, ancho y profundidad (3D), también cuenta con la cualidad de la cuarta dimensión, que es tiempo. Esto quiere decir que el objeto se encuentra en un espacio y en un tiempo real (en su defecto, tiempo virtual).

3D: O tres dimensiones, que son: largo, ancho y profundidad.

AEVI (Asociación Española de Videojuegos): La Asociación aspira a aglutinar a todos los agentes implicados en la cadena de valor del videojuego -desarrolladores, editores, comercializadores, etc. - en una asociación común que nace con el firme propósito de fortalecer y defender los intereses de una de las industrias tecnológicas con mayor proyección en nuestro país.

BLEND: formato que utiliza el software Blender para guardar proyectos realizados con la herramienta.

BOE (Boletín Oficial del Estado): Diario oficial del Estado español dedicado a la publicación de determinadas leyes, disposiciones y actos de inserción obligatoria.

C#: Lenguaje de programación orientado a objetos desarrollado y estandarizado por Microsoft como parte de su plataforma .net.

FBX: es un formato de archivo 3D que proporciona acceso al contenido creado en cualquier paquete de software. Las creaciones de Blender se exportan en este formato, para después ser importadas por Unity3D.

FPS (First Person Shooter): Juego de disparos en primera persona.

GPL: General Public License (Licencia Pública General) es la licencia más ampliamente usada en el mundo del software y garantiza a los usuarios finales (personas, organizaciones, compañías) la libertad de usar, estudiar, compartir (copiar) y modificar el software.

HUD (Head Up Display): Interfaz de comunicación con el jugador durante el juego, todo lo que vemos en pantalla de información: vida, munición, etc.

IA (Inteligencia Artificial): es un área multidisciplinaria que, a través de ciencias como las ciencias de la computación, la lógica y la filosofía, estudia la creación y diseño de entidades capaces de resolver cuestiones por sí mismas utilizando como paradigma la inteligencia humana.

JPG o JPEG: Joint Photographic Experts Group, es un formato gráfico que utiliza habitualmente un algoritmo de compresión con pérdida para reducir el tamaño de los archivos de imágenes. Formato para texturas reconocido por Unity3D.

JS: lenguaje de programación interpretado. Se define como orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico.

LOPD (Ley Organiza de Protección de Datos: Ley Orgánica española que tiene por objeto garantizar y proteger, en lo que concierne al tratamiento de los datos personales.

MMORPG (Massive Multiplayer Online Role-Playing Games): Género de videojuego de rol en donde participan un gran número de jugadores interactuando entre sí en un mundo virtual.

OpenGL (Open Graphics Library): Especificación estándar que define una API multilenguaje y multiplataforma para escribir aplicaciones que produzcan gráficos 2D y 3D.

PEGI (Pan European Game Information): El sistema de clasificación por edades que se estableció con el objeto de ayudar a los progenitores europeos a tomar decisiones informadas a la hora de adquirir juegos de ordenador.

PNG: Portable Network Graphics, es un formato grafico basado en un algoritmo de compresión sin pérdida para bitmaps. Formato para texturas reconocido por Unity3D.

Anexo II. Glosario de términos

3ds Max: Programa de creación de gráficos y animación 3D desarrollado por Autodesk.

AutoRealm: Software gratuito para crear juegos de rol, pero que por su sencillez lo podemos usar para crear croquis de mapas de videojuegos.

Blender: Programa informático multiplataforma, dedicado especialmente al modelado, iluminación, renderizado, animación y creación de gráficos tridimensionales.

CryEngine: Motor de juego creado por la empresa alemana desarrolladora de software Crytek.

DirectX: Colección de API desarrolladas para facilitar las complejas tareas relacionadas con multimedia, especialmente programación de juegos y vídeo, en la plataforma Microsoft Windows.

Diseño 3D: proceso de programar, proyectar, coordinar, seleccionar y organizar una serie de elementos para producir objetos visuales destinados a comunicar mensajes específicos a grupos determinados.

Game Engine: Software diseñado para la creación y desarrollo de videojuegos. Los desarrolladores de videojuegos los utilizan para crear juegos para consolas, dispositivos móviles y ordenadores personales.

Google Art Project: Web que presenta una recopilación de imágenes en alta resolución de obras de arte expuestas en varios museos del mundo.

JavaScript: Lenguaje de script multiplataforma orientado a objetos.

Leap Motion: Dispositivo de control gestual que se coloca frente a la pantalla de nuestro ordenador, es capaz de capturar con una precisión enorme los movimientos de nuestras manos, dedos e incluso objetos.

Maya: Creado por Autodesk, es un software de animación en 3D.

Motor de juego: Término que hace referencia a una serie de rutinas de programación que permiten el diseño, la creación y la representación de un videojuego.

Multiplataforma: Atributo conferido a programas informáticos o métodos y conceptos de cómputo que son implementados e interoperan en múltiples plataformas informáticas.

Oculus Rift: Gafas de realidad virtual que están siendo desarrolladas por Oculus VR.

Python: Lenguaje de programación interpretado cuya filosofía hace hincapié en una sintaxis que favorezca un código legible.

Ray-Casting: Técnica de renderizado de gráficos en 3D en un mapeado 2D que calcula los objetos que son visibles para el jugador mediante la emisión de "rayos", optimizando los recursos de la máquina.

Renderizado: Proceso de generar una imagen o vídeo mediante el cálculo de iluminación GI partiendo de un modelo en 3D.

Shader: Un procedimiento de sombreado e iluminación que permite al diseñador/programador especificar el renderizado de un vertex o de un pixel.

Software propietario: Tipo de licencia, es el antónimo del concepto de software libre, este concepto se aplica a cualquier programa informático que no es libre o que sólo lo es parcialmente (semilibre), sea porque su uso, redistribución o modificación está prohibida, o sea porque requiere permiso expreso del titular del software.

Streaming: Distribución digital de multimedia a través de una red de computadoras de manera que el usuario consume el producto, generalmente archivo de video o audio, en paralelo mientras se descarga.

Unity 3D: Motor de videojuego multiplataforma creado por Unity Technologies.

Unreal Engine: Motor de juego de PC y consolas creados por la compañía Epic Games.

Wiimote: Mando principal de la consola Wii de Nintendo, el cual también es compatible con la consola Wii U.

Anexo III. Metodología de desarrollo Scrum

En este proyecto se ha utilizado una metodología de desarrollo ágil llamada Scrum, muy utilizada en empresas de desarrollo de software en general y de videojuegos en particular.

Consiste en entender el desarrollo software como algo muy complejo y con posibilidad de sufrir muchas variaciones, con lo que nos ofrece soluciones para ir corrigiendo nuestro proyecto en todo momento y adaptarlo a las necesidades.

Scrum trata de que todo el equipo conozca perfectamente en qué estado va el proyecto, qué falta por desarrollar y mostrar lo que ya llevamos hecho. Esto se consigue a través de reuniones de control, en las que no sólo se le explica al jefe lo que se ha hecho, sino que se le explica a todo el equipo. Tras una reunión de control, todo el mundo debe tener una idea muy clara del estado del proyecto.

Un Scrum lo forman diversos conceptos que debemos desarrollar y trabajar:

- Product Backlog: documento en el cual se detallan, de forma priorizada, los requisitos del sistema. Detallado en el apartado [Requisitos software](#).
- Sprint: asignación de cada una de las tareas para una fecha concreta. Detallado en el apartado [Planificación](#).
- ProductOwner: es el líder del equipo, la persona que tiene que estar al tanto de todos los problemas que surjan. En nuestro proyecto este es el jefe de proyecto.
- ScrumMaster: es la persona encargada de conseguir que se cumplan los Sprint, de esto también se encargará el jefe de proyecto.
- Team: el equipo, es nuestro proyecto está formado por el jefe de proyecto, un analista, un diseñador, un programador y un tester.
- Reunión de planificación: se desarrollan al comienzo de un Sprint, una vez que el equipo decida todo lo que hay que hacer en ese Sprint.
- Reunión diaria: es una reunión rápida en la que se suelen hacer 3 preguntas, ¿Qué hiciste ayer?, ¿Qué problemas te has encontrado? y ¿Qué vas a hacer hoy? De esta forma todo el equipo se mantiene al tanto del estado y evolución del proyecto.
- Reunión de revisión: se trata de revisar todo lo que se ha logrado hacer al final del Sprint para tomar decisiones.

Anexo IV. Requisitos del sistema

En este apartado se declara la lista de requisitos de sistema.

Plantilla

<i>Identificador</i>	<i>RSF-X / RSNF-X</i>
<i>Nombre</i>	
<i>Descripción</i>	
<i>Prioridad</i>	
<i>Claridad</i>	
<i>Estabilidad</i>	
<i>Verificabilidad</i>	
<i>Necesidad</i>	
<i>Fuente</i>	

Tabla 14. Plantilla requisitos software funcionales y no funcionales

Donde cada campo significa:

- Identificador: Código numérico único que identifica cada requisito.
- Nombre: nombre descriptivo del requisito.
- Descripción: explicación breve, clara y concisa del requisito.
- Prioridad: nivel de importancia o urgencia de un requisito, a mayor prioridad, mayor necesidad de cumplirlo cuanto antes.
- Estabilidad: probabilidad de que el requisito cambie. A mayor estabilidad, la probabilidad de que el requisito se vea modificado es menor.
- Claridad: a mayor claridad, menor ambigüedad tendrá el requisito y menos posibilidades de malinterpretación.
- Verificabilidad: facilidad que posee el requisito de comprobar su correcta aplicación.
- Necesidad: nivel de importancia para el cliente que tiene el requisito.
- Fuente: origen del requisito, persona (cliente) o documentación.

Requisitos software funcionales

Identificador *RSF-01*

<i>Nombre</i>	Menú principal
<i>Descripción</i>	La aplicación deberá tener un menú principal desde el que se puede seleccionar: <ul style="list-style-type: none"> • Visita virtual • Visita virtual alternativa • Salir
<i>Prioridad</i>	Alta
<i>Claridad</i>	Alta
<i>Estabilidad</i>	Alta
<i>Verificabilidad</i>	Alta
<i>Necesidad</i>	Básica
<i>Fuente</i>	Tutor

Tabla 15. RSF-01. Menú principal

Identificador *RSF-02*

<i>Nombre</i>	Menú visita virtual
<i>Descripción</i>	La aplicación deberá tener un menú visita virtual desde el que se puede seleccionar: <ul style="list-style-type: none"> • Leap Motion • Volver (al menú principal)
<i>Prioridad</i>	Alta
<i>Claridad</i>	Alta
<i>Estabilidad</i>	Alta
<i>Verificabilidad</i>	Alta
<i>Necesidad</i>	Básica
<i>Fuente</i>	Tutor

Tabla 16. RSF-02. Menú visita virtual

Identificador *RSF-03*

<i>Nombre</i>	Menú visita virtual alternativa
<i>Descripción</i>	La aplicación deberá tener un menú de visita virtual alternativa desde el que se puede seleccionar: <ul style="list-style-type: none"> • PC o mando • Leap Motion • Volver (al menú principal)
<i>Prioridad</i>	Alta
<i>Claridad</i>	Alta
<i>Estabilidad</i>	Alta
<i>Verificabilidad</i>	Alta
<i>Necesidad</i>	Básica
<i>Fuente</i>	Tutor

Tabla 17. RSF-03. Menú visita virtual alternativa

<i>Identificador</i>	<i>RSF-04</i>
<i>Nombre</i>	Sistema de juego FPS
<i>Descripción</i>	El tipo de juego a desarrollar tiene que ser un FPS (First Person Shooter), para las modalidades de juego: <ul style="list-style-type: none"> • Visita virtual alternativa: <ul style="list-style-type: none"> ○ PC o mando
<i>Prioridad</i>	Alta
<i>Claridad</i>	Alta
<i>Estabilidad</i>	Alta
<i>Verificabilidad</i>	Alta
<i>Necesidad</i>	Básica
<i>Fuente</i>	Tutor

Tabla 18. RSF-04. Sistema de juego FPS, para las modalidades PC o mando

<i>Identificador</i>	<i>RSF-05</i>
<i>Nombre</i>	Sistema de juego FP
<i>Descripción</i>	El tipo de juego a desarrollar tiene que ser un FP (First Person), para las modalidades de juego: <ul style="list-style-type: none"> • Visita virtual y visita virtual alternativa: <ul style="list-style-type: none"> ○ Leap Motion ○ Leap Motion y Oculus Rift
<i>Prioridad</i>	Alta
<i>Claridad</i>	Alta
<i>Estabilidad</i>	Alta
<i>Verificabilidad</i>	Alta
<i>Necesidad</i>	Básica
<i>Fuente</i>	Tutor

Tabla 19. RSF-05. Sistema de juego FP, para las modalidades que utilizan Leap Motion

<i>Identificador</i>	<i>RSF-06</i>
<i>Nombre</i>	Iniciar una nueva visita virtual, con Leap Motion
<i>Descripción</i>	El sistema debe permitir al usuario iniciar una nueva partida, cuando el usuario lo desee
<i>Prioridad</i>	Alta
<i>Claridad</i>	Alta
<i>Estabilidad</i>	Alta
<i>Verificabilidad</i>	Alta
<i>Necesidad</i>	Básica
<i>Fuente</i>	Tutor

Tabla 20. RSF-06. Iniciar una nueva visita virtual, con Leap Motion

<i>Identificador</i>	<i>RSF-07</i>
<i>Nombre</i>	Iniciar una nueva visita virtual, con el Leap Motion y las Oculus Rift
<i>Descripción</i>	El sistema debe permitir al usuario iniciar una nueva partida, cuando el usuario lo desee
<i>Prioridad</i>	Alta
<i>Claridad</i>	Alta
<i>Estabilidad</i>	Alta
<i>Verificabilidad</i>	Alta
<i>Necesidad</i>	Básica
<i>Fuente</i>	Tutor

Tabla 21. RSF-07. Iniciar una nueva visita virtual, con el Leap Motion y las Oculus Rift

<i>Identificador</i>	<i>RSF-08</i>
<i>Nombre</i>	Iniciar una nueva visita virtual alternativa, con PC o mando de consola
<i>Descripción</i>	El sistema debe permitir al usuario iniciar una nueva partida, cuando el usuario lo desee
<i>Prioridad</i>	Alta
<i>Claridad</i>	Alta
<i>Estabilidad</i>	Alta
<i>Verificabilidad</i>	Alta
<i>Necesidad</i>	Básica
<i>Fuente</i>	Tutor

Tabla 22. RSF-08. Iniciar una nueva visita virtual alternativa, con PC o mando de consola

<i>Identificador</i>	<i>RSF-09</i>
<i>Nombre</i>	Iniciar una nueva visita virtual alternativa, con el Leap Motion
<i>Descripción</i>	El sistema debe permitir al usuario iniciar una nueva partida, cuando el usuario lo desee
<i>Prioridad</i>	Alta
<i>Claridad</i>	Alta
<i>Estabilidad</i>	Alta
<i>Verificabilidad</i>	Alta
<i>Necesidad</i>	Básica
<i>Fuente</i>	Tutor

Tabla 23. RSF-09. Iniciar una nueva visita virtual alternativa, con el Leap Motion

<i>Identificador</i>	<i>RSF-10</i>
<i>Nombre</i>	Mostrar escenario Museo Príncipe Felipe
<i>Descripción</i>	El mapa principal de la visita virtual debe estar inspirado en el Museo de las ciencias Príncipe Felipe, de Valencia
<i>Prioridad</i>	Alta
<i>Claridad</i>	Alta
<i>Estabilidad</i>	Alta
<i>Verificabilidad</i>	Alta
<i>Necesidad</i>	Básica
<i>Fuente</i>	Tutor

Tabla 24. RSF-10. Mostrar escenario Museo Príncipe Felipe

<i>Identificador</i>	<i>RSF-11</i>
<i>Nombre</i>	Desplazamiento personaje básico
<i>Descripción</i>	Para las modalidades de juego: <ul style="list-style-type: none"> • Visita virtual y visita virtual alternativa: <ul style="list-style-type: none"> ○ Teclado y ratón → teclado ○ Mando de consola → joystick
<i>Prioridad</i>	Alta
<i>Claridad</i>	Alta
<i>Estabilidad</i>	Alta
<i>Verificabilidad</i>	Alta
<i>Necesidad</i>	Básica
<i>Fuente</i>	Tutor

Tabla 25. RSF-11. Desplazamiento personaje básico

<i>Identificador</i>	<i>RSF-12</i>
<i>Nombre</i>	Desplazamiento personaje avanzado
<i>Descripción</i>	Para las modalidades de juego: <ul style="list-style-type: none"> • Visita virtual y visita virtual alternativa: <ul style="list-style-type: none"> ○ Leap Motion ○ Leap Motion y Oculus Rift <p>En ambos casos el personaje se mueve con el uso de las manos</p>
<i>Prioridad</i>	Alta
<i>Claridad</i>	Alta
<i>Estabilidad</i>	Alta
<i>Verificabilidad</i>	Alta
<i>Necesidad</i>	Básica
<i>Fuente</i>	Tutor

Tabla 26. RSF-12. Desplazamiento personaje avanzado

<i>Identificador</i>	<i>RSF-13</i>
<i>Nombre</i>	ESC sales de la pantalla de juego
<i>Descripción</i>	Al pulsar ESC, desde la partida de juego, exceptuando la de Leap Motion y Oculus Rift, se vuelve al menú principal
<i>Prioridad</i>	Alta
<i>Claridad</i>	Alta
<i>Estabilidad</i>	Alta
<i>Verificabilidad</i>	Alta
<i>Necesidad</i>	Básica
<i>Fuente</i>	Tutor

Tabla 27. RSF-13. ESC sales de la pantalla de juego

<i>Identificador</i>	<i>RSF-14</i>
<i>Nombre</i>	ESC Oculus Rift sales del juego
<i>Descripción</i>	Al pulsar ESC, desde la partida de juego de Leap Motion y Oculus Rift, se sale del juego
<i>Prioridad</i>	Alta
<i>Claridad</i>	Alta
<i>Estabilidad</i>	Alta
<i>Verificabilidad</i>	Alta
<i>Necesidad</i>	Básica
<i>Fuente</i>	Tutor

Tabla 28. RSF-14. ESC Oculus Rift sales del juego

Requisitos software no funcionales

<i>Identificador</i>	<i>RSNF-01</i>
<i>Nombre</i>	Compatibilidad con resoluciones
<i>Descripción</i>	La aplicación deberá ser compatible con distintas resoluciones de pantalla
<i>Prioridad</i>	Alta
<i>Claridad</i>	Alta
<i>Estabilidad</i>	Alta
<i>Verificabilidad</i>	Alta
<i>Necesidad</i>	Esencial
<i>Fuente</i>	Desarrollador

Tabla 29. RSNF-01. Compatibilidad con resoluciones

<i>Identificador</i>	<i>RSNF-02</i>
<i>Nombre</i>	Ficheros ejecutables
<i>Descripción</i>	La aplicación deberá tener dos ejecutables, uno para la interacción con el Leap Motion y las Oculus Rift, y otro para el resto de interacciones: Leap Motion y PC
<i>Prioridad</i>	Alta
<i>Claridad</i>	Alta
<i>Estabilidad</i>	Alta
<i>Verificabilidad</i>	Alta
<i>Necesidad</i>	Esencial
<i>Fuente</i>	Desarrollador

Tabla 30. RSNF-02. Ficheros ejecutables

<i>Identificador</i>	<i>RSNF-03</i>
<i>Nombre</i>	Música y efectos de sonido
<i>Descripción</i>	El sistema deberá utilizar música y efectos de sonido.
<i>Prioridad</i>	Alta
<i>Claridad</i>	Alta
<i>Estabilidad</i>	Alta
<i>Verificabilidad</i>	Alta
<i>Necesidad</i>	Deseable
<i>Fuente</i>	Tutor

Tabla 31. RSNF-03. Música y efectos de sonido

Anexo V. Casos de uso

A continuación se muestran los detalles de cada uno de los casos de uso contenidos en el catálogo de casos de uso. Para ello se va a utilizar el siguiente modelo de tabla:

Plantilla

<i>Identificador</i>	<i>CU-X</i>
<i>Nombre</i>	
<i>Descripción</i>	
<i>Actor</i>	
<i>Precondiciones</i>	
<i>Postcondiciones</i>	
<i>Escenario</i>	

Tabla 32. Plantilla casos de uso

Donde cada elemento significa:

- Identificador: Numero único que identifica cada caso de uso.
- Nombre: título descriptivo del caso de uso.
- Descripción: explicación clara y concisa del caso de uso.
- Actor: tipo de usuario, en nuestro caso siempre será el jugador.
- Precondiciones: condiciones que se deben cumplir antes de ejecutarse el caso de uso.
- Postcondiciones: condiciones que debe cumplir el sistema al concluir el caso de uso.
- Escenario: etapas del caso de uso.

<i>Identificador</i>	<i>CU-01</i>
<i>Nombre</i>	Visita virtual
<i>Descripción</i>	El jugador tiene como primera opción una visita virtual
<i>Actor</i>	Jugador
<i>Precondiciones</i>	Ninguna
<i>Postcondiciones</i>	Ejecuta el menú visita virtual
<i>Escenario</i>	El jugador inicia la aplicación y accede al menú principal, donde, tras seleccionar la opción <i>visita virtual</i> se cargara el menú de visita virtual.

Tabla 33. CU-01. Visita virtual

<i>Identificador</i>	<i>CU-02</i>
<i>Nombre</i>	Visita virtual alternativa
<i>Descripción</i>	El jugador tiene como segunda opción una visita virtual alternativa
<i>Actor</i>	Jugador
<i>Precondiciones</i>	Ninguna
<i>Postcondiciones</i>	Ejecuta el menú visita virtual alternativa
<i>Escenario</i>	El jugador inicia la aplicación y accede al menú principal, donde, tras seleccionar la opción <i>visita virtual alternativa</i> se cargara el menú de visita virtual alternativa.

Tabla 34. CU-02. Visita virtual alternativa

<i>Identificador</i>	<i>CU-03</i>
<i>Nombre</i>	Salir de la aplicación
<i>Descripción</i>	El jugador tiene como tercera opción salir de la aplicación
<i>Actor</i>	Jugador
<i>Precondiciones</i>	La aplicación tiene que estar iniciada
<i>Postcondiciones</i>	Cierra la aplicación
<i>Escenario</i>	El jugador inicia la aplicación y accede al menú principal, donde, tras seleccionar la opción <i>salir</i> se cierra la aplicación

Tabla 35. CU-03. Salir de la aplicación

<i>Identificador</i>	<i>CU-04</i>
<i>Nombre</i>	Visita virtual con Leap Motion
<i>Descripción</i>	El jugador tiene como primera opción iniciar una visita virtual, para jugar con el Leap Motion
<i>Actor</i>	Jugador
<i>Precondiciones</i>	La aplicación tiene que estar iniciada y seleccionado el menú de visita virtual en el menú principal
<i>Postcondiciones</i>	Inicia la aplicación para la visita virtual con el Leap Motion
<i>Escenario</i>	El Jugador tras acceder al menú de visita virtual desde el menú principal y seleccionar la opción <i>Leap Motion</i> , se carga la aplicación para la visita virtual con el Leap Motion

Tabla 36. CU-04. Visita virtual con Leap Motion

<i>Identificador</i>	CU-05
<i>Nombre</i>	Visita virtual con Leap Motion y Oculus Rift
<i>Descripción</i>	El jugador tiene como segunda opción iniciar una visita virtual, para jugar con el Leap Motion y las Oculus Rift
<i>Actor</i>	Jugador
<i>Precondiciones</i>	La aplicación tiene que estar iniciada y seleccionado el menú de visita virtual en el menú principal
<i>Postcondiciones</i>	Inicia la aplicación para la visita virtual con el Leap Motion y las Oculus Rift
<i>Escenario</i>	El Jugador tras acceder al menú de visita virtual desde el menú principal y seleccionar la opción <i>Leap Motion y Oculus Rift</i> , se carga la aplicación para la visita virtual con el Leap Motion y las Oculus Rift

Tabla 37. CU-05. Visita virtual con Leap Motion y Oculus Rift

<i>Identificador</i>	CU-06
<i>Nombre</i>	Cargar menú principal desde el menú visita virtual
<i>Descripción</i>	El jugador tiene como tercera opción volver al menú principal
<i>Actor</i>	Jugador
<i>Precondiciones</i>	La aplicación tiene que estar iniciada y seleccionado el menú de visita virtual en el menú principal
<i>Postcondiciones</i>	Carga el menú principal
<i>Escenario</i>	El Jugador tras acceder al menú de visita virtual desde el menú principal y seleccionar la opción <i>volver</i> , se carga el menú principal

Tabla 38. CU-06. Cargar menú principal desde el menú visita virtual

<i>Identificador</i>	CU-07
<i>Nombre</i>	Visita virtual alternativa con PC o mando
<i>Descripción</i>	El jugador tiene como primera opción iniciar una visita virtual alternativa, para jugar con el PC (teclado y ratón) o mando
<i>Actor</i>	Jugador
<i>Precondiciones</i>	La aplicación tiene que estar iniciada y seleccionado el menú de visita virtual alternativa en el menú principal
<i>Postcondiciones</i>	Inicia la aplicación para la visita virtual alternativa con el Leap Motion
<i>Escenario</i>	El Jugador tras acceder al menú de visita virtual alternativa desde el menú principal y seleccionar la opción <i>PC o mando</i> , se carga la aplicación para la visita virtual alternativa con el PC (teclado y ratón) o mando

Tabla 39. CU-07. Visita virtual alternativa con PC o mando

<i>Identificador</i>	CU-08
<i>Nombre</i>	Visita virtual alternativa con Leap Motion
<i>Descripción</i>	El jugador tiene como segunda opción iniciar una visita virtual alternativa, para jugar con el Leap Motion
<i>Actor</i>	Jugador
<i>Precondiciones</i>	La aplicación tiene que estar iniciada y seleccionado el menú de visita virtual alternativa en el menú principal
<i>Postcondiciones</i>	Inicia la aplicación para la visita virtual alternativa con el Leap Motion
<i>Escenario</i>	El Jugador tras acceder al menú de visita virtual alternativa desde el menú principal y seleccionar la opción <i>Leap Motion</i> , se carga la aplicación para la visita virtual alternativa con el Leap Motion

Tabla 40. CU-08. Visita virtual alternativa con Leap Motion

<i>Identificador</i>	CU-9
<i>Nombre</i>	Cargar menú principal desde el menú visita virtual alternativa
<i>Descripción</i>	El jugador tiene como cuarta opción volver al menú principal
<i>Actor</i>	Jugador
<i>Precondiciones</i>	La aplicación tiene que estar iniciada y seleccionado el menú de visita virtual alternativa en el menú principal
<i>Postcondiciones</i>	Carga el menú principal
<i>Escenario</i>	El Jugador tras acceder al menú de visita virtual alternativa desde el menú principal y seleccionar la opción <i>volver</i> , se carga el menú principal

Tabla 41. CU-09. Cargar menú principal desde el menú visita virtual alternativa

<i>Identificador</i>	CU-10
<i>Nombre</i>	Mover cámara
<i>Descripción</i>	El jugador tiene la posibilidad de mover la cámara, usando: <ul style="list-style-type: none"> • PC: Ratón • Mando: Joystick derecho • Leap Motion: mano izquierda
<i>Actor</i>	Jugador
<i>Precondiciones</i>	La aplicación ha de estar iniciada y cargada la partida
<i>Postcondiciones</i>	Mueve la cámara alrededor del personaje
<i>Escenario</i>	El Jugador tras acceder al juego, puede mover la cámara

Tabla 42. CU-10. Mover cámara

<i>Identificador</i>	<i>CU-11</i>
<i>Nombre</i>	Mover personaje
<i>Descripción</i>	El jugador tiene la posibilidad de mover el personaje, usando: <ul style="list-style-type: none"> • PC: teclas (W,S,A,D o las flechas de dirección) • Mando: Joystick izquierdo • Leap Motion: mano izquierda
<i>Actor</i>	Jugador
<i>Precondiciones</i>	La aplicación ha de estar iniciada y cargada la partida
<i>Postcondiciones</i>	Mueve el personaje por el escenario
<i>Escenario</i>	El Jugador tras acceder al juego, puede mover el personaje

Tabla 43. CU-11. Mover personaje

<i>Identificador</i>	<i>CU-12</i>
<i>Nombre</i>	Disparar
<i>Descripción</i>	El jugador tiene la posibilidad de mover el personaje, usando: <ul style="list-style-type: none"> • PC: ratón (click izquierdo) • Mando: R1 • Leap Motion: mano derecha
<i>Actor</i>	Jugador
<i>Precondiciones</i>	La aplicación ha de estar iniciada y cargada la partida
<i>Postcondiciones</i>	Mueve el personaje por el escenario
<i>Escenario</i>	El Jugador tras acceder al juego, puede mover el personaje

Tabla 44. CU-12. Disparar

<i>Identificador</i>	<i>CU-13</i>
<i>Nombre</i>	Salir de la pantalla de juego, al menú principal con ESC
<i>Descripción</i>	El jugador pulsando ESC sale de la partida, y se carga el menú principal
<i>Actor</i>	Jugador
<i>Precondiciones</i>	La aplicación ha de estar iniciada y cargada la partida
<i>Postcondiciones</i>	Carga el menú principal
<i>Escenario</i>	El jugador inicia la aplicación y accede al menú principal, donde, tras pulsar el botón ESC se carga el menú principal

Tabla 45. CU-13. Salir de la pantalla de juego, al menú principal con ESC

<i>Identificador</i>	<i>CU-14</i>
<i>Nombre</i>	Salir del juego con ESC para las Oculus Rift
<i>Descripción</i>	El jugador pulsando ESC salir de la partida y de la aplicación
<i>Actor</i>	Jugador
<i>Precondiciones</i>	La aplicación ha de estar iniciada y cargada la partida
<i>Postcondiciones</i>	Salir de la aplicación
<i>Escenario</i>	El jugador inicia la aplicación y accede a la partida, tras pulsar el botón ESC se sale de la aplicación

Tabla 46. CU-14. Salir del juego con ESC para las Oculus Rift

Anexo VI. Gestión de datos

A continuación se va a detallar como se han gestionado los datos que hacen uso de cada una de las herramientas del proyecto. Al utilizar Unity3D todas las nuevas carpetas se almacenan dentro de la carpeta Assets, que se crea automáticamente cuando se inicia el proyecto por primera vez en Unity 3D. Se puede ver un esquema en la Ilustración 115.



Ilustración 115. Gestión de datos

Carpeta **Animaciones**: contiene las animaciones asociadas a los personajes u objetos.

Carpeta **Escena**: contiene las distintas escenas del juego, en Unity3D, cada menú se considera una escena diferente.

Carpeta **Imágenes:** contiene las imágenes usadas para crear todas las texturas del juego, los fondos y diseño de los botones de los menús.

Carpeta **LeapMotion:** contiene el material necesario para el uso de Leap Motion, se crea al importar el plugin.

Carpeta **Objetos:** contiene todos los modelos creados en Blender (.fbx).

Carpeta **OVR:** contiene el material necesario para el uso de Oculus Rift, se crea al importar el plugin.

Carpeta **Scripts:** contiene toda la lógica del juego, así como los controladores de los distintos modos de interacción.

Carpeta **Sonidos:** contiene la música y los sonidos utilizados.

Carpeta **VRGUI:** contiene el material necesario para que los textos sean visibles con las Oculus Rift, se crea al importar el plugin.

Anexo VII. Manual de usuario

En este apartado se va a detallar que pasos hay que realizar para utilizar el software creado, y como se utiliza cada forma de interacción, en este proyecto concreto.

Primeros pasos

Para el funcionamiento del programa existen dos ejecutables:

- Museo.exe
En este ejecutable se encuentra tanto la visita virtual como la visita virtual alternativa para las opciones de PC o mando y Leap Motion.
- MuseoVisitaOculusRift_DirectToRift.exe
Este ejecutable muestra directamente el proyecto en una pantalla extendida, para su correcta visualización en las Oculus Rift.

Interacción con el PC

Si se selecciona la opción de realizar la visita virtual con PC, se utilizará el teclado y el ratón, por lo que el personaje se controlaría mediante scripts de la siguiente forma:

- Mover cámara: movimientos de ratón.
- Mover Personaje:
 - Delante: W y flecha arriba.
 - Atrás: S y flecha abajo.
 - Izquierda: D y flecha derecha.
 - Derecha: A y flecha izquierda.
- Disparar: botón izquierdo ratón. Esta opción sólo está disponible para la visita virtual alternativa.

Interacción con un mando de consola

Para la interacción con el mando, es necesario instalar el programa “Better DS3”, disponible de forma gratuita en su página web:

<http://betterds3.ciebiera.net/>.

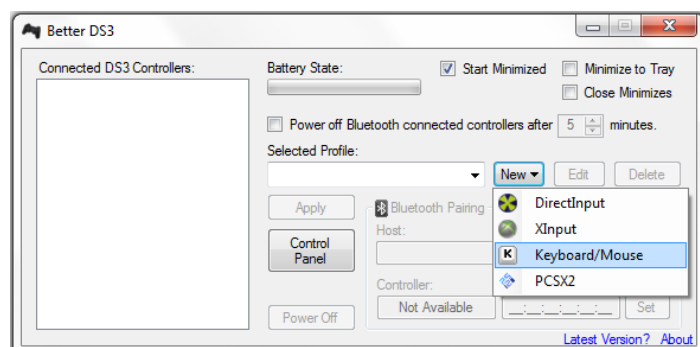


Ilustración 116. Interfaz Better DS3, selección tipo de archivo

Además requiere la siguiente configuración:

1. Pulsamos “New”, nos aparecerá un desplegable que se observa en la Ilustración 116, en el que seleccionamos Keyboard/Mouse,

2. Rellenamos las distintas opciones como se indica en la Ilustración 117.

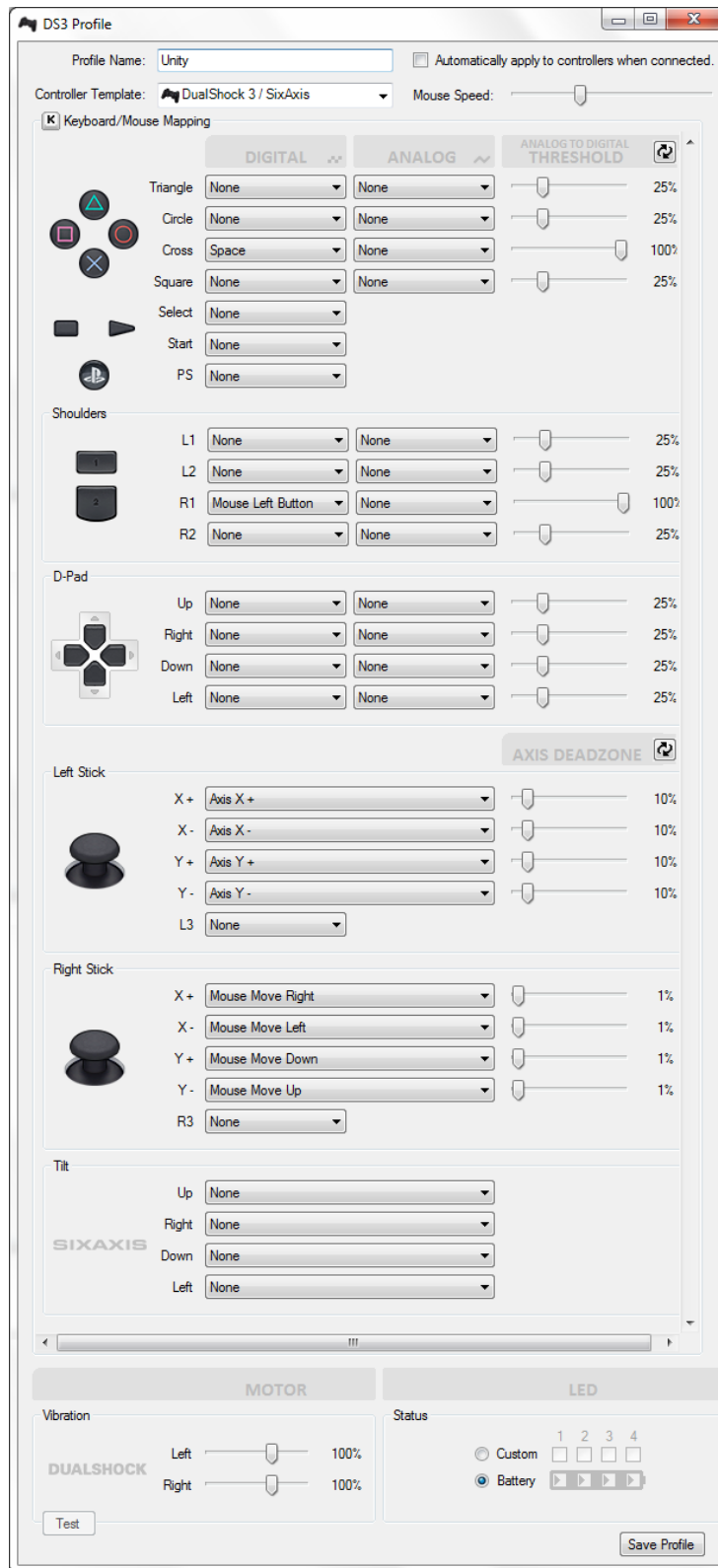


Ilustración 117. Interfaz Better DS3, configuración de acciones

3. Conectamos el mando mediante un cable usb al ordenador.
4. Pulsamos “Control Panel”, y seleccionamos el mando como dispositivo el mando que acabamos de conectar, Ilustración 118.

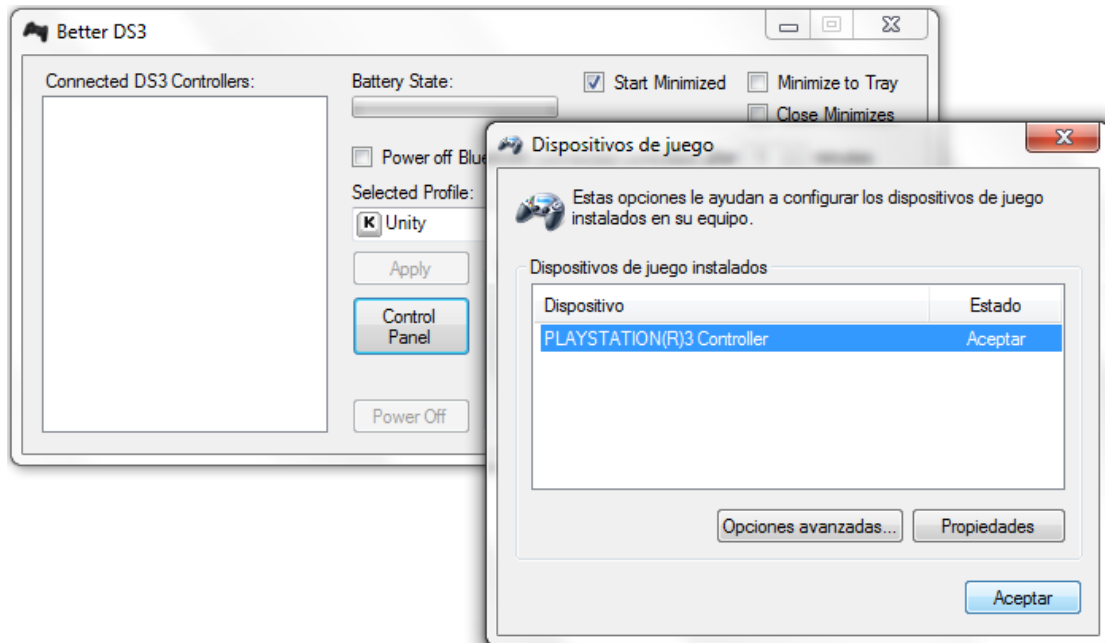


Ilustración 118. Interfaz Better DS3, asociar mando

5. Seleccionamos la opción de realizar la visita virtual con mando.

El personaje se controlaría mediante scripts de la siguiente forma:

- Mover cámara: joystick derecho.
- Mover Personaje: joystick izquierdo.
- Disparar: R1

Interacción con Leap Motion

Si se selecciona la opción de realizar la visita virtual con Leap Motion, el personaje se controlaría mediante scripts de la siguiente forma:

- Mover cámara: girar a la izquierda o derecha, la muñeca izquierda.
- Mover Personaje:
 - Delante: cerrar mano izquierda.
 - Atrás: no tiene esta opción, en su lugar deberá girar la cámara hacia uno de los lados y andar hacia delante.
 - Izquierda: no tiene esta opción, en su lugar deberá girar la cámara a la izquierda y andar hacia delante.
 - Derecha: no tiene opción, en su lugar deberá girar la cámara a la derecha y andar hacia delante.
- Disparar: abrir mano derecha. Esta opción sólo está disponible para la visita virtual alternativa.
- Recargar energía de disparo: cerrar mano derecha. Esta opción sólo está disponible para la visita virtual alternativa.

Interacción con Leap Motion y Oculus Rift

Si se selecciona la opción de realizar la visita virtual con Leap Motion y Oculus Rift, el personaje se controlaría exactamente igual que en el caso anterior, en el que sólo se utiliza Leap Motion, ya que el controlador es el mismo.

Anexo VIII. Análisis de la planificación y coste del proyecto tras su finalización

A continuación se muestra como ha sido la planificación real del proyecto. En ningún momento se han modificado las tareas que se tenían que realizar y se han realizado todas satisfactoriamente, aunque en más tiempo del planeado en un inicio.

Planificación real:

Para el proyecto se había estimado un total de 100 días, aunque finalmente se han necesitado 120 días, en la Ilustración 119 se muestra el tiempo real que se ha necesitado para cada una de las tareas, cada periodo representa 2 días.

Planificación real

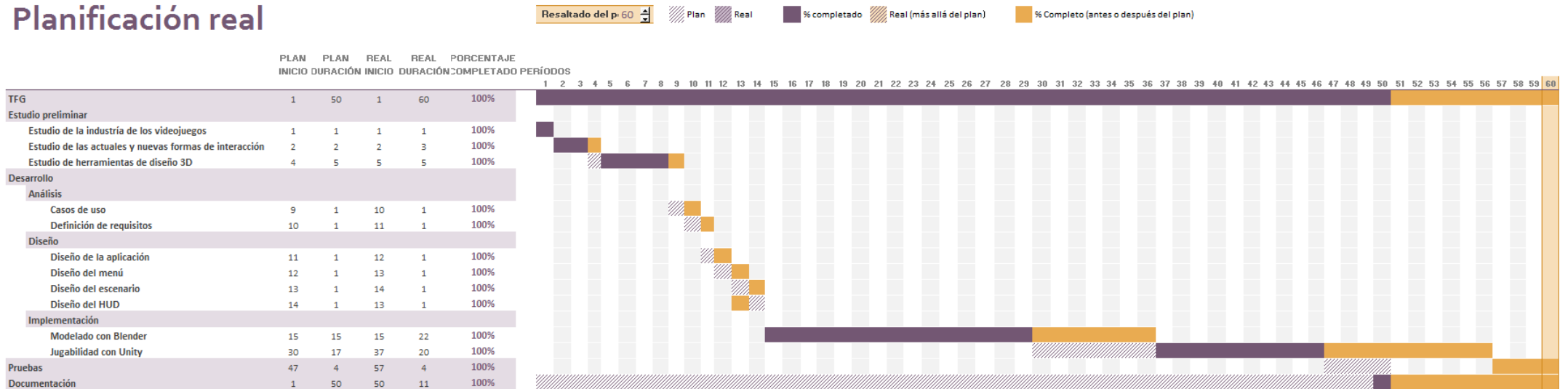


Ilustración 119. Planificación real. 1 periodo equivale a 2 días

A continuación se muestra el recalcu­lo del coste del proyecto utilizando la planificación real.

Recursos humanos

Para realizar una estimación correcta, para cada puesto se ha calculado el coste por día, tomando como base que en un mes trabajarían 20 días, obteniendo como resultado los costes indicados en la Tabla 47.

<i>Puesto</i>	<i>Dedicación (nº días)</i>	<i>Coste/día</i>	<i>Total</i>
<i>Jefe de proyecto</i>	18	164,45 €	2.960,10 €
<i>Analista</i>	4	108,00 €	432,00 €
<i>Diseñador</i>	6	110,00 €	660,00 €
<i>Programador</i>	84	125,00 €	10.500,00 €
<i>Tester</i>	8	50,00 €	400,00 €
TOTAL	120		14.952,10 €

Tabla 47. Coste real recursos humanos

Recursos materiales

Para la realización del proyecto se ha realizado una estimación de los costes materiales, se han dividido en dos categorías hardware y software, Tabla 48 y Tabla 49.

Hardware

<i>Descripción</i>	<i>Coste</i>	<i>% de uso</i>	<i>Dedicación</i>	<i>Depreciación</i>	<i>Coste imputable</i>
<i>PC Intel I7</i>	1.299,00 €	100%	5	60	108,25 €
<i>Mando PS3</i>	57,69 €	100%	1	60	0,96 €
<i>Leap Motion con montura para Oculus Rift</i>	96,99 €	100%	1	60	1,62 €
<i>Oculus Rift</i>	350,00 €	100%	1	60	5,83 €
TOTAL					116,66 €

Tabla 48. Coste real recursos materiales hardware

Software

<i>Descripción</i>	<i>Coste</i>	<i>% de uso</i>	<i>Dedicación</i>	<i>Depreciación</i>	<i>Coste imputable</i>
<i>Windows 7</i>	79,00 €	100%	3	60	3,95 €
<i>Microsoft Office</i>	79,00 €	100%	3	60	3,95 €
TOTAL					7,90 €

Tabla 49. Coste real recursos materiales software

Habiendo estimado los recursos necesarios para el correcto desarrollo del proyecto, el presupuesto final se desglosa en la Tabla 50.

<i>Descripción</i>	<i>Coste</i>
<i>Recursos humanos</i>	14.952,10 €
<i>Recursos materiales hardware</i>	116,66 €
<i>Recursos materiales software</i>	7,90 €
TOTAL	15.076,66 €

Tabla 50. Coste real total del presupuesto

El coste estimado del proyecto era de 12.467,76 €, añadiéndole un riesgo del 25% cuyo valor era de 3.116,94 €. Los gastos del proyecto se han incrementado en 2.608,90 €, pero este valor está por debajo del riesgo con el que habíamos contado. Por lo que los beneficios iniciales que se habían marcado, son beneficios reales, ya que no se han tocado para cubrir gastos. Todo ello representado en la Tabla 51. Por lo que desde el punto de vista económico, el proyecto se ha resuelto satisfactoriamente.

<i>Descripción</i>	<i>Coste</i>
TOTAL ESTIMADO	12.467,76 €
<i>Riesgo (25%)</i>	3.116,94 €
TOTAL REAL	15.076,66 €
REAL - ESTIMADO	2.608,90 €

Tabla 51. Calculo de ganancias/perdidas del proyecto