UNIVERSITAT POLITÈCNICA DE VALÈNCIA

_ TELECOM ESCUELA TÉCNICA VLC SUPERIOR DE INGENIERÍA DE TELECOMUNICACIÓN

DESARROLLO DE UNA VERSIÓN VIRTUAL DEL MUSEO DE LA TELECOMUNICACIÓN VICENTE MIRALLES SEGARRA

Alejandro Vecina Dolz

Tutor: Joaquín Cerdá Boluda

Cotutor: María Carmen Bachiller Martín

Trabajo Fin de Grado presentado en la Escuela Técnica Superior de Ingenieros de Telecomunicación de la Universitat Politècnica de València, para la obtención del Título de Graduado en Ingeniería de Tecnologías y Servicios de Telecomunicación

Curso 2018-19

Valencia, 3 de julio de 2019

Escuela Técnica Superior de Ingeniería de Telecomunicación
Universitat Politècnica de València
Edificio 4D. Camino de Vera, s/n, 46022 Valencia
Tel. +34 96 387 71 90, ext. 77190
www.etsit.upv.es

VLC/ CAMPUS VALENCIA, INTERNATIONAL CAMPUS OF EXCELLENCE

EMAS Gestión ambiental verificada REG.NO. ES-CV- 000030

Resumen

En este Trabajo Fin de Grado, se detalla cómo se ha desarrollado una versión virtual del Museo de la Telecomunicación Vicente Miralles Segarra para su uso en dispositivos de realidad virtual como son las gafas 3D Oculus Rift.

A lo largo del documento, se incluyen diversas imágenes, enlaces, fechas y otras referencias para su mejor entendimiento.

Primero, se hará una pequeña introducción a las nuevas tecnologías en los museos con un repaso histórico desde sus inicios hasta la actualidad, haciendo hincapié en las invenciones más importantes. Posteriormente, se explicará la realidad virtual, sus características, tipos y distintas aplicaciones de uso.

A continuación, se introducirán los softwares elegidos para el trabajo, como son el motor de desarrollo Unity y el entorno virtual Oculus, para terminar, detallando la aplicación de realidad virtual partiendo del modelado del edificio y sus objetos, la navegación a través de él y terminando con la interacción del usuario con el entorno.

Finalmente, se concluirá con los resultados obtenidos, posibilidad de ampliación y conclusiones del trabajo.

Resum

En aquest Treball Fi de Grau, es detalla com s'ha desenvolupat una versió del Museu de la Telecomunicació Vicente Miralles Segarra per al seu us en dispositius de realitat virtual com són les ulleres 3D Oculus Rift.

Al llarg del document, s'inclouen diverses imatges, enllaços, dates i altres referències per al seu millor enteniment.

Primer, es farà una xicoteta introducció a les noves tecnologies als museus amb un repàs històric des dels seus inicis fins l'actualitat, fent insistència en les invencions més importants. Posteriorment, s'explicarà la realitat virtual, les seues característiques, tipus i distintes aplicacions d'us.

A continuació, s'introduiran els softwares elegits per al treball, com són el motor de desenvolupament Unity i l'entorn virtual Oculus, per a finalitzar detallant l'aplicació de realitat virtual partint del modelat de l'edifici i els seus objectes, la navegació a través d'ell i acabant amb la interacció de l'usuari amb l'entorn.

Finalment, es conclourà amb els resultats obtinguts, possibilitat d'ampliació i conclusions del treball.

Abstract

In this Degree Final Project, it is detailed how a virtual version of the Vicente Miralles Segarra Telecommunication Museum has been developed for use in virtual reality devices such as the Oculus Rift 3D glasses.

Throughout the document, various images, links, dates and other references are included for better understanding.

First, there will be a short introduction to new technologies in museums with a historical review from its beginnings to the present, emphasizing the most important inventions. Later, the virtual reality, its characteristics, types and different applications will be explained.

Next, the software chosen for the work will be introduced, such as the Unity development engine and the Oculus virtual environment, to finish detailing the virtual reality application based on the modeling of the building and its objects, navigation through it and ending with the user's interaction with the environment.

Finally, there will conclude with the obtained results, possibility of extension and conclusions of the work.

Índice

Capítulo 1. Introducción 3
1.1 Objetivos 3
1.2 Motivación 3
1.3 Plan de trabajo 3
1.4 Estructura 5
Capítulo 2. Museos y nuevas tecnologías 6
2.1 Repaso histórico 6
2.2 Realidad virtual 6
2.2.1 Características y tipos 7
2.2.2 Hardware 8
2.2.3 Historia 8
2.2.4 Aplicaciones 12
Capítulo 3. Unity y Oculus 15
3.1 Descripción 15
3.2 Capacidades 16
3.2.1 Escenas y objetos 16
3.2.2 Importación 16
3.2.3 Programación 17
3.2.4 Oculus Touch Controllers 17
3.2.5 Multiplataforma 18
Capítulo 4. La aplicación de realidad virtual 19
4.1 Descripción 19
4.2 Entorno 21
4.2.1 Interfaz 21

4.2.2 Creación de objetos 22
4.2.3 Modelado y colocación de objetos 25
4.3 Navegación 38
4.3.1 Configuración de las gafas Oculus Rift 38
4.3.2 Integración y uso en Unity 39
4.4 Interacción 45
4.4.1 Contenidos multimedia de realidad aumentada 45
4.4.2 Contenidos multimedia de realidad virtual 46
Capítulo 5. Resultados 54
5.1 Galería de imágenes 54
5.2 Posibilidad de ampliación 68
5.3 Pruebas con usuarios 68
Capítulo 6. Conclusiones y líneas futuras 69
Bibliografía 70
Anexo 72
Tutorial y ejemplo de uso 72

Capítulo 1. Introducción

1.1 Objetivos

El objetivo de este trabajo es el desarrollo de una versión virtual del Museo de la Telecomunicación Vicente Miralles Segarra, ubicado en el edificio de la Escuela Técnica Superior de Ingeniería de la Telecomunicación (ETSIT) en la Universitat Politècnica de València. La aplicación estará diseñada para dispositivos de realidad virtual con el fin de ofrecer una visita del museo sin la necesidad de estar allí presente.

Para lograrlo, uno de los objetivos es conseguir una recreación lo más fiel posible a la realidad tanto del edificio que contiene al museo como de cada uno de los objetos expuestos junto con sus vitrinas, carteles descriptivos y demás componentes. El otro objetivo es el de ofrecer al usuario un uso sencillo e intuitivo de los dispositivos virtuales para facilitarle poder centrarse en la exposición. Con todo ello, se pretende alcanzar una alta sensación de realismo y ofrecer al usuario una experiencia innovadora para complementar o sustituir la visita física al museo.

1.2 Motivación

La palabra museo suele estar ligada al arte y se suele asociar con la antigüedad. Se considera que los primeros museos pudieron surgir ya en la antigua Grecia, pasando por Egipto y teniendo un gran apogeo durante el Renacimiento Italiano. Es por eso que, a día de hoy, abundan tanto los museos de arte antiguo. A esto, cabe añadir que suelen exponerse en edificios también antiguos y que ofrecen una visita clásica en forma de “paseo”, ya sea libre o guiado.

Una de las principales motivaciones de este trabajo es la de poder modernizar dichas exposiciones y visitas y hacer llegar una nueva y muy reciente tecnología como es la realidad virtual al entorno clásico y antiguo de un museo. Esto, aparte de enriquecer y complementar las visitas, puede acercar esa parte de cultura y conocimiento histórico a las nuevas generaciones de personas totalmente adaptadas a la era tecnológica actual.

La otra motivación es la de que, al tratarse de un Museo de la Telecomunicación con los aspectos de modernidad y futuro que esta rama de la ciencia conlleva, no podía faltarle una experiencia tecnológica innovadora como es una visita virtual.

1.3 Plan de trabajo

El proyecto se ha dividido en dos grandes partes claramente diferenciables. Una, para la preparación y puesta a punto de todo el hardware necesario para poder empezar a crear la aplicación de realidad virtual; y, otra, para crearla.

La primera de ellas, ha consistido en la habilitación de un espacio de trabajo donde desarrollar el proyecto adecuando una pequeña habitación de 3x2,5 metros proporcionada por la escuela, la cual se usa de almacén de muchos de los objetos no expuestos del museo. Se ha limpiado, reordenado en estanterías y dejado un espacio libre para poder moverse al usar las gafas de realidad virtual.

Figura 1. Setup de trabajo.

Seguidamente, se ha colocado una mesa, una silla y se ha instalado una buena conexión a Internet, ya que carecía de ella.

En cuanto al hardware, se ha invertido en un ordenador nuevo y potente, tanto en procesador como en tarjeta gráfica, capaz de soportar los altos requerimientos de la realidad virtual. También se han adquirido las Oculus Rift con sus gafas, controladores y sensores. Finalmente, ha sido necesario configurar tanto el ordenador como los softwares a utilizar en el proyecto, como son Unity y Oculus, para dejarlo todo listo y empezar a trabajar.

La segunda parte ha consistido en todo el proceso de diseño, edición y programación de la aplicación de realidad virtual.

El primer paso ha sido la importación del modelo base del edificio de la ETSIT y la adaptación del proyecto para su uso con las gafas de realidad virtual. Luego, ha sido necesario hacer numerosas fotografías, tanto a los objetos que forman parte del museo (vitrinas, carteles, tablones, indicaciones) como a los que forman parte de la escuela (bancos, sillas, carteles, extintores, plantas, papeleras, etc.), todo para tener una referencia de su distribución y poder plasmar en el modelo virtual una copia exacta del interior del edificio.

A continuación, se procede a programar el comportamiento tanto del usuario con el entorno como de los contenidos multimedia disponibles en algunos objetos expuestos. Y, finalmente, se comprueba el correcto funcionamiento de la aplicación y, de ser necesario, se solucionan errores, se mejoran detalles y se deja totalmente funcional y operativa para ser usada.

1.4 Estructura

En el *Capítulo 2. Museos y nuevas tecnologías* se hará una breve introducción histórica sobre las nuevas tecnologías aplicadas a los museos, desde sus inicios hasta la actualidad. Posteriormente, se definirá y detallará la realidad virtual y sus características, tipos y aplicaciones.

En el *Capítulo 3. Unity y Oculus* se hará una introducción general al entorno de desarrollo Unity y al software de realidad virtual Oculus, y se detallarán sus principales contenidos y capacidades para crear, importar y programar una escena virtual.

En el *Capítulo 4. La aplicación de realidad virtual* se describirá y detallará la aplicación de realidad virtual creada en este proyecto. Se empezará por su entorno, incluyendo la interfaz del editor con muchas de sus opciones y ventanas, y el modelado de todo el edificio y los objetos que lo componen. Seguidamente, se explicará todo el sistema de navegación a través de la aplicación, cómo se han configurado las gafas Oculus Rift y cómo se han integrado en el proyecto. Por último, se detallará el funcionamiento y las posibilidades que se ofrecen al usuario para interactuar con objetos del entorno y reproducir los contenidos multimedia de los que dispone la aplicación.

Finalmente, en el *Capítulo 5. Resultados* se expondrán los resultados obtenidos en este proyecto, el resultado de pruebas con usuarios y la posibilidad de ampliación en un futuro; y, para terminar, en el *Capítulo 6. Conclusiones y líneas futuras* se concluirá el proyecto determinando si se han alcanzado los objetivos propuestos.

Como anexo, se adjunta un pequeño tutorial y ejemplo de uso.

Capítulo 2. Museos y nuevas tecnologías

En este capítulo se hará una breve introducción histórica sobre las nuevas tecnologías aplicadas a los museos, desde sus inicios hasta la actualidad. Posteriormente, se definirá y detallará la realidad virtual y sus características, tipos y aplicaciones.

2.1 Repaso histórico

En 1967, una agrupación informal de museos de Nueva York creó la Museum Computer Network (MCN) con la finalidad de automatizar sus registros de visitas. Este hecho podría describirse como el inicio de la inclusión tecnológica en el ámbito de la exposición del arte en museos.

Tiempo después, el crecimiento de Internet y posteriormente la aparición de los Smartphone hicieron que, en la actualidad, más de las tres cuartas partes de los museos del mundo tengan como mínimo página web y presencia en redes sociales, y ya bien entrados en el siglo XXI se empiece a estandarizar el uso de aplicaciones móviles.

A día de hoy, ya son muchos los recursos tecnológicos usados en los museos. Desde módulos y paneles interactivos fijos con digitalización de imágenes, vídeos o audios explicativos, pasando por las audio-guías para escuchar durante una visita, hasta algo de lo más reciente: las aplicaciones móviles de realidad aumentada.

Este tipo de realidad hace posible la interacción del usuario con su entorno a través de la pantalla de su Smartphone o Tablet. Es un mundo digital superpuesto a la realidad, a lo que el usuario ve. Para ello, es necesario descargar una app y luego escanear con la cámara algún tipo de identificador situado en cualquier objeto expuesto, como por ejemplo un código QR, el cual "lanzará" el evento con el que interactuar. Dicho evento puede ser un audio explicativo, un vídeo ejemplificativo, un objeto en 3D o cualquier tipo de interacción que se pudiera realizar a través de la pantalla del dispositivo móvil. Con este y con todos los recursos tecnológicos mencionados es posible añadir a la clásica exposición visual una experiencia más personal e independiente destinada a cada usuario con multitud de interacciones y hacer aflorar nuevas sensaciones y estímulos para los sentidos.

Y es que la inclusión de las Nuevas Tecnologías de la Información y Comunicación (NTICs) en los espacios de exposición del arte, historia y cultura antigua (y moderna) ha logrado unir épocas muy distantes en el tiempo con el fin de modernizar y mejorar la calidad de las visitas a los museos.

Ahora bien, la gran apuesta tecnológica llamada a cambiar la forma concebida en la sociedad actual de la visualización y representación del arte tradicional no es otra que la realidad virtual.

2.2 Realidad virtual

La realidad virtual (RV) es un entorno 3D generado completamente de forma artificial con el objetivo de crear en el usuario la sensación de estar inmerso en él. Pretende ser un mundo ficticio que parezca real.

2.2.1 Características y tipos

El aspecto más importante de la realidad virtual es la inmersión. El usuario debe notar su presencia en el escenario en cuestión, es decir, sentirse "dentro de él", que "está allí" y así evadirse de la realidad. Para ello se usan estímulos visuales, auditivos e incluso táctiles para poder interactuar con objetos virtuales del entorno. Otro aspecto a destacar es la capacidad de "transportar" al usuario al lugar escogido. Se podría decir que es una forma de encontrarse en lugares recónditos y de "viajar" a cualquier parte del mundo o a cualquier época sin moverse del salón de casa.

Existen dos tipos de realidad virtual: *inmersiva* y *no inmersiva*. La diferencia principal es que los métodos *inmersivos* se aprovechan del uso de cascos, gafas, guantes u otros dispositivos para manipular el entorno generado por un ordenador, los cuales añaden una sensación mucho mayor de inmersión. Por el contrario, la realidad virtual *no inmersiva* únicamente utiliza teclado y ratón en ordenadores o incluso nada si se trata de una pantalla táctil. Esto hace la experiencia más accesible para cualquier usuario al requerir menor coste económico y computacional, sacrificando realismo.

Como ejemplos, tanto de RV inmersiva como no inmersiva, es posible encontrar simuladores de conducción de coches o pilotaje de aviones, proyección de imágenes reales convertidas a 3D para navegación como Google Maps o representación en ordenadores para videojuegos. Son ejemplos que sirven para ambos tipos porque simplemente usando unas gafas 3D y en algunos casos un tipo de controlador diferente, es posible, siempre que el software esté preparado para ello, pasar de una experiencia no inmersiva a otra inmersiva al instante.

De aquí en adelante, ya que es la finalidad de este proyecto, se tratará únicamente la realidad virtual inmersiva en entornos virtuales con gafas 3D y sus accesorios.

Figura 2. Simulador de conducción.

Figura 3. Vista en 3D de Google Maps.

2.2.2 Hardware

El hardware utilizado son unas gafas 3D totalmente cerradas con pantalla incorporada las cuales hacen que la inmersión y la presencia en el entorno sea aún mayor que la conseguida en la pantalla de un ordenador o de un dispositivo móvil. Vienen acompañadas de dos mandos o controladores, uno para cada mano, con botones y joysticks. Además, estos incorporan sensores de detección de posición y movimiento que facilitan la interacción con objetos del entorno y hacen que los gestos sean muy naturales e intuitivos. Los modelos de gafas comerciales más utilizados son las Oculus Rift (Facebook), Samsung Gear VR, HTC Vive o PlayStation VR (Sony).

Figura 4. Head-mounted-display (HMD) Oculus Rift.

Figura 5. Mandos controladores Oculus Rift.

2.2.3 Historia

Históricamente, la realidad virtual tiene su origen en la Segunda Guerra Mundial. Un proyecto del MIT llamado Whirlwind finalizó en 1951 con la creación de un simulador de vuelo para el entrenamiento de pilotos de bombarderos de la Marina de Guerra de los Estados Unidos. Fue 8 años más tarde cuando las Fuerzas del Aire retomaron el proyecto y empezó así un uso civil de la tecnología 3D.

Años más tarde se crearon diversos sistemas de realidad virtual. En 1962, uno de los pioneros de la realidad virtual y cineasta Morton Heilig construyó una máquina llamada Sensorama que permitía visualizar imágenes estereoscópicas en 3D mientras se estimulaban el oído, el olfato y el tacto.

Figura 6. Sensorama.

En 1968, el científico estadounidense Ivan Sutherland creó lo que fue el primer intento de casco de realidad virtual, llamado “La Espada de Damocles” (The Sword of Damocles). Era un sistema compuesto por un brazo mecánico articulado, fijado al techo y unido a la cabeza del usuario mediante una especie de casco binocular. Un ordenador generaba imágenes en una pantalla estereoscópica usando wireframe, un algoritmo de renderización basado únicamente en el trazado de líneas de las aristas del mallado de objetos. Además, gracias a los sensores del brazo articulado el sistema era capaz de detectar el movimiento de la cabeza del usuario y así ir cambiando el punto de vista dentro del entorno. Fue una gran invención, muy moderna para la época, pero la falta de tecnología en aquellos tiempos hizo que no se continuara avanzando en su desarrollo.

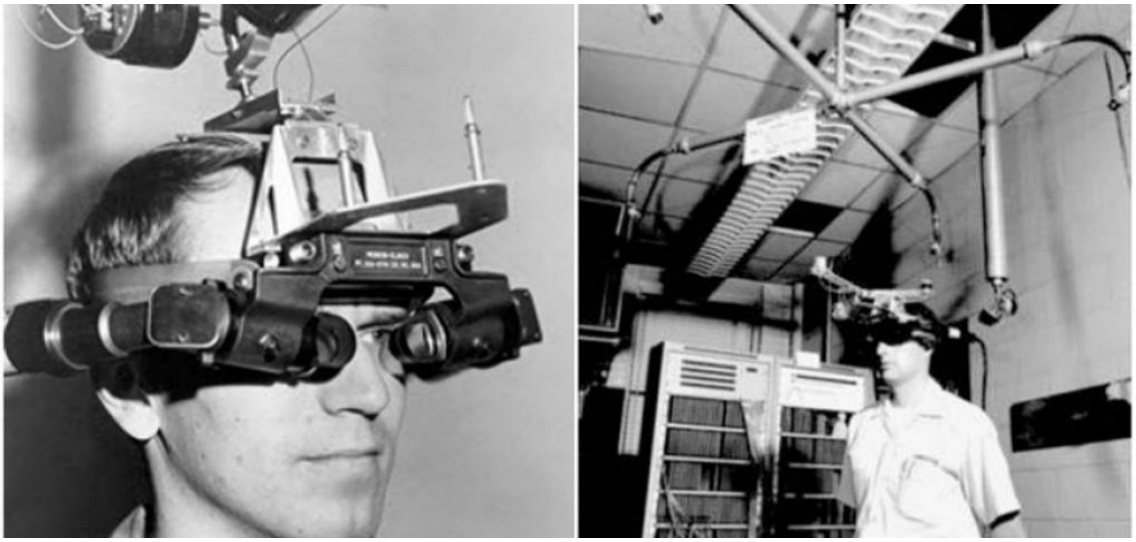

Figura 7. Espada de Damocles.

En 1978, un equipo del MIT realizó el Aspen Movie Map, el primer programa que permitía recorrer las calles de la ciudad de Aspen (Colorado, EEUU) e interactuar con ciertos edificios.

Fue ya iniciado el siglo XXI, en 2001, cuando apareció Google Earth, un programa informático compuesto por imágenes tomadas por satélites, fotografías aéreas y modelos creados por ordenador para crear una representación en 2D o 3D de todo el globo terráqueo. La aplicación permite la navegación por todo el planeta y acceder a cualquier lugar del mundo, dando la sensación de estar allí.

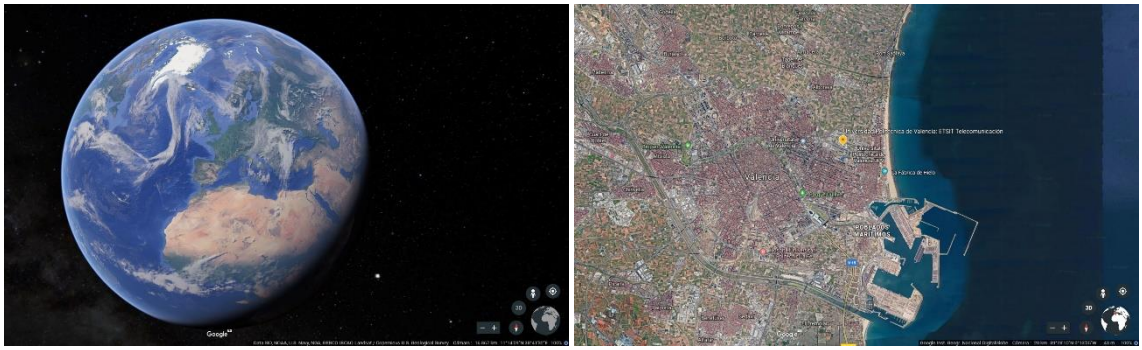

Figura 8. Vistas de la Tierra y València en Google Earth.

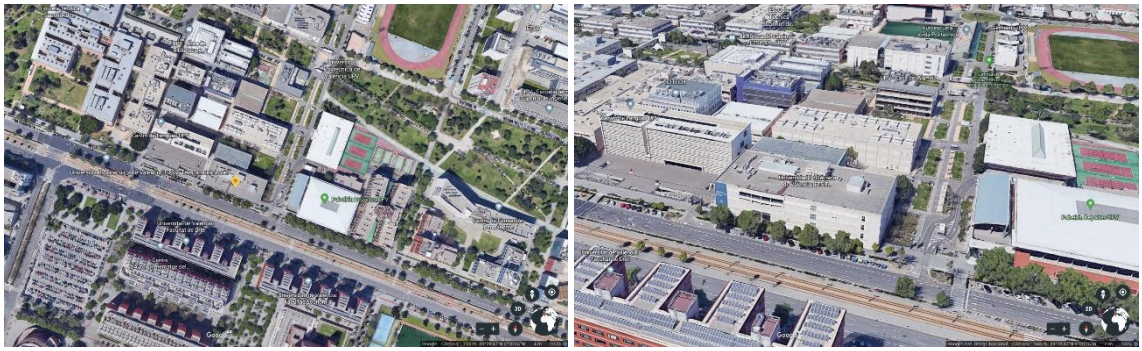

Figura 9. Vistas en 3D de la ETSIT.

A raíz de esto, la misma empresa desarrolló en 2005 Google Maps, una aplicación basada en la ya mencionada pero mucho más práctica y accesible, pasando a usarse más con fines utilitarios para información o navegación y dejando a Google Earth un poco más enfocada al entretenimiento. Google Maps ofrece imágenes 2D o 3D de mapas desplazables, zoom y sistemas de búsqueda por ciudad, dirección, etc. Además, permite cambiar entre tipo de vista Mapa o Satélite.

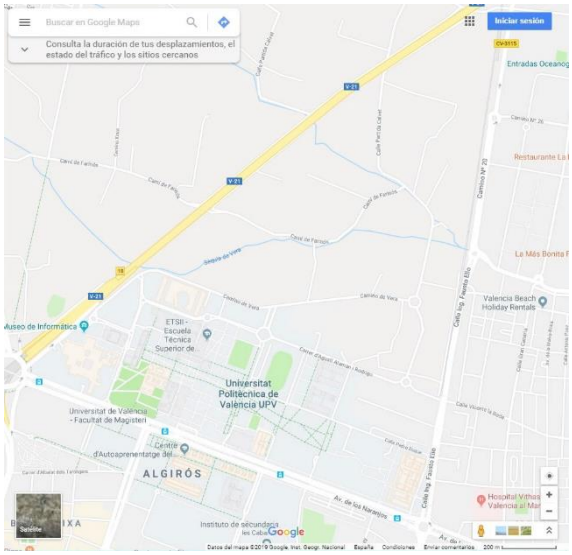

Figura 10. Vista mapa en Google Maps.

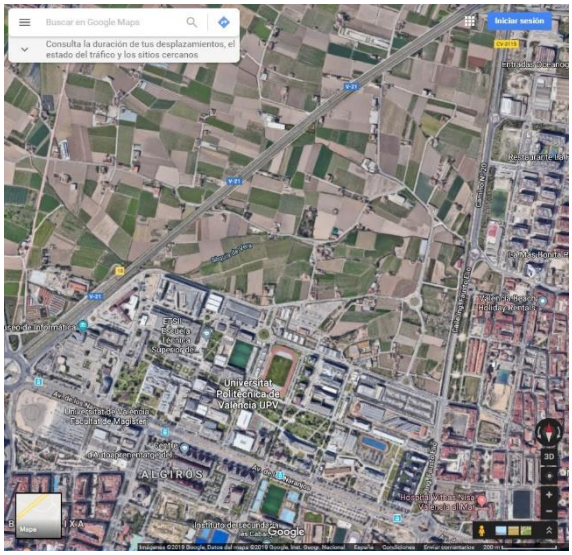

Figura 11. Vista satélite en Google Maps.

Dos años más tarde se incluyó Google Street View, una prestación que proporciona panorámicas de 360º en alta resolución a pie de calle simulando al usuario ser un vehículo o viandante.

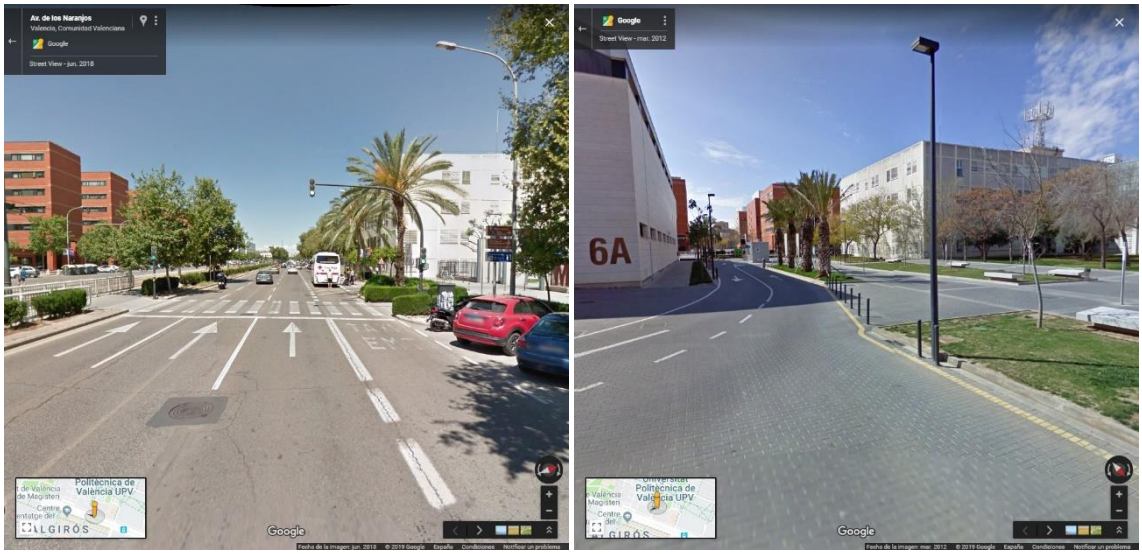

Figura 12. Vista Google Street View.

Finalmente, el año 2011 nace Google Art Project (ahora Google Arts & Culture, https://artsandculture.google.com/), un proyecto con sitio web y app para dispositivos móviles integrado por una recopilación de imágenes de obras de arte y en el que poder explorar de forma virtual más de, actualmente, 1700 museos en 70 países de todo el mundo. Ofrece galerías de imágenes, reportajes, colecciones detalladas y hasta visitas virtuales que utilizan la misma tecnología 3D que Google Street View, permitiendo desplazarse en primera persona por el interior de los museos y consultar toda la información que ofrecen. Además, no se limita a museos, sino que incluye monumentos antiguos, centros históricos de ciudades y edificios de interés entre otros.

Esta herramienta es, a día de hoy, uno de los mayores proyectos de integración de la realidad virtual y las nuevas tecnologías en los museos.

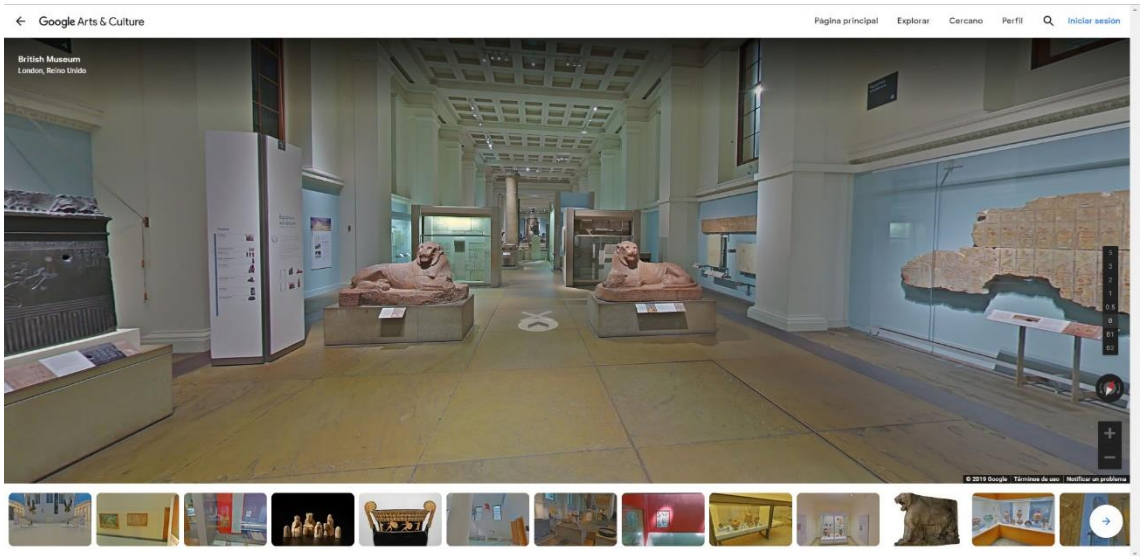

Figura 13. Vista de la visita virtual al British Museum en Londres.

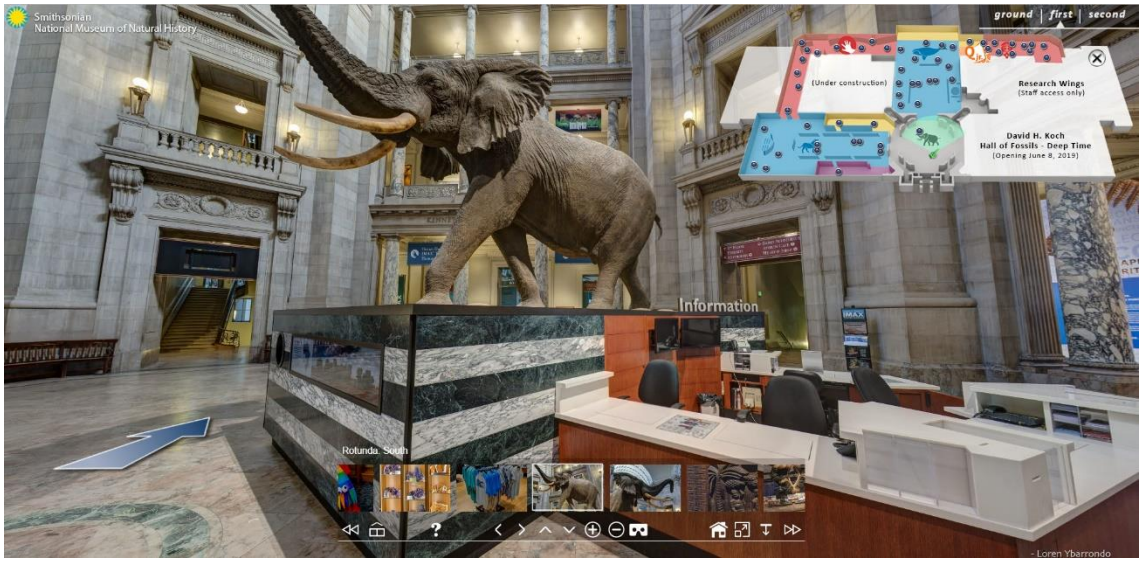

Figura 14. Vista de la visita virtual al National Museum of Natural History en Nueva York.

2.2.4 Aplicaciones

Existen muchos ámbitos de aplicación de la realidad virtual, entre los que se destacan los siguientes.

2.2.4.1 Videojuegos

El sector más importante y más creciente. Cada vez se desarrollan videojuegos más completos y más reales gracias al avance de la tecnología y a la mayor potencia y prestaciones de las consolas, ordenadores y Smartphone actuales. El objetivo suele ser, en la mayoría de los casos, el de crear un videojuego de máxima calidad técnica y gráfica para acercarse a una experiencia lo más realista posible. Aquí es donde entra en juego la realidad virtual para adaptar estos videojuegos a sistemas

en 3D y ofrecer una experiencia total permitiendo al usuario "traspasar" las pantallas. La mayoría de desarrolladores de nuevos videojuegos multiplataforma de hoy en día ya implementan también versiones para gafas 3D y se espera que la realidad virtual en los videojuegos siga creciendo en los próximos años.

2.2.4.2 Sanidad

Se prevé que, en los próximos años, el sector de la sanidad sea el que más crezca después de los videojuegos. Una de las áreas más importantes dentro de este sector es la realidad virtual aplicada a pacientes:

- Tratamiento de salud mental en el ámbito de la psiquiatría para tratar fobias. Más efectivo que la imaginación, mayor aceptación por parte de los pacientes frente a terapias de exposición y mayor control del entorno por parte del facultativo.

Figura 15. Tratamiento en RV contra fobia.

- Diagnóstico de enfermedades mediante inmersión en entornos virtuales y así poder llevar a cabo cualquier tipo de prueba en situaciones determinadas. También para valoración de lesiones y su posterior rehabilitación.

- Rehabilitación de pacientes que puedan haber sufrido, por ejemplo, un ictus. La realidad virtual puede ayudar con un entrenamiento cognitivo y motor que permita a los pacientes exponerse a situaciones reales que tendrán que afrontar en su día a día fuera del hospital.

Figura 16. Rehabilitación con RV de una paciente.

Y, por último, no olvidar la realidad virtual aplicada al profesional sanitario, mayormente para ayudar a mejorar a través de simuladores virtuales donde puedan practicar, por ejemplo, cirugías complejas sin poner en riesgo al paciente.

2.2.4.3 Educación

En este sector, la realidad virtual añade a la educación una moderna técnica para complementar el aprendizaje. Ofrece posibilidades de viajar sin salir de clase, tanto en espacio para explorar cualquier parte del mundo como en tiempo para viajar a cualquier época del pasado y vivir en primera persona muchos de los acontecimientos más importantes de la historia. Puede ayudar a empatizar con otras culturas y razas, por ejemplo, viajando a países subdesarrollados pudiendo ver la realidad del mundo y haciéndolo más cercano. O también puede servir de ayuda para orientar a los alumnos en su elección de estudiar una carrera profesional.

Capítulo 3. Unity y Oculus

En este capítulo se hará una introducción general al entorno de desarrollo Unity y al software de realidad virtual Oculus, y se detallarán sus principales contenidos y capacidades para crear, importar y programar una escena virtual.

3.1 Descripción

Para la creación de una aplicación de realidad virtual se necesita un entorno de desarrollo en 3D. De los principales y más conocidos entornos como son Unreal Engine, AppGameKit, CryEngine o Unity se ha elegido este último por su compatibilidad con los sistemas operativos iOS, Android, Windows, Mac OS, Linux, etc. y por soportar todas las más importantes plataformas de realidad virtual (Oculus Rift, PlayStation VR, Gear VR, etc.). Además, es el entorno más usado y accesible, y dispone de infinidad de información, manuales, librerías y todo tipo de soporte por Internet.

Unity es un motor de videojuegos creado por Unity Technologies disponible como entorno de desarrollo para Windows, Mac OS y Linux. Dispone de una licencia gratuita para uso personal o docente como proyectos universitarios, lo cual es ideal para este caso. Utiliza el lenguaje de programación C# y soporta las siguientes APIs: Direct3D en Windows, OpenGL en Windows, Mac OS y Linux, además de otras para iOS y Android u otras plataformas de videojuegos.

Para la creación de una aplicación, Unity hace uso de assets. Un asset es la representación de cualquier objeto que puede ser usado dentro del proyecto. Puede venir, por ejemplo, de un modelo 3D creado fuera de Unity, de un archivo de audio, de vídeo, una imagen o cualquier tipo de archivo soportado. Para todo ello, el entorno cuenta con una tienda (Asset Store) integrada en el editor desde donde poder buscar, comprar, descargar e importar al proyecto cualquier asset existente.

Una vez confeccionada en el editor lo que será la aplicación de realidad virtual con todo el modelaje, efectos, iluminación, scripts, etc. es necesario situar una cámara que hará de ojos, importar los modelos de las gafas, manos o mandos que se vayan a utilizar, sus texturas y scripts de comportamiento e interacción, etc., todo ello incluido en el kit de desarrollo de software (SDK) que Oculus proporciona para integrar sus modelos de gafas y periféricos en Unity.

Oculus VR es una compañía americana de tecnología de realidad virtual fundada en 2012 y comprada por Facebook en 2014. Su principal y más conocido producto es el modelo de gafas Oculus Rift, además de otros como las Oculus Rift S, Oculus Go o Gear VR en colaboración con Samsung. También dispone de hardware complementario como los sensores de seguimiento o los controles táctiles basados en el mando de una consola Xbox.

Para este proyecto se ha optado por Oculus y en concreto por el modelo de gafas Oculus Rift como dispositivo de realidad virtual por ser el pionero y más usado en el sector, además de, al igual que Unity, disponer de una enorme cantidad de información, manuales y recursos tanto en su plataforma web de desarrollo (https://developer.oculus.com/rift/) como por todo Internet.

3.2 Capacidades

3.2.1 Escenas y objetos

La principal funcionalidad de Unity es la de la creación de escenas animadas para poder componer un juego. Cada escena contiene sus propios menús, sus objetos, su distribución y sus comportamientos, y puede ser guardada o cargada en cualquier momento. El objeto más importante dentro de una escena es un GameObject. Cada objeto contenido en una escena es un GameObject, ya sea un avatar, una cámara, un objeto cualquiera dentro de una habitación, las luces, los efectos especiales… Todo. Y siempre estático. Ningún GameObject hace nada por sí sólo. Es necesario darle propiedades como texturas, efectos de audio, comportamiento frente a luces, reflejos y sombras, o adjuntándole un script (archivo de texto con código de programación) con determinada función programada previamente.

Un recurso interesante es el de crear, configurar y guardar un GameObject como un modelo predefinido (Prefab), manteniendo todas sus cualidades citadas anteriormente y pudiendo utilizarse en futuras ocasiones incluso en otro proyecto ajeno. De igual manera que es posible descargar Prefabs existentes de otros usuarios a través de la Asset Store y utilizarlos en el proyecto propio.

3.2.2 Importación

Una cualidad muy práctica y una de las principales razones por las que se ha elegido Unity en este proyecto (más adelante se explicará con detenimiento) es la de poder importar modelos de objetos creados con otros programas de edición 3D externos. Unity cuenta con su propia herramienta de edición y modelado, pero muy básica ya que no es su principal cometido. Puede que se prefiera trabajar con otro software más dedicado, por gusto o simplemente por otras razones cualesquiera. Así pues, es posible importar objetos con su tamaño, forma, color y textura predefinidos y usarlos directamente como un GameObject.

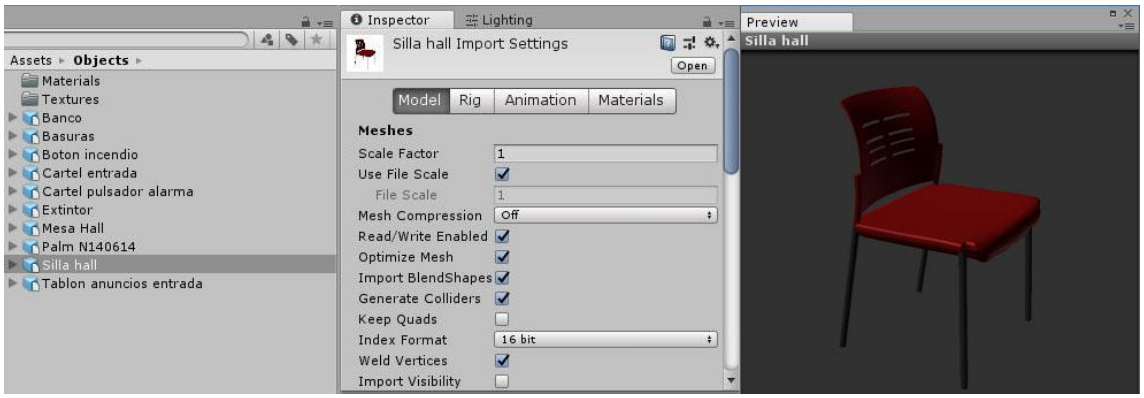

Figura 17. Ventanas en Unity del GameObject importado "Silla hall" con características y vista previa.

3.2.3 Programación

Otro de los aspectos importantes de Unity es la creación y edición de scripts. El lenguaje C# y todo el sistema de librerías que utiliza permite hacer prácticamente cualquier cosa que pudiera hacer el editor de forma visual con ratón y teclado en cuanto a creación y edición de modelos. Además, es la mejor forma de programar comportamientos de objetos dentro de la escena, ya sea al inicio de la misma o al producirse cualquier circunstancia provocada o no por la interacción del usuario. Por ejemplo, al tocar con el dedo índice un botón "Play" concreto se reproduce un sonido concreto. O el poder coger un objeto dentro de la escena e interactuar con él, moverlo, lanzarlo, modificar su tamaño, etc. Todo comportamiento requiere código de programación en un script asociado al objeto en cuestión.

3.2.4 Oculus Touch Controllers

Asimismo, toda interacción del usuario con la escena requiere el uso de los controles de las Oculus Rift. Con estos controles es posible tener un comportamiento intuitivo e idéntico al de las manos humanas. Mediante los botones de los dedos índice y corazón y de los sensores de seguimiento, es posible leer cada gesto de las manos como puede ser extender el dedo índice o cerrar el puño, e incluso medir la velocidad de movimiento a la hora de golpear o lanzar un objeto. Los controles son como una extensión virtual de las manos del usuario dentro de la escena.

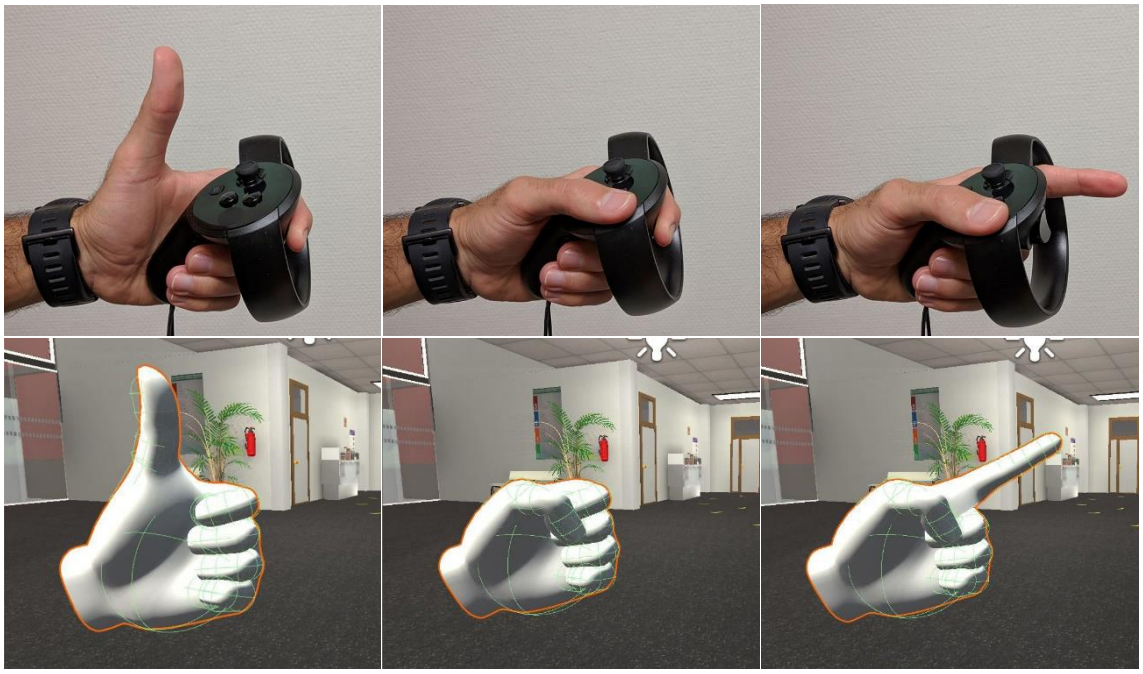

Figura 18. Comparativa de gestos en la realidad y en el mundo virtual.

3.2.5 Multiplataforma

Por último, Unity dispone de una gran facilidad para, una vez terminada la escena o escenas, construir la aplicación final para cualquier plataforma compatible con el tipo de proyecto. Por ejemplo, es posible construirla para Oculus Rift en Windows y muy fácilmente desactivar el uso de las gafas, cambiar cuatro opciones y modificarla para poder usarla en un Smartphone Android o iOS.

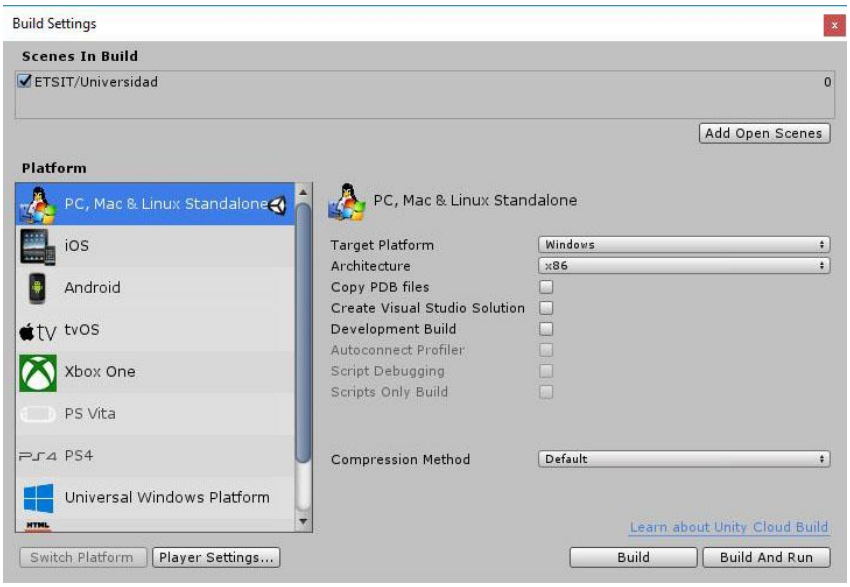

Figura 19. Ventana de selección de escenas y plataforma para la construcción del proyecto.

Capítulo 4. La aplicación de realidad virtual

En este capítulo se describirá y detallará la aplicación de realidad virtual. Se empezará por su entorno, incluyendo la interfaz del editor con muchas de sus opciones y ventanas, y el modelado de todo el edificio y los objetos que lo componen. Seguidamente, se explicará todo el sistema de navegación a través de la aplicación, cómo se han configurado las gafas Oculus Rift y cómo se han integrado en el proyecto. Por último, se detallará el funcionamiento y las posibilidades que se ofrecen al usuario para interactuar con objetos del entorno y reproducir los contenidos multimedia de los que dispone la aplicación.

4.1 Descripción

En esta aplicación se pretende desarrollar una versión virtual del Museo de la Telecomunicación Vicente Miralles Segarra (http://museotelecomvlc.etsit.upv.es/), situado en el edificio 4D de la Escuela Técnica Superior de Ingeniería de la Telecomunicación (ETSIT) en la Universitat Politècnica de València. En dicho museo se exponen alrededor de doscientas cincuenta piezas repartidas en 5 áreas temáticas: telefonía, telegrafía, equipos audiovisuales, instrumentos para medir e investigar y radiocomunicación. Todas las piezas son objetos o equipos antiguos como teléfonos y centralitas, radios, TV, fonógrafos, telégrafos, multímetros, radar… todos ellos pertenecientes a la historia de la telecomunicación.

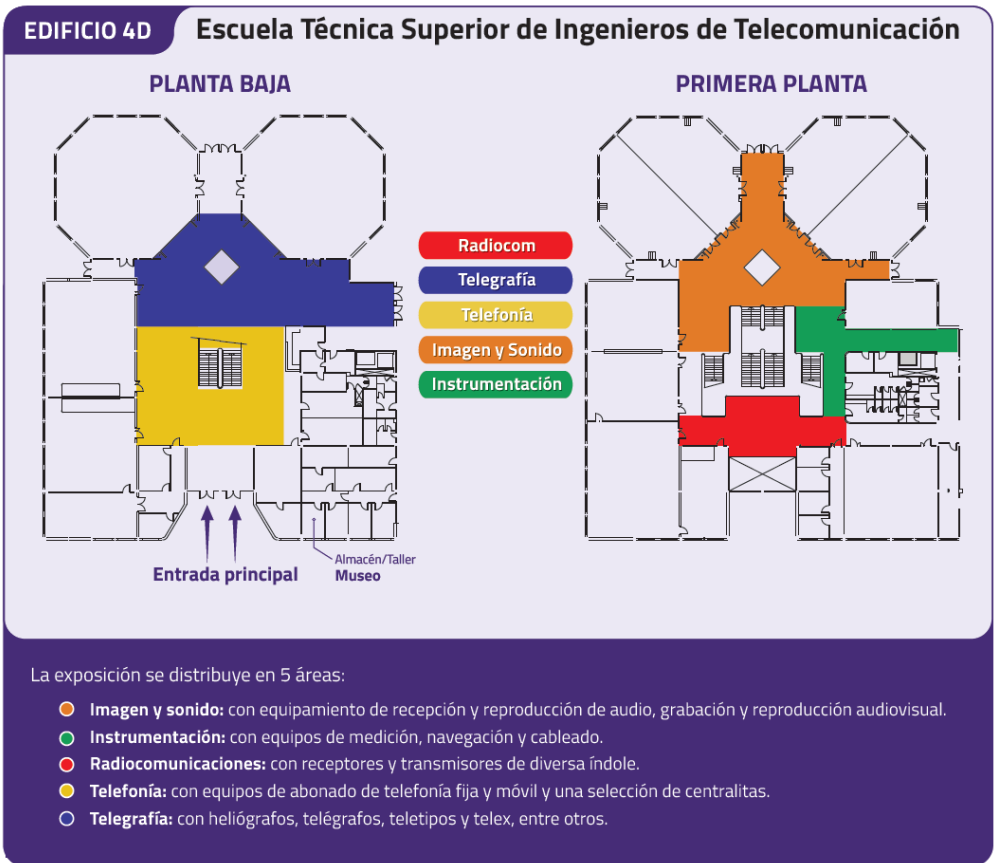

Figura 20. Plano de la ETSIT y áreas temáticas del museo.

La finalidad de la aplicación es ofrecer una experiencia innovadora para complementar la visita física al museo o sustituirla sin necesidad de estar presente. Además, añade contenidos multimedia únicamente disponibles también a través de la experiencia de realidad aumentada que ofrece el museo. Es una aplicación disponible para descargar desde un ordenador y usarla por medio de unas gafas de realidad virtual a través de la plataforma ofrecida por éstas.

Una vez iniciada, sitúa al usuario en la entrada del edificio de la ETSIT donde está ubicado el museo y deja libertad de movimiento para iniciar la visita por él. Dispone de las mismas vitrinas con objetos expuestos que ofrece el museo en la realidad con sus correspondientes carteles descriptivos. Además, añade contenido multimedia extra en vitrinas seleccionadas reproducibles mediante la interacción del usuario. Todo está diseñado para conseguir una copia los más exacta posible del museo y llevarlo al mundo virtual.

Figura 21. Comparativa del primer piso de la ETSIT en el mundo virtual y en la realidad.

4.2 Entorno

El desarrollo de la aplicación parte de un modelo en 3D ya creado de todo el edificio de la ETSIT. Este modelo es fruto del Trabajo de Fin de Grado "Visita virtual al Museo de las Telecomunicaciones Vicente Miralles" del exalumno de la escuela Luis Pérez Pau realizado en el año 2015. En dicho TFG, se creó una aplicación para representar de forma virtual el museo y poder visitarlo desde un ordenador con teclado y ratón.

El modelo del edificio se creó con un software de edición 3D llamado Blender para luego importarlo a Unity. Está formado por toda la estructura, fachada y todo su interior, habiendo diversos objetos como bancos, mesas, sillas, plantas, papeleras, etc. Todo lo que formaba parte de la planta baja y del primer piso, además de un modelo de vitrina, para ajustarse a la disposición del interior del edificio en el año 2015.

Figura 22. Modelo importado del edificio de la ETSIT en el año 2015.

Ahora, en esta nueva versión, se parte del modelo mencionado, se completa, se modifica para ajustarlo a la actualidad y se adapta para ser usado en las Oculus Rift.

4.2.1 Interfaz

La interfaz de Unity se compone de diversas ventanas, todas ellas modificables en cuanto a situación en la pantalla. Las más idóneas y que se han usado para este proyecto son las siguientes.

En la parte central, la ventana de escena (*Scene*) donde se pueden ver y editar de forma visual todos los objetos que la componen. A la izquierda una ventana con la jerarquía (*Hierarchy*) de objetos que componen dicha escena. En la parte derecha, arriba una pequeña ventana de juego (*Game*) con la vista de cámara, la cual muestra en todo momento lo que ve la cámara, bien en modo edición (cámara estática) o bien cuando se pone en marcha la escena (cámara movible acompañando al movimiento de las gafas 3D). Y justo debajo, una ventana de inspección (*Inspector*) con todos los componentes de cada objeto para poder modificarlos o añadir cualquier otro (posición, mallado, características físicas, material, textura, scripts…). Por último, en la parte inferior, la ventana del proyecto (*Project*) donde se muestran todos los directorios y ficheros que éste contiene.

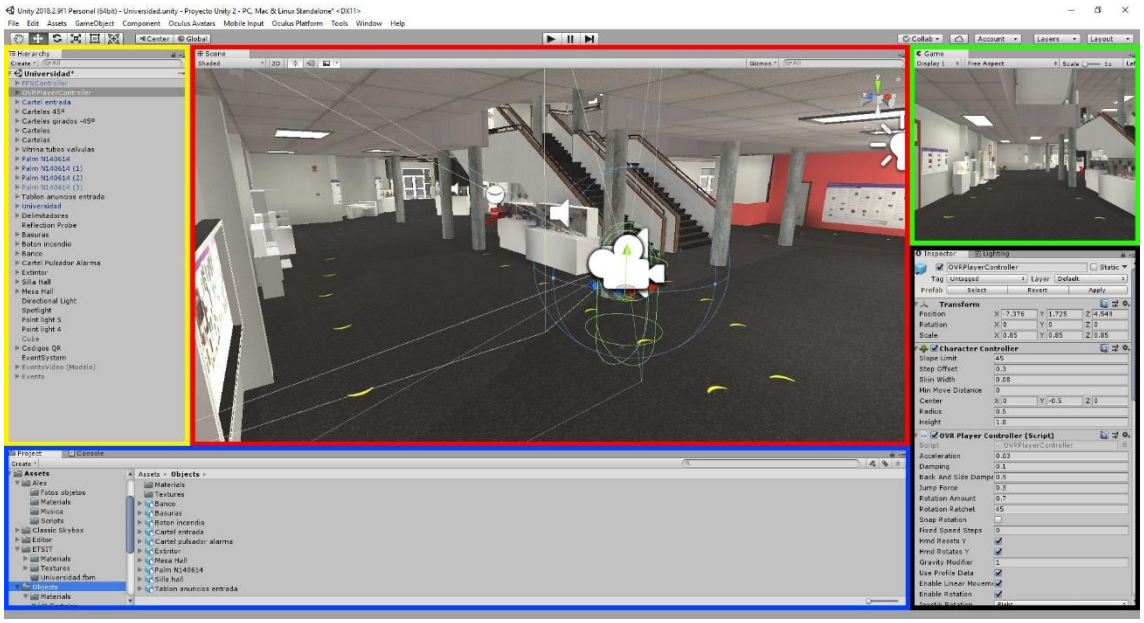

Figura 23. Interfaz general del entorno Unity.

4.2.2 Creación de objetos

Aunque Unity no es un software de edición y modelado de objetos 3D como tal, sí que permite la creación de figuras geométricas básicas como cubos, esferas, cilindros, planos y modificar su tamaño, además de añadirles texturas y efectos de iluminación. Por ello, todos los objetos necesarios que no se encontraban originalmente en el modelo del edificio importado como base, han podido ser creados en Unity sin necesidad de usar otro software externo.

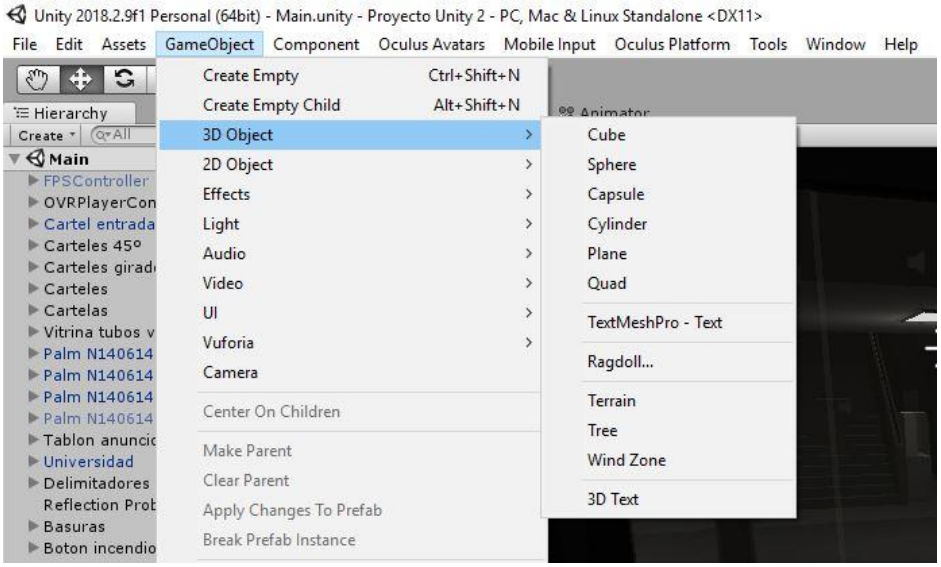

Figura 24. Menú de creación de objetos 3D y figuras geométricas básicas.

El procedimiento de creación es el siguiente. Se elige un tipo de objeto y se crea en la jerarquía o directamente en la ventana de edición de escena. Bien con el ratón de forma gráfica o bien en el primer componente del inspector, se modifica el tamaño, la posición y la escala del objeto. A continuación, es posible añadir otros componentes como un mallado o caja de contorno, hacer el objeto rígido para evitar ser atravesado, darle gravedad o añadirle un material y una textura mediante una imagen externa. Además, es posible adjuntarle un script con código programado para un determinado comportamiento como puede ser hacer que un objeto se pueda coger, que cambie de color al tocarlo o que suene un sonido determinado. Existen infinidad de acciones programables con scripts.

Para ejemplificar lo mencionado, se detalla la creación y colocación del cartel de uno de los objetos expuestos. Desde el menú GameObject > 3D Object > Cube se crea un cubo gris. Con el ratón sobre la ventana de escena, se alarga en forma de rectángulo, se aplana para darle la forma de un cartel y se arrastra para colocarlo en la pared.

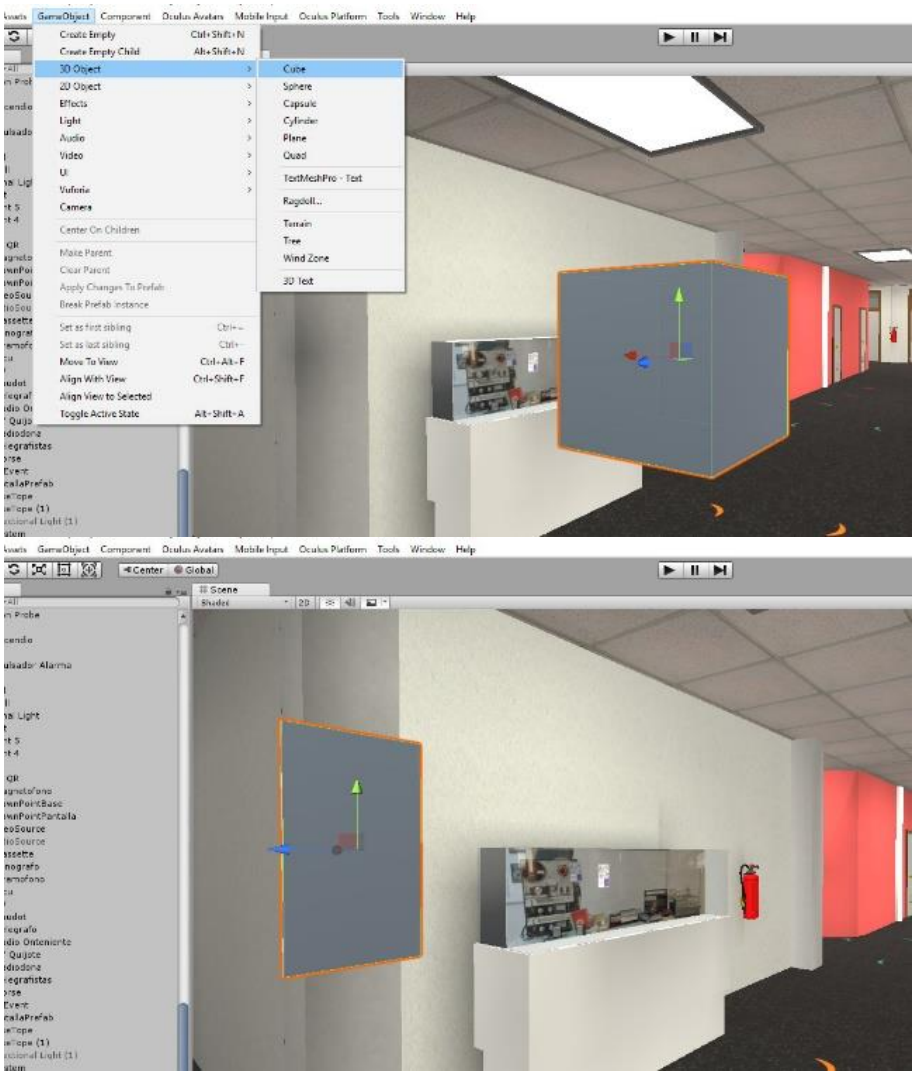

Figura 25. Creación y redimensionado de un cubo.

Una vez con las dimensiones y la forma deseadas, se selecciona el objeto y, desde la ventana de inspección, se añade el componente *Box Collider* seleccionándolo o buscándolo en la lista que aparece al dar click en el botón inferior *Add Component*. Esto dotará al objeto de más realismo al hacer que pueda "colisionar" con otros objetos del entorno y no pueda ser atravesado.

Seguidamente, desde la ventana inferior de proyecto, en el directorio deseado, con un click derecho se selecciona Create > Material para crear un nuevo material. Ahora, desde Windows, se arrastra a esa misma ventana una fotografía hecha al cartel real para importarla al proyecto. Se selecciona ésta última y se arrastra al recuadro correspondiente dentro del componente del material creado previamente.

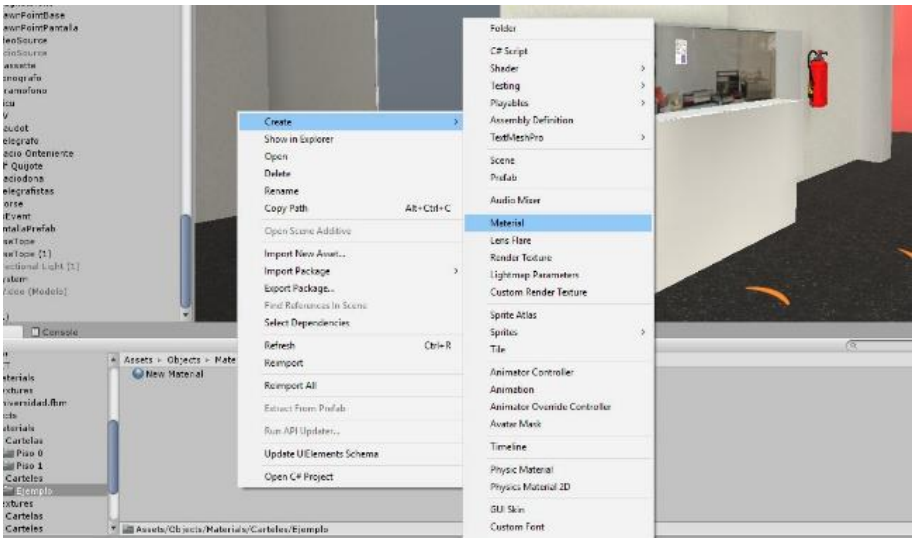

Figura 26. Creación de un nuevo material.

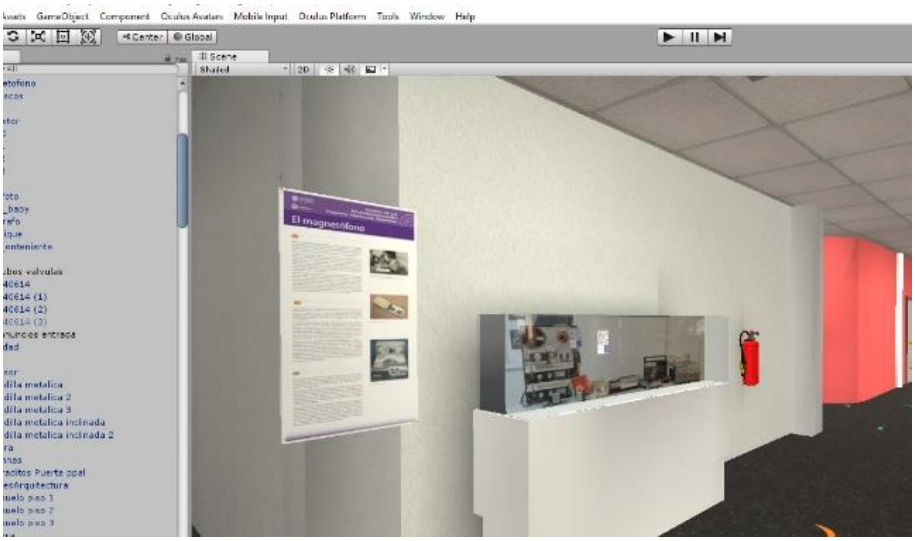

Figura 27. Textura del cartel y colocación en la pared.

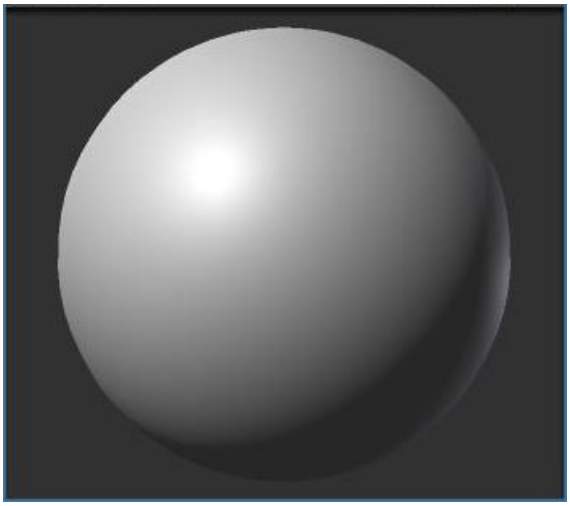

Figura 28. Material nuevo, sin textura.

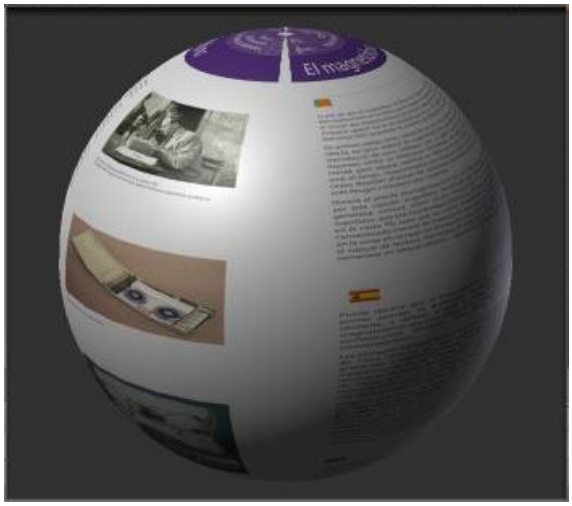

Figura 29. Material con textura de cartel.

Así se obtiene un material con la textura del cartel real y sólo queda arrastrarlo y soltarlo encima del cartel situado en la pared. Esta es la forma básica de crear un objeto, darle características y colocarlo en la escena. Ahora ya es posible duplicar objetos tantas veces como se precise.

4.2.3 Modelado y colocación de objetos

El primer paso del modelado es completar el modelo del edificio con todos los objetos que faltan de la forma que se ha detallado en el punto anterior.

4.2.3.1 Objetos ajenos al museo

Se empieza por recolocar, uno a uno, todos los objetos movibles que forman parte del edificio ajenos al museo. En este caso, no es necesario crear ni modificar ninguno ya que son los mismos objetos que había en el año 2015. Basta con moverlos para que coincidan con su disposición en la realidad. Son los bancos y sillas, carteles de información, papeleras, plantas, alarmas de incendio y extintores.

Figura 30. Objetos ajenos al museo.

4.2.3.2 Objetos del museo

A continuación, se colocan todos los objetos pertenecientes al museo, ya sean duplicados o creados (si todavía no existen), y se les añaden las características necesarias dependiendo de la función de cada uno, las cuales se detallarán en la sección de cada objeto.

4.2.3.2.1 Vitrinas

Se coloca una nueva vitrina con las dimensiones que le corresponda en cada lugar del edificio donde se sitúe una de ellas, tanto en la planta baja como en el primer piso.

El museo está formado por 40 vitrinas de diferentes tamaños y alturas (rectangulares, cuadradas, más o menos alargadas y más o menos altas), dependiendo del objeto u objetos de exposición que éstas contengan.

El modelo genérico de vitrina está formado por dos partes: el pie de vitrina, hecho con un material de plástico blanco opaco; y el cristal, hecho con un material transparente y con la característica de incidencia de luz difusa.

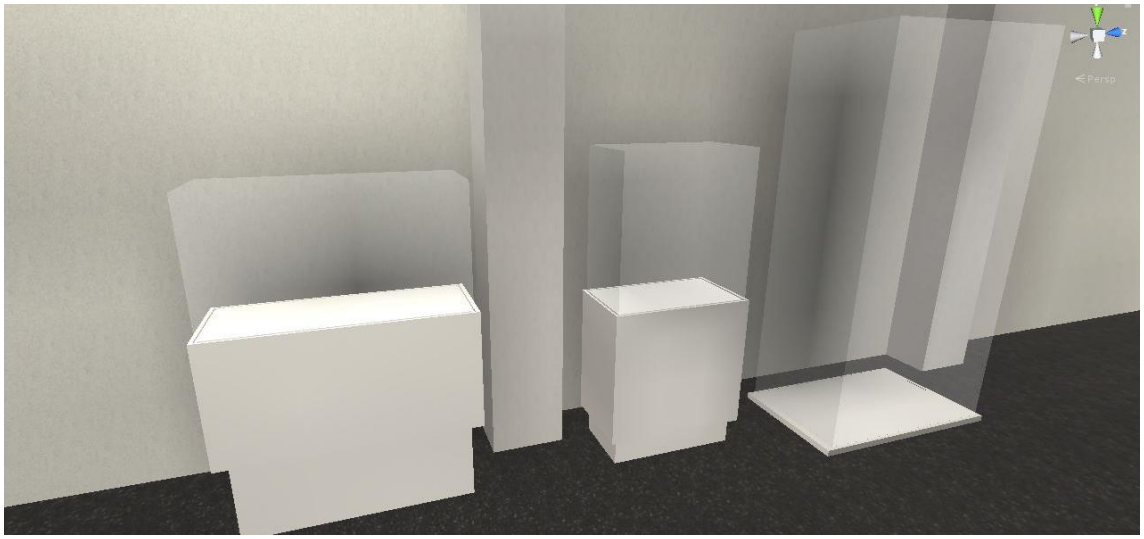

Figura 31. Diferentes modelos de vitrina.

4.2.3.2.2 Carteles

Se crean todos los carteles que describen los objetos expuestos y se colocan junto a sus vitrinas correspondientes.

Figura 32. Modelo de cartel.

Cada vitrina (o cada par de ellas) dispone de su cartel descriptivo, de aproximadamente 40x80 cm de ancho por largo, situado en la misma vitrina o a su alrededor, en la pared, en una columna o en el lugar idóneo para lograr una buena visibilidad. En estos carteles se detalla un modelo genérico del objeto u objetos expuestos en esa vitrina, junto con descripciones, información histórica, curiosidades, otros datos de interés e imágenes. Todo en tres idiomas: valenciano, castellano e inglés.

El modelo genérico de cartel es el ejemplificado en el punto *4.2.2 Creación de objetos*, formado por un cubo redimensionado y un material por defecto, el cual figurará como objeto (GameObject) “cartel”.

Ahora bien, la parte diferencial de estos modelos es su textura, la cual hace posible que se le pueda asignar a cada objeto “cartel” una imagen externa. Para ello, es necesario importar al proyecto (basta con arrastrar y soltar en la ventana inferior de proyecto) todas las imágenes en alta resolución de los carteles del museo. Y, seguidamente, se asigna, uno a uno, cada archivo de imagen a su objeto “cartel” correspondiente.

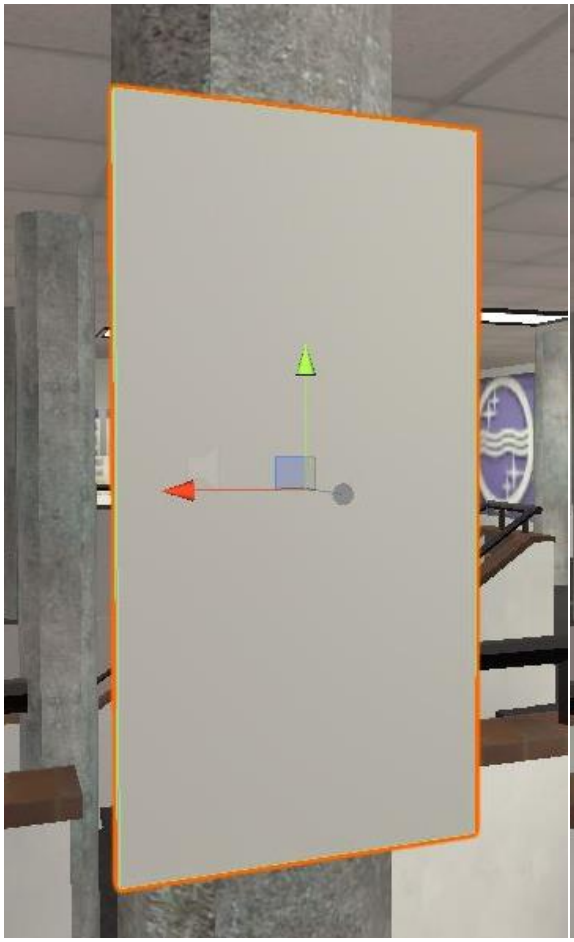

Figura 33. Modelo de cartel sin textura.

Figura 34. Modelo de cartel con textura.

Figura 35. Diferentes medidas de carteles: vertical y horizontal.

4.2.3.2.3 Objetos expuestos

Uno de los aspectos importantes de la aplicación es el modelado de los objetos expuestos en las vitrinas, que al fin y al cabo es la parte principal del museo y lo que se viene a ver. Estos modelos, está realizándolos un alumno de Bellas Artes en su Trabajo de Fin de Grado. Como todavía no están disponibles, se ha optado por colocar fotografías hechas a las vitrinas en lugar de los modelos 3D como medida provisional hasta que estos estén terminados.

Para ello, se han realizado fotografías con buena resolución a cada una de las vitrinas del museo. Se han hecho de frente y se han tenido en cuenta la perspectiva y la inclinación con las que observaría una persona humana de entre 160 y 170 cm de altura. De esta forma, se consigue un mayor realismo.

Una vez hechas todas las fotos, se editan, se recortan en forma rectangular y, de ser necesario, se retoca la iluminación para ajustarlas lo mejor posible a la situación de cada vitrina en el mundo virtual. Luego, se importan al proyecto y se crea un modelo básico en forma de cubo como el usado en los carteles. Se redimensiona, se duplica tantas veces como vitrinas haya y se añade una foto a cada modelo en forma de textura. Por último, se colocan dentro del cristal de cada vitrina y se reajustan para acoplarse al tamaño adecuado.

Figura 36. Comparativa de vitrina real y vitrina virtual. Teléfonos y centralitas de campaña.

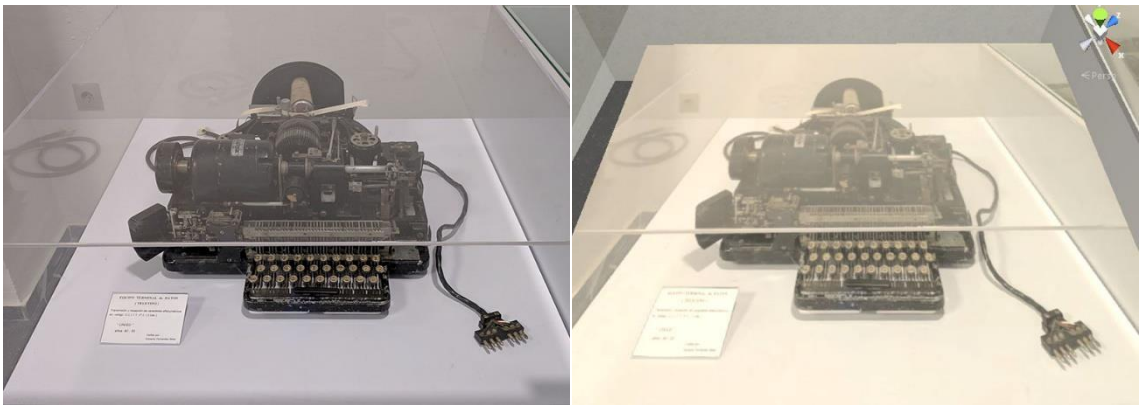

Figura 37. Comparativa de vitrina real y vitrina virtual. Teletipo Creed B7.

Figura 38. Comparativa de vitrina real y vitrina virtual. Fonógrafo y gramófono.

Figura 39. Comparativa de vitrina real y vitrina virtual. Pathé baby.

Figura 40. Comparativa de vitrina real y vitrina virtual. Tubos y válvulas de vacío.

4.2.3.2.4 Cartelas

Se crean todas las cartelas identificativas y se colocan a pie de vitrina debajo del objeto expuesto correspondiente.

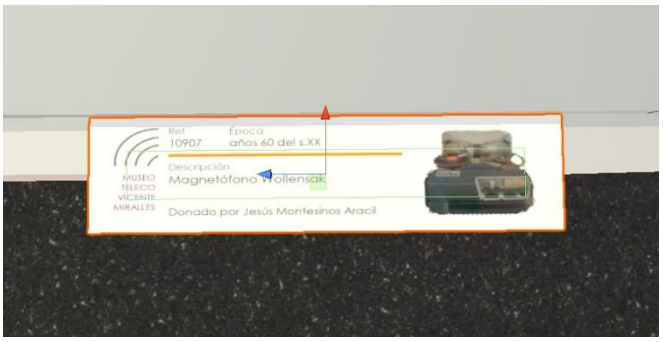

Figura 41. Modelo de cartela.

Cada objeto expuesto del museo lleva asociada una pequeña cartela rectangular de aproximadamente 11x4 cm de ancho por largo. En ella figura el nombre, número de referencia y época del objeto, además de una foto y el donante que lo ha proporcionado al museo. Esto es muy útil ya que, además de informar sobre el objeto que se está observando durante la visita, es posible usar esta información para buscar el objeto en cuestión en la base de datos de catalogación online de la que dispone el museo (http://colteleco.webs.upv.es/).

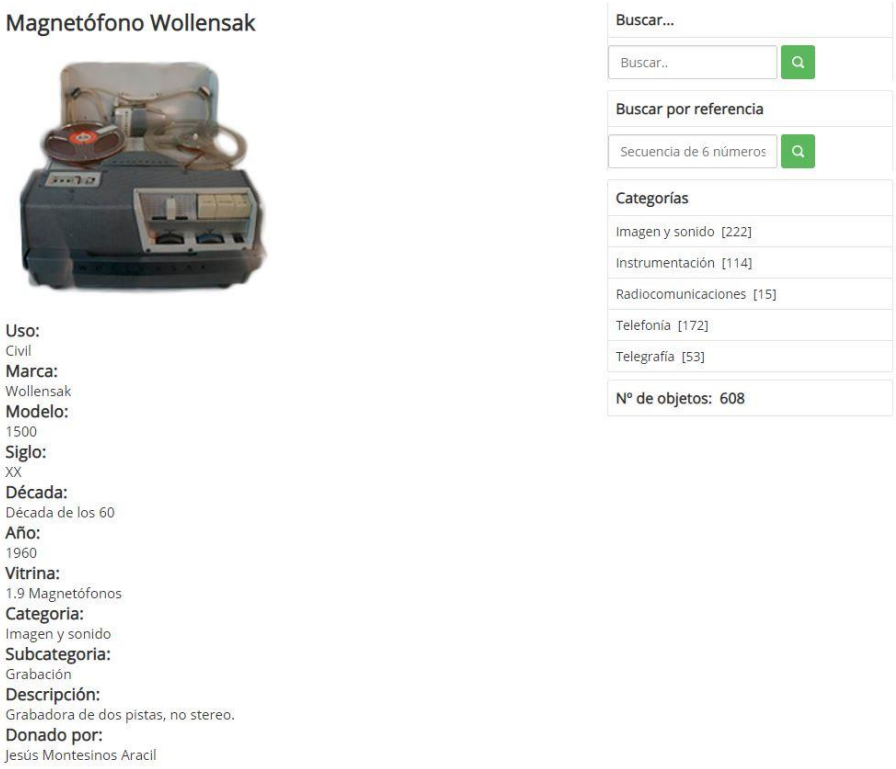

Figura 42. Información de un objeto de la catalogación online.

El modelo genérico de cartela es el mismo que el de un cartel, redimensionando su tamaño y medidas y asignándole a cada objeto "cartela" su imagen externa correspondiente en alta resolución, tal y como se ha descrito anteriormente.

Figura 43. Vitrina con cuatro cartelas.

4.2.3.2.5 Códigos QR

Se crean los diversos códigos QR que se usarán como activadores de los contenidos multimedia (se detallará en un punto más adelante) y se colocan sobre el cristal de las vitrinas correspondientes.

Figura 44. Modelo de pegatina con código QR.

El museo dispone, de momento, de diez archivos de vídeo y tres archivos de audio relacionados con trece de los objetos expuestos. Cinco de los vídeos muestran el funcionamiento de objetos como el fonógrafo, magnetófono, gramófono, tocadiscos *Picú* y un teléfono antiguo. Otro, muestra un clip en blanco y negro del programa *Un, dos, tres… responda otra vez* de RTVE como ejemplo de cómo era el televisor Philips expuesto de la década de los 60. Estos contenidos multimedia ayudan a entender el funcionamiento de objetos antiguos los cuales puede que muchas de las personas jóvenes que visiten el museo nunca hayan tenido la oportunidad de ver.

El modelo genérico de código QR es el mismo que el de un cartel o una cartela. Esta vez, con todos los tamaños y texturas iguales.

4.2.3.2.6 Marcas de itinerario

Se crean las marcas de colores que sirven de itinerario en la visita al museo y se colocan en el suelo de ambas plantas siguiendo la ruta establecida.

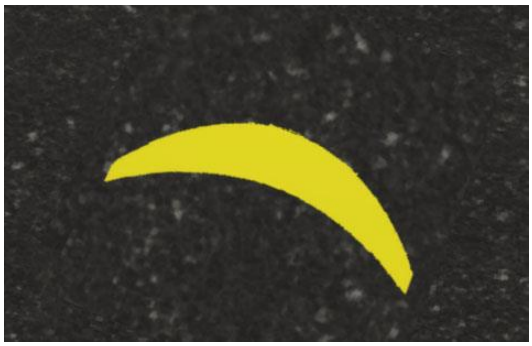

Figura 45. Modelo de marca de itinerario.

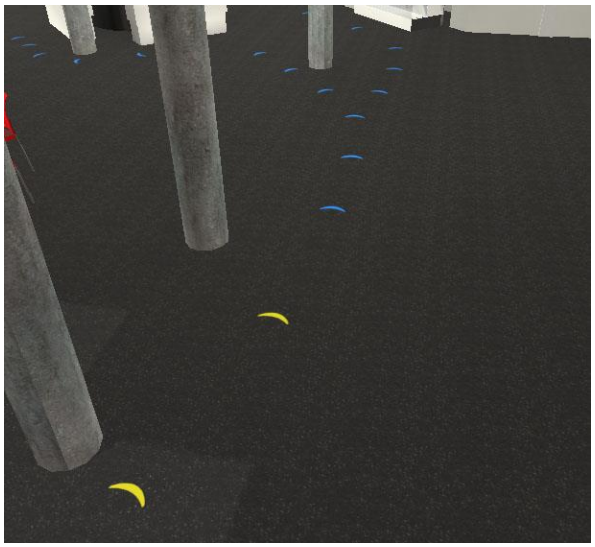

Figura 46. Itinerario de la visita al museo.

Desde la entrada principal del edificio, empieza un itinerario marcado en el suelo que recorre todo el museo. Aunque es posible moverse libremente, es aconsejable seguir las marcas para hacer una visita óptima, ya que éstas están coloreadas acorde a cada una de las 5 áreas temáticas en las que está dividido el museo, como se puede apreciar en el plano de las dos plantas del edificio.

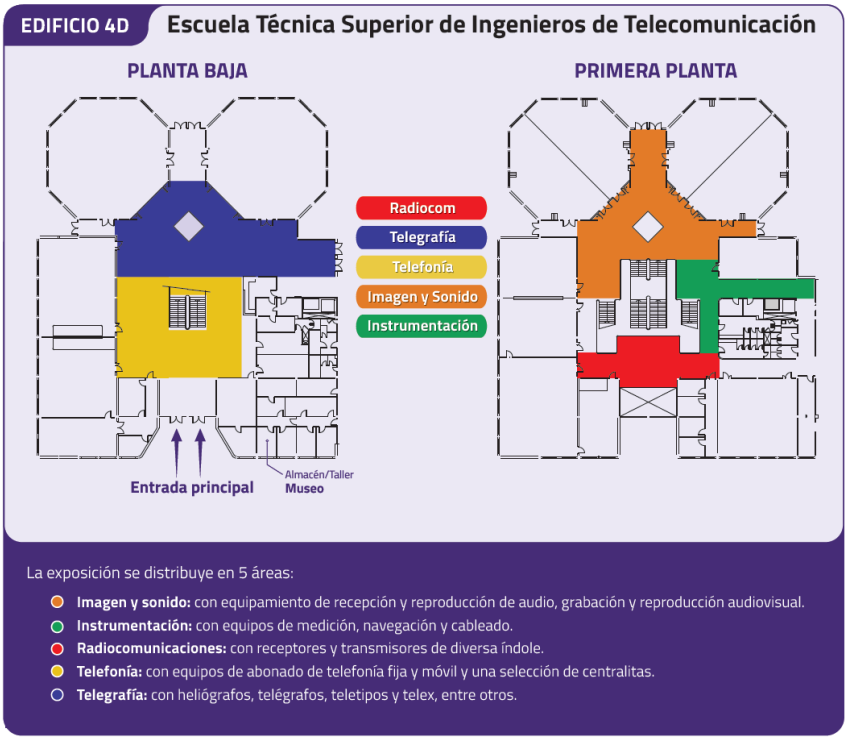

Figura 47. Plano de la ETSIT y zonas temáticas del museo.

4.2.3.3 Colisiones

Una de las características más importantes de todo el proyecto es la capacidad de los objetos de colisionar con otros objetos. Esto incluye la interacción del usuario con cualquier objeto del entorno, ya que las partes del cuerpo de éste son también consideradas objetos. La importancia de todo esto radica en la detección de estas colisiones y, en consecuencia, en poder asignarles determinado comportamiento.

Básicamente, el objetivo es establecerle a cada objeto una silueta o un contorno rígido. Para ello, se selecciona el objeto en cuestión y, desde la ventana de inspección (*Inspector*), se le añade un nuevo componente llamado *Collider*.

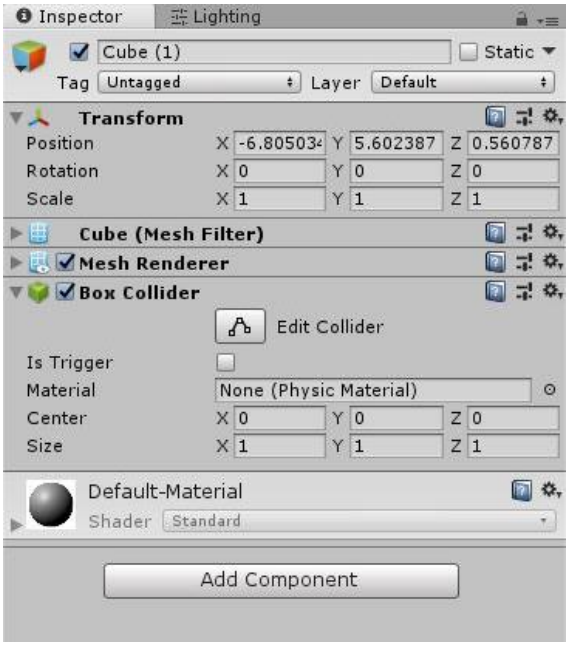

Figura 48. Ventana *Inspector* de un cubo cualquiera.

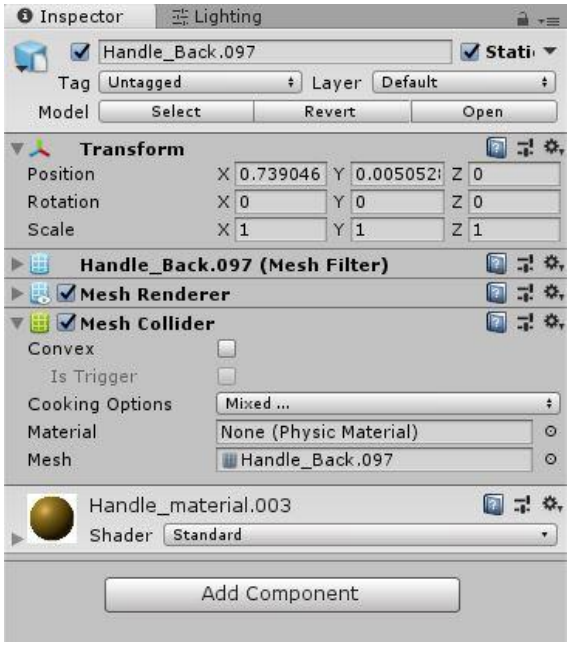

Figura 49. Venta *Inspector* de un pomo de una puerta.

Dependiendo de la forma geométrica del objeto, éste componente varía mínimamente. Para objetos rectangulares resulta más óptimo usar un *Box Collider*, como indica su nombre, en forma de caja. Existen más tipos como los *Capsule Collider* (para cápsulas) o *Sphere Collider* (para esferas), aunque el más usado es el *Mesh Collider*.

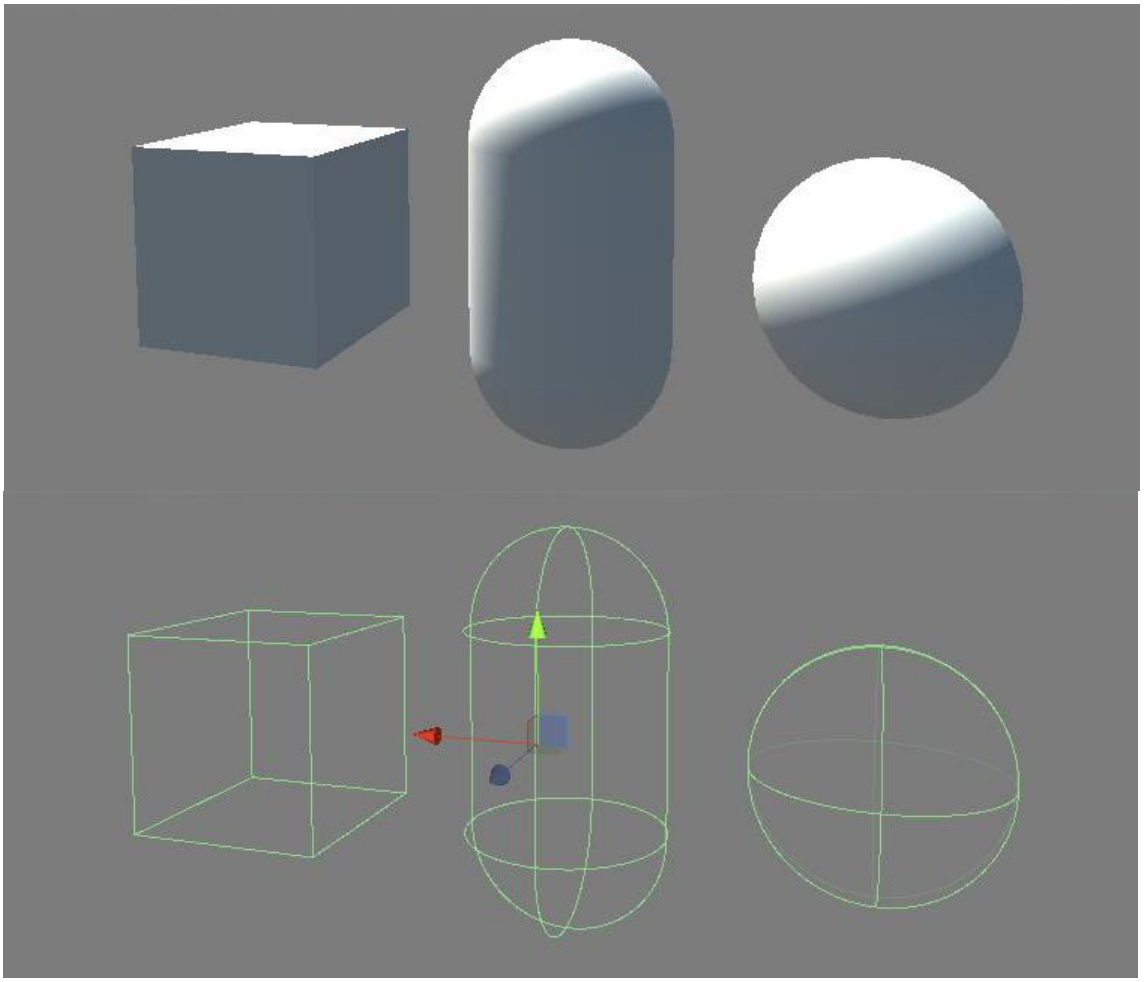

Figura 50. Box, Capsule y Sphere Colliders.

El *Mesh Collider*, aunque requiere más recursos de procesamiento, es el más idóneo para objetos sin una forma geométrica concreta. Como indica su nombre, se basa en un mallado que se adapta perfectamente al contorno del objeto, por muy irregular que sea. Esto se consigue al pasarle como parámetro el mallado importado, hecho con un software de modelado externo, que en este caso es Blender. Todos los mallados que se han usado en este proyecto venían incluidos en el modelo del edificio importado del TFG de Luis Pérez Pau, como bien se indica al inicio de éste capítulo.

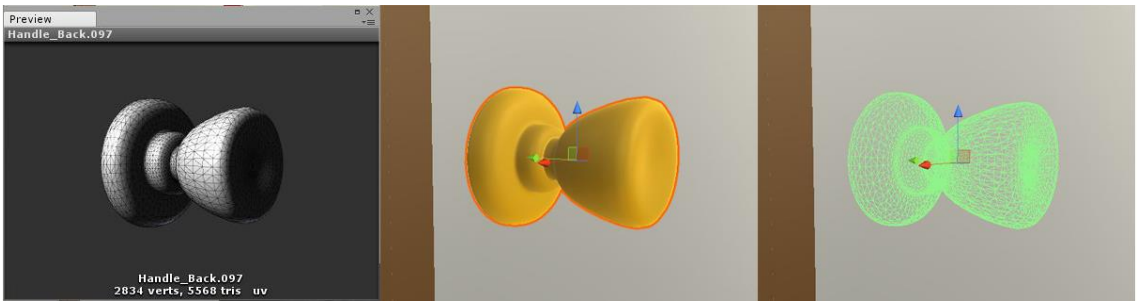

Figura 51. Mallado importado, objeto final y *Mesh Collider* del pomo de una puerta.

Una vez situados todos los *Colliders* en los objetos que se desee, ya podrían producirse colisiones entre ellos y, al detectarse, ya sería posible programar comportamientos mediante scripts. Esto es útil para el uso de los contenidos multimedia mostrados en el punto anterior, los cuales deberán accionarse mediante la colisión del dedo índice del usuario con las pegatinas de códigos QR situadas en ciertas vitrinas.

4.2.3.4 Iluminación

La iluminación es una parte básica e imprescindible del proyecto. Existen diferentes tipos luz. La más importante es la luz direccional (*Directional Light*), la cual simula a la perfección los rayos del Sol. Además, no tiene un punto fuente de luz concreto, por lo que es posible colocarla en cualquier parte de la escena siendo capaz de iluminar todo.

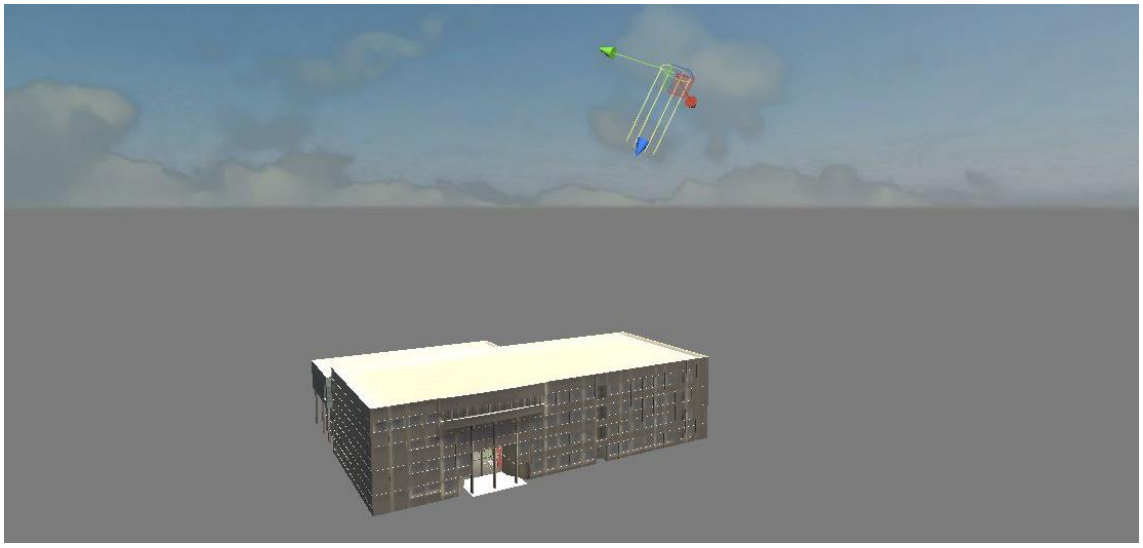

Figura 52. Vista exterior del edificio de la ETSIT con *Directional Light* simulando el Sol.

Por otra parte, están las luces que sí tienen un punto fuente concreto y se usan para iluminar zonas relativamente pequeñas y puntuales. La *Point Light* sería similar a una bombilla radiando luz en todas las direcciones. La *Spot Light*, en cambio, solo ilumina un ángulo concreto.

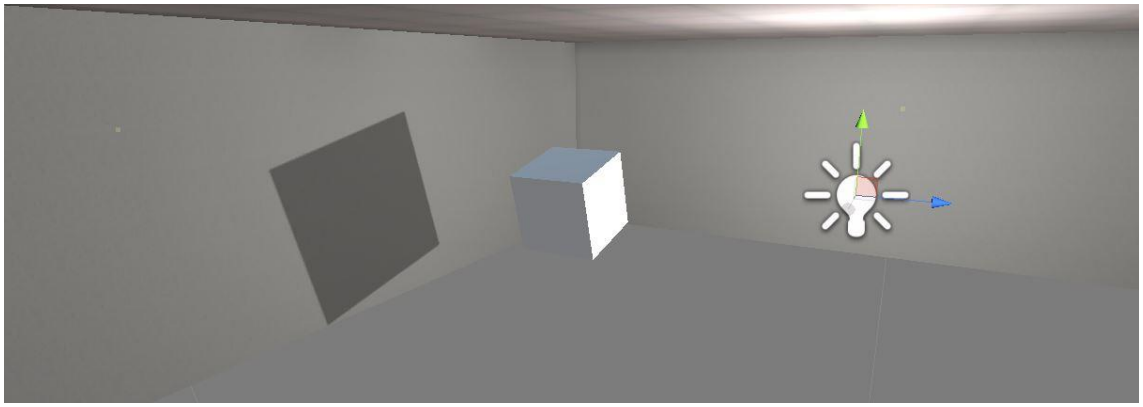

Figura 53. Ejemplo de *Point Light* sobre un cubo.

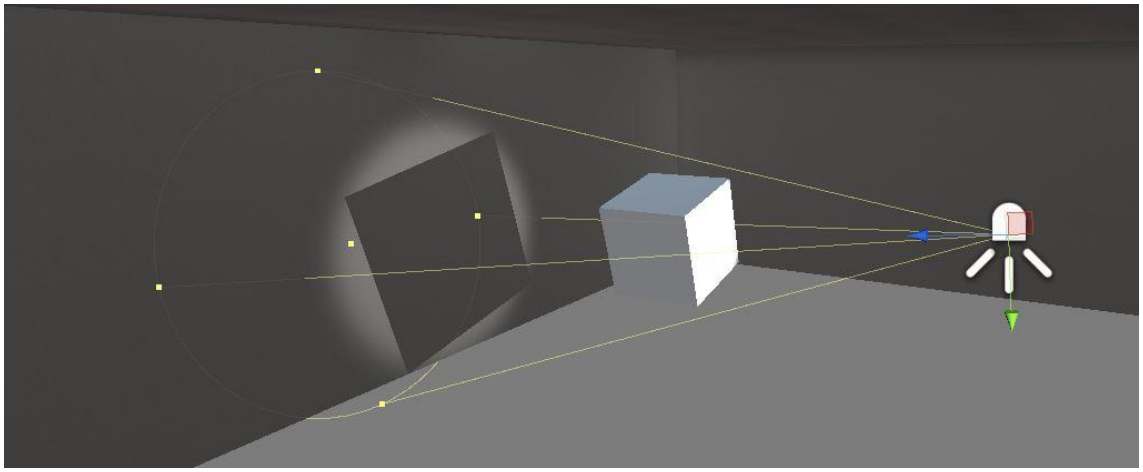

Figura 54. Ejemplo de *Spot Light* sobre un cubo.

Éstas últimas, suelen colocarse en lámparas o focos como es el caso de este proyecto.

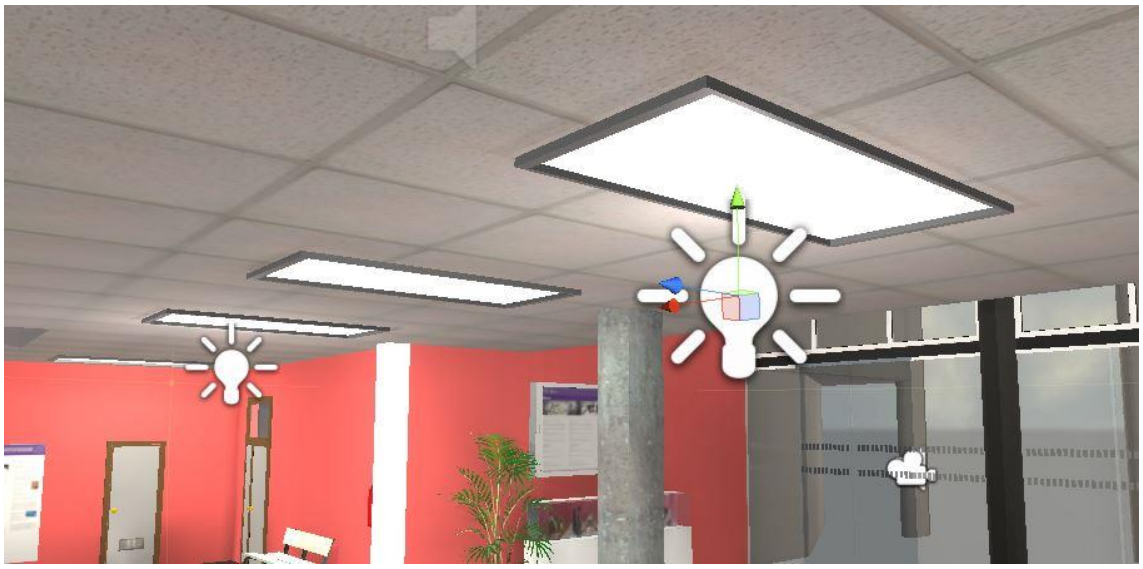

Figura 55. *Point Light* colocados sobre dos lámparas de techo.

En cuanto a los objetos en sí, es posible añadirles materiales que emitan luz propia. Esto es útil para conseguir una buena iluminación en zonas bastante amplias y cerradas haciendo que, por ejemplo, las luces de techo, como en este proyecto, emitan su propia luz sin tener que colocar un *Point Light*. También es interesante para resaltar algún objeto en concreto por querer darle más importancia o si no le llega bien la luz ambiente. En la imagen de arriba puede observarse la propia emisión de luz de las lámparas de techo y, además, al ser la entrada principal del edificio, se ha reforzado la iluminación con dos *Point Light* en esos dos puntos únicamente.

4.3 Navegación

4.3.1 Configuración de las gafas Oculus Rift

Nada más abrir la aplicación de Oculus, el primer paso es configurar y calibrar las gafas, los mandos controladores y los sensores de movimiento. Todo, siguiendo las indicaciones que aparecen en pantalla de forma muy fácil e intuitiva.

Primero, se emparejan los dos mandos controladores, se conectan las gafas y los sensores de movimiento al ordenador mediante los cables HDMI y USB 3.0, y se establece la altura del usuario que está llevando a cabo la configuración.

Seguidamente, se colocan los sensores con una distancia mínima entre ellos de 1 o 2 metros de manera que señalen a la zona de juego. Se coge un mando controlador y se siguen los pasos en pantalla para verificar que todos los sensores son capaces de seguir su movimiento. Seguidamente, se configura el Sistema Guardián, el cual establece unos límites dentro de la zona de juego para evitar no salirse de ella ni colisionar con ningún objeto cercano. Esto es muy importante hacerlo bien porque cuando el usuario lleva las gafas puestas no suele tener consciencia alguna de dónde está físicamente en la realidad, por lo que, al moverse, es muy fácil chocar contra cualquier objeto cercano.

Por último, el usuario se coloca las gafas ajustadas correctamente y ya puede empezar un mini tutorial para aprender a utilizar los mandos controladores (botones, acciones y gestos) y entender el entorno virtual de Oculus.

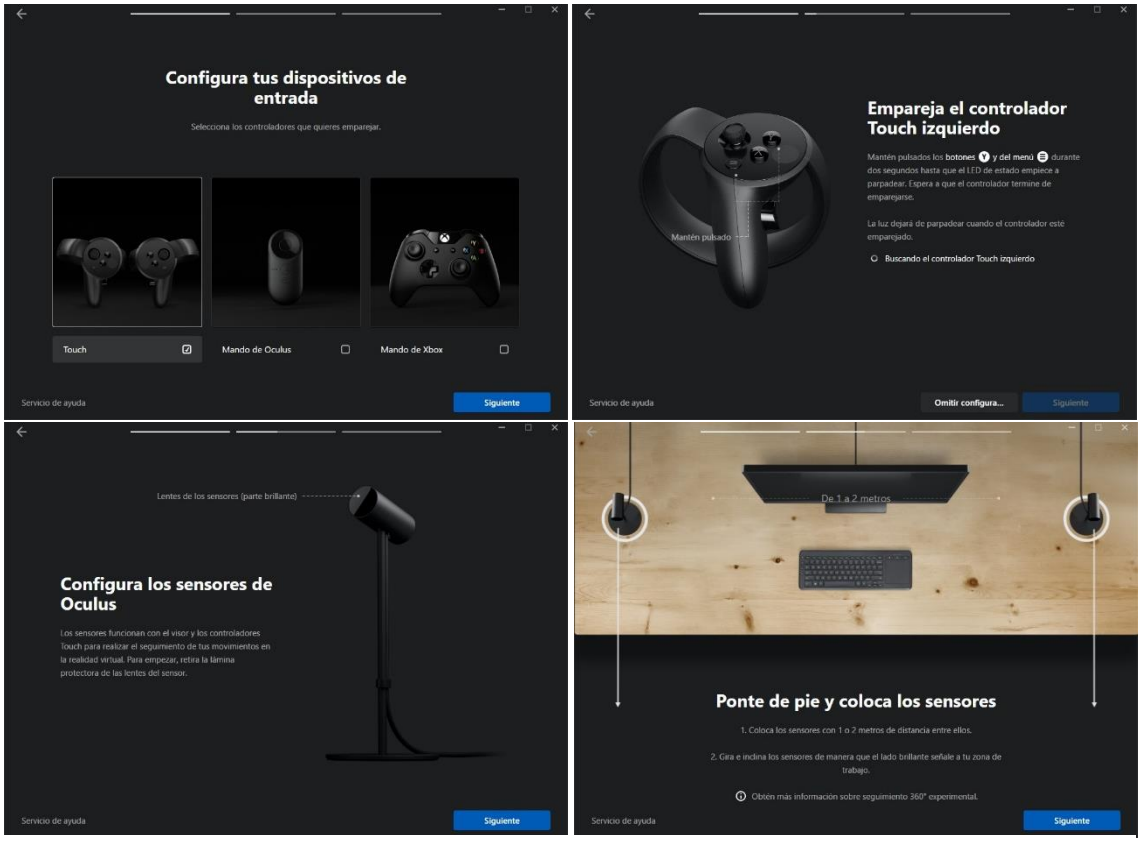

Figura 56. Configuración de controladores y sensores de Oculus Rift.

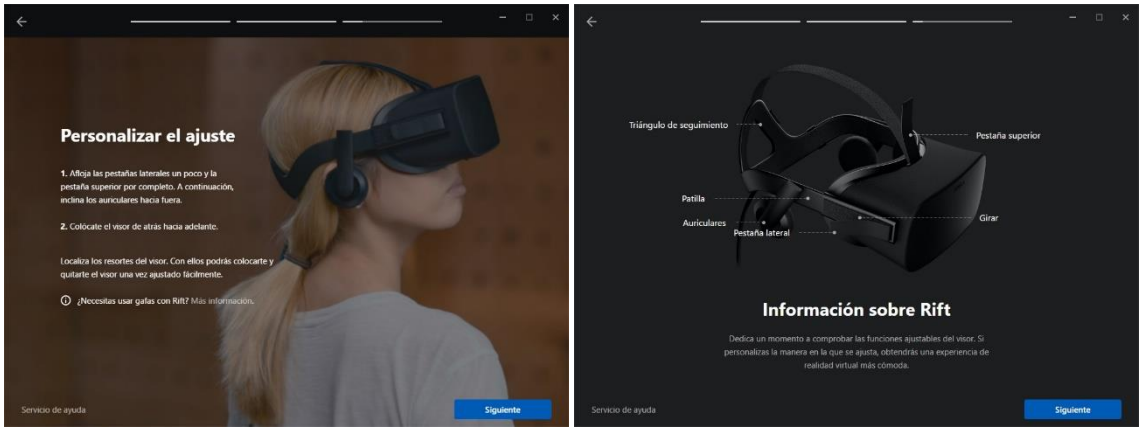

Figura 57. Configuración e información de las gafas (HMD) Oculus Rift.

4.3.2 Integración y uso en Unity

Volviendo al proyecto de Unity, una vez todo modelado y bien situado, el siguiente paso es hacer que el entorno sea "navegable", es decir, que el usuario pueda moverse por él.

Para ello, se hace uso del SDK de Oculus ya mencionado anteriormente. Este kit de desarrollo permitirá integrar las Oculus Rift en el proyecto de una forma muy sencilla. Desde la misma tienda (Asset Store) de Unity es posible descargar el paquete gratuito de utilidades buscando "Oculus Integration" y directamente se importa al proyecto. Esto añade varias carpetas con multitud de utilidades y soporte para desarrollo de renderizados, avatares y comportamientos, entre muchos otros.

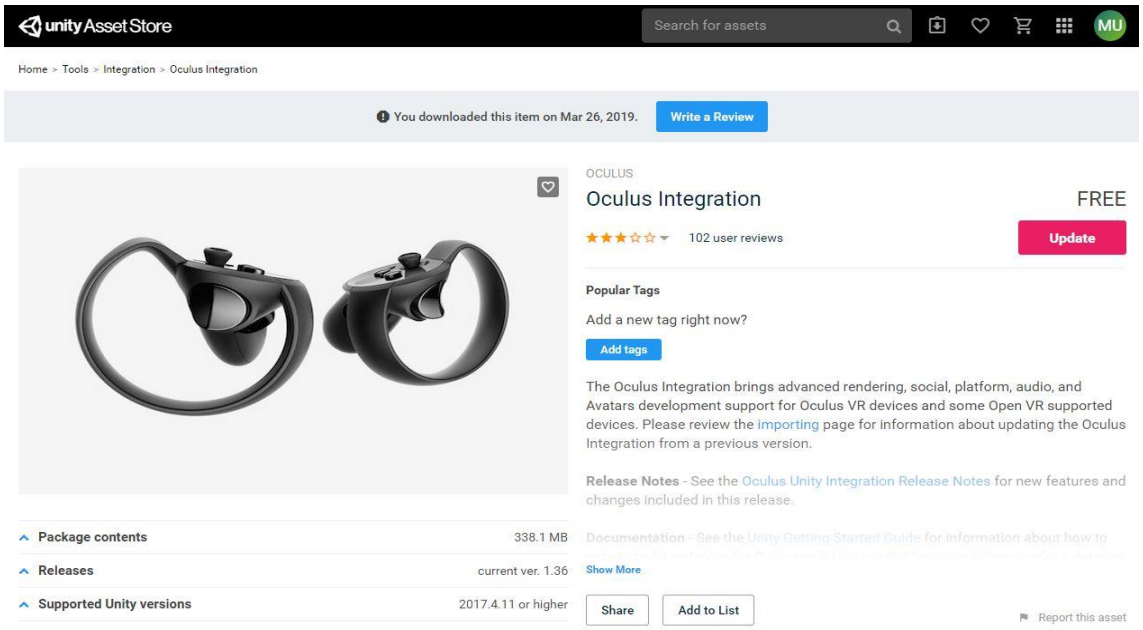

Figura 58. Paquete de utilidades de Oculus para descargar desde la Asset Store.

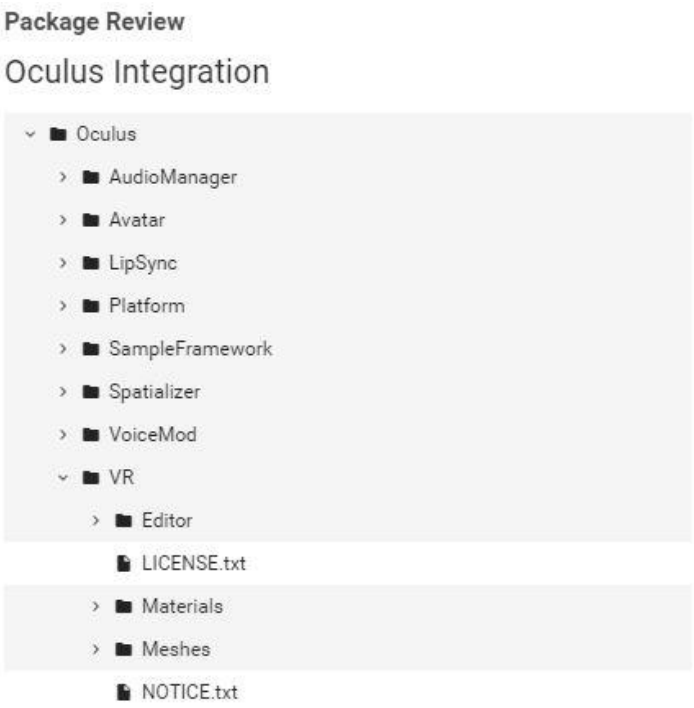

Figura 59. Contenido del paquete de utilidades de Oculus.

El paquete de utilidades contiene todo lo básico para hacer funcionar el entorno virtual en las gafas 3D. Dispone de avatares del cuerpo y manos o, si se prefiere, avatares de mandos controladores virtuales, además de todos los mallados, materiales y texturas necesarias para ellos. También tiene escenas ya creadas a modo de ejemplo para hacerse una idea de cómo quedaría antes de aplicarlo a la escena propia. Por último, pero no menos importante, dispone de multitud de scripts ya programados para controlar el comportamiento de la cámara y de las manos, modificar opciones y características y añadir acciones con el entorno, además de diversas utilidades interesantes. Cabe destacar que es un paquete muy completo y ofrece infinidad de recursos, pero el hecho de que esté en el paquete no implica una inmediata activación, sino que es necesario entender cada opción y saber cómo integrarlo y aplicarlo al proyecto.

4.3.2.1 OVRPlayerController

Una vez todo descargado, primero, se sitúa en la escena el Prefab "OVRPlayerController", el cual hará de "jugador", en este caso el usuario que llevará las gafas puestas.

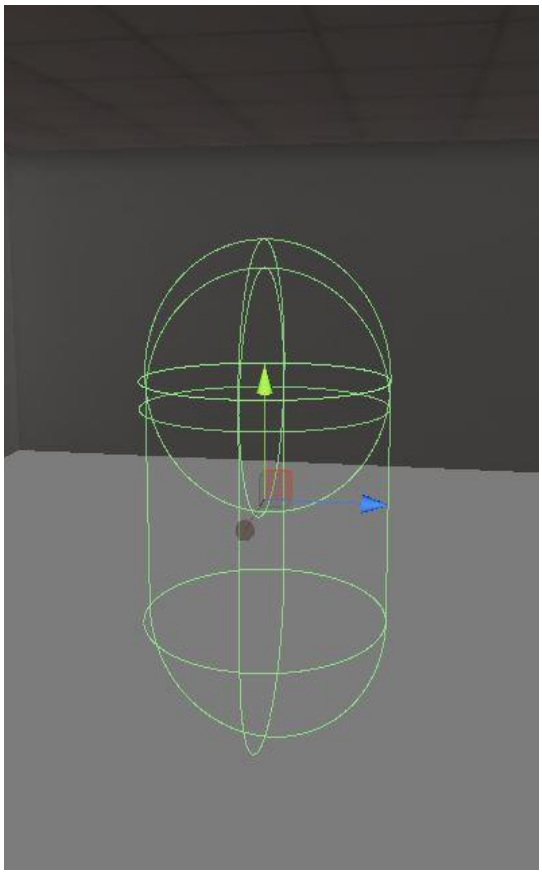

Figura 60. *Capsule Collider* del OVRPlayerController.

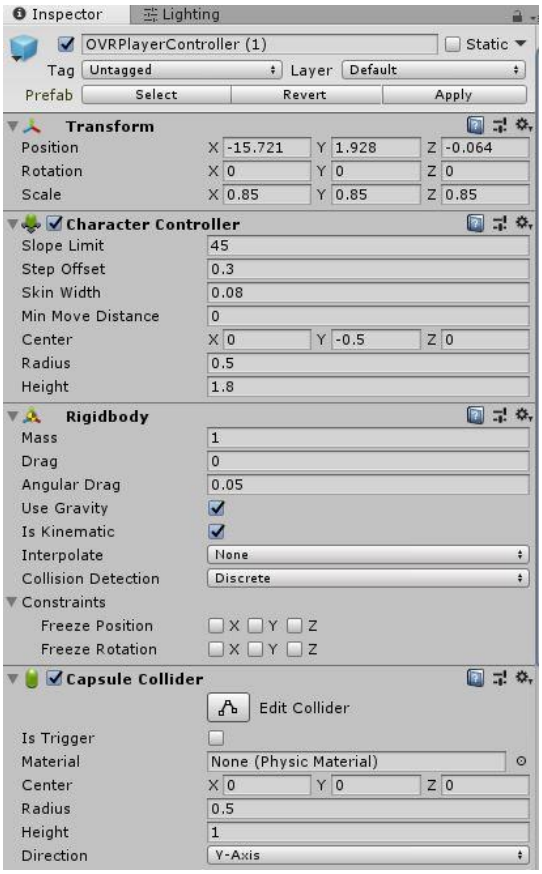

Figura 61. Ventana de *Inspector* del OVRPlayerController.

En la Figura 60 de arriba pueden observarse las siluetas que simularán el contorno de la cabeza y del cuerpo, por separado. Esto se consigue usando dos *Capsule Collider* como los ya vistos en el punto *4.2.3.3 Colisiones* para establecer los límites del cuerpo humano y poder detectar colisiones con otros objetos del entorno. Son dos porque se trata la cabeza y el cuerpo de forma independiente. El componente *Character Controller* de la Figura 61 es el encargado de controlar los valores de la altura, la anchura, el centro y el radio de la cápsula de la cabeza, totalmente modificables. Y el componente *Capsule Collider* de forma análoga con la cápsula del cuerpo. El otro componente, del que no se ha hablado todavía, es el *Rigidbody*, el cual hace que el objeto asociado adquiera un cuerpo rígido y se comporte bajo el control de la física. Esto, básicamente, sirve para evitar que el cuerpo del jugador atraviese el suelo, paredes u objetos de la escena. Además, son configurables parámetros como la masa en kilogramos, la resistencia del aire o la acción de la gravedad. Este componente logra un movimiento mucho más realista.

4.3.2.2 OVRCameraRig

Dentro del prefab OVRPlayerController, se coloca un prefab cámara OVRCameraRig ya configurada con sus scripts correspondientes que simula a la perfección el movimiento de la cabeza humana. Ésta, a su vez, lleva adheridos puntos de seguimiento para ambos ojos por separado (LeftEyeAnchor y RightEyeAnchor) y para los controles de ambas manos (LeftHandAnchor y RightHandAnchor), todo con sus componentes totalmente modificables.

Figura 62. Árbol de jerarquía del OVRCameraRig.

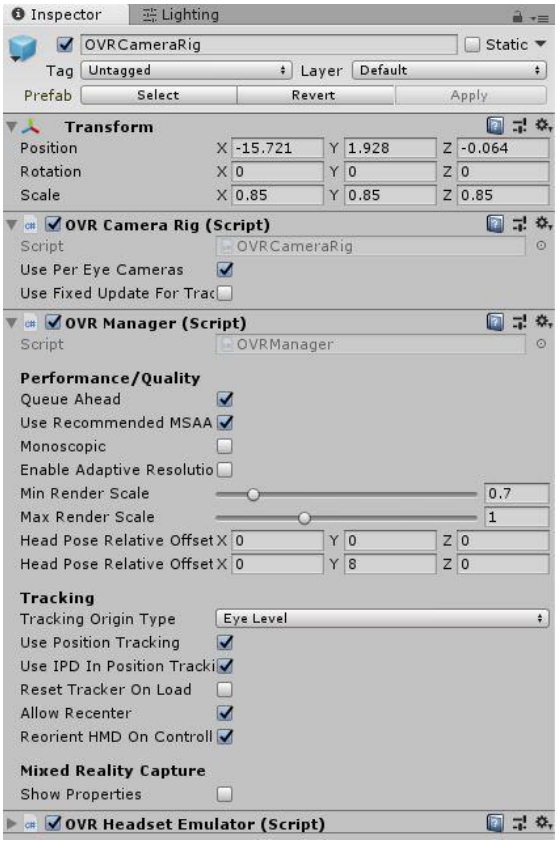

Figura 63. Ventana de *Inspector* del OVRCameraRig.

En la Figura 63 pueden observarse los dos scripts de configuración de la cámara. Con ellos, es posible modificar opciones para usar una sola cámara con una vista centrada o usar dos cámaras, situadas una en cada ojo. También son importantes las opciones de rendimiento y calidad para optimizar el funcionamiento general de la aplicación. Y, por último, opciones de seguimiento, posición y demás.

En cuanto a los componentes de los puntos de seguimiento de ojos y manos, pueden observarse en las Figuras 64 y 65.

Los ojos llevan asociados una cámara idéntica cada uno con opciones de proporciones, perspectivas, tamaño del campo de visión, profundidad, renderizado y demás. Y es importante configurar ambos ojos con las mismas opciones para lograr un efecto natural y real.

Las manos, al igual que los ojos, llevan asociados los mismos componentes tanto en la izquierda como en la derecha. Básicamente se trata de un *Rigidbody* como el descrito en el OVRPlayerController y un *Sphere Collider* para la detección de colisiones. Cabe añadir que, en este último componente, se ha activado la opción “Is Trigger”, la cual indica que éste *Collider* podrá ser usado como lanzador de eventos, es decir, para interactuar con otros objetos y poder activar los contenidos multimedia del museo. Por último, este objeto lleva también asociado un script de comportamiento para darle a la mano la capacidad de coger y manipular objetos y aumentar, más si cabe, su realismo.

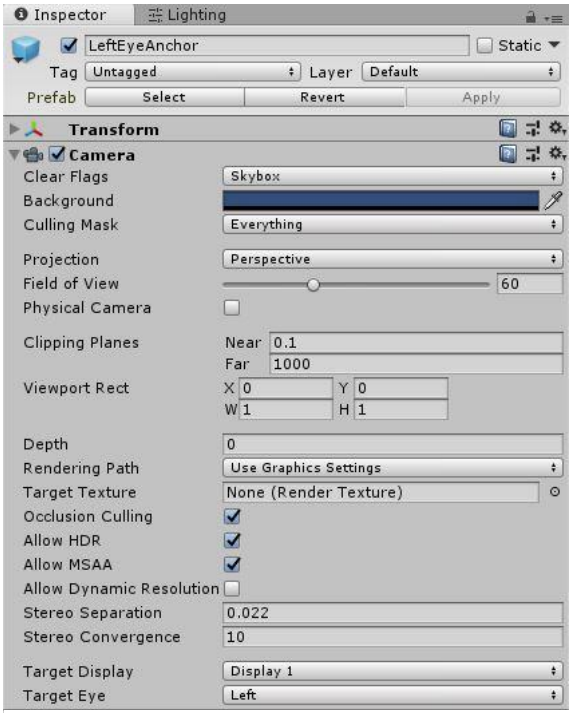

Figura 64. Ventana *Inspector* del LeftEyeAnchor.

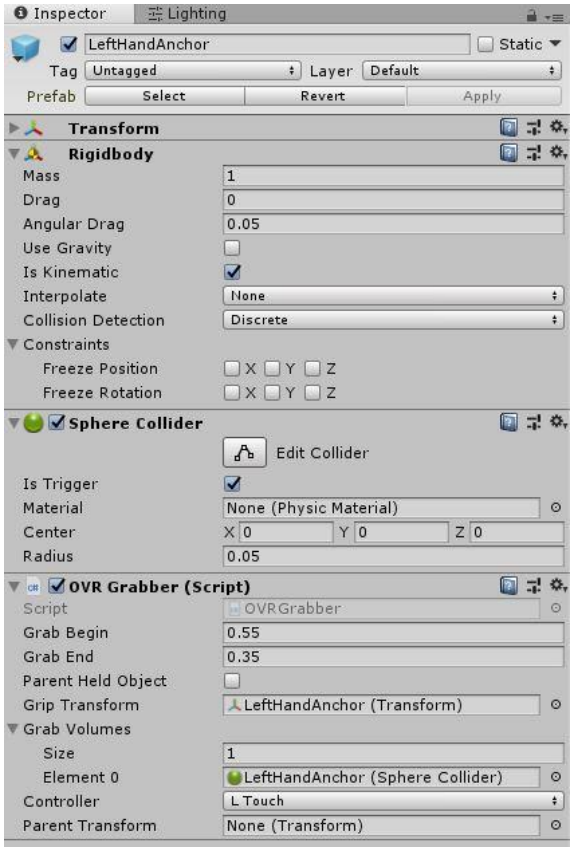

Figura 65. Ventana de *Inspector* del LeftHandAnchor.

4.3.2.3 LocalAvatar

Llegados a este punto, ya habría un funcionamiento completo y el usuario podría moverse e interactuar con el entorno. Sin embargo, el toque final es añadir los avatares de las partes del cuerpo que se desee.

Esto solamente es un aspecto visual pero que mejora sustancialmente la experiencia y la hace todavía más real.

Para llevarlo a cabo, basta con añadir el prefab LocalAvatar al ya existente OVRCameraRig de la cámara. Luego, se desactivan los avatares de los mandos controladores (controller_left y controller_right) ya que, en este caso, se ha elegido usar avatares de las manos.

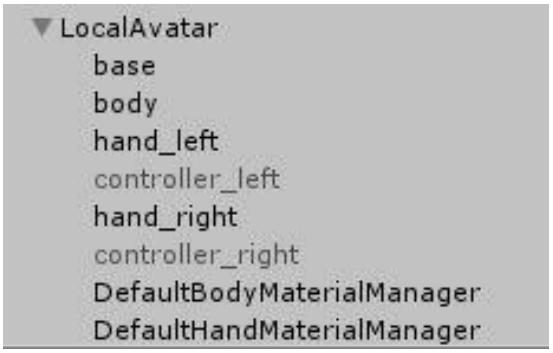

Figura 66. Árbol de jerarquía del LocalAvatar.

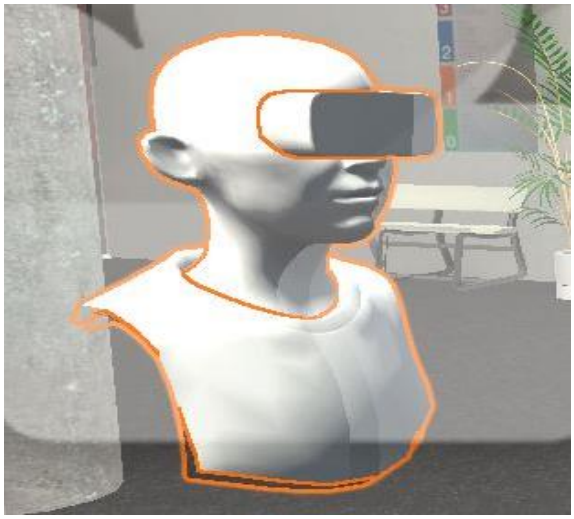

Figura 67. Avatar del busto con las gafas puestas.

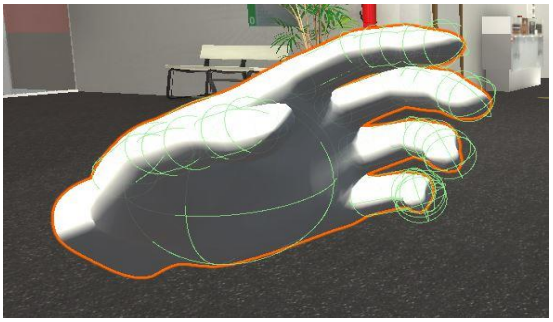

Figura 68. Avatar de la mano.

Para terminar esta parte, se muestra en la Figura 69 la jerarquía completa del OVRPlayerController. Éste incluye todo lo visto en los últimos puntos: la cámara, los avatares y todas las opciones de configuración y personalización.

A continuación, en el siguiente punto, se tratará la configuración para obtener el mejor movimiento posible del usuario, siempre intentando que sea lo más fiel a la realidad, y se solucionarán pequeños problemas e inconvenientes.

Figura 69. Árbol de jerarquía del OVRPlayerController.

4.3.2.4 Configuración del movimiento

Ahora, ya sería posible moverse libremente por el entorno, aunque hay un inconveniente. La aplicación está diseñada para usarse en un espacio cerrado y más bien pequeño como una habitación, ya que las gafas están conectadas por cable a un ordenador. Entonces, el jugador puede moverse unos pocos pasos y girar sobre sí mismo, pero con limitaciones obvias de espacio. Para eliminar esta limitación y poder moverse por todo el museo, con los joysticks de los mandos controladores también es posible andar y girar sin necesidad de que la persona se mueva del sitio, además de subir y bajar escaleras como si estuviera presente en el propio museo.

Así pues, una vez cogido el mando controlador izquierdo y correctamente posicionado en la mano, moviendo el joystick hacia adelante o hacia atrás permite hacer avanzar o retroceder al usuario en línea recta. Además, si mientras se anda se presiona y mantiene el botón del dedo índice, aumentará la velocidad hasta soltar el botón o dejar de andar. Sabiendo esto, para cambiar de dirección basta con avanzar con el joystick y girar la cabeza de forma natural hacia cualquier lado. El usuario avanzará en la dirección en la que esté observando.

De forma análoga, el joystick del mando controlador derecho permite al usuario girar la cámara en sentido horario o anti horario sin desplazarse del sitio. Ésta, es una opción disponible más pero totalmente desaconsejable, por lo que está desactivada por defecto. La razón es que girar con el joystick es una forma irreal y antinatural de usar la realidad virtual y puede provocar graves mareos. En lugar de eso es preferible girar la cabeza de forma más natural como se haría en la vida real. Una persona en la vida real, cuando quiere mirar en otra dirección, no deja la cabeza quieta y mueve el mundo, sino que gira la cabeza hacia esa dirección. Cabe recordar que esto no es un videojuego mostrado en la pantalla de un monitor, sino que es un mundo virtual que imita a la realidad y cada movimiento debe ser lo más natural posible. Ahí radica la gran capacidad de inmersión de este tipo de realidad virtual con gafas 3D.

Para optimizar el movimiento, se usa el script asociado al prefab OVRPlayerController que se muestra en la Figura 70. En él es posible modificar la aceleración, la inercia, activar o desactivar el desplazamiento y la rotación (los dos joysticks mencionados antes) y demás.

Las opciones del script más críticas han sido las dos primeras. Ha sido necesario hacer muchas pruebas con diferentes valores de aceleración y de inercia para conseguir reducir al máximo la sensación de mareo, ya que es uno de los aspectos más importantes de cualquier aplicación de realidad virtual.

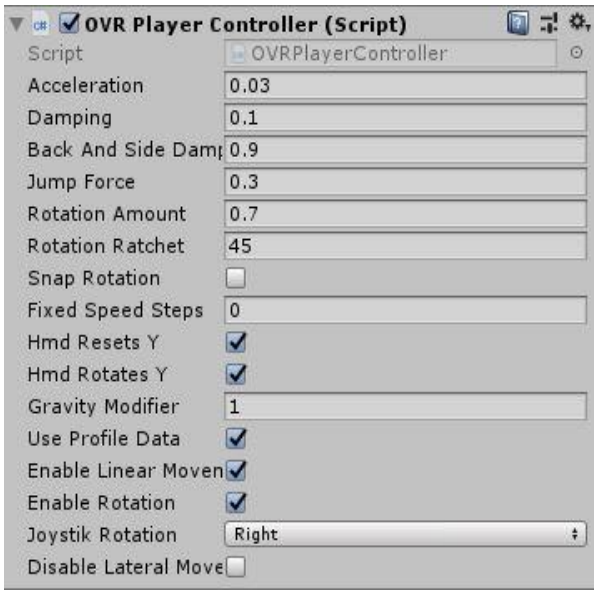

Figura 70. Script de configuración del OVRPlayerController.

4.4 Interacción

La última parte del desarrollo y la que va a darle a la aplicación mayor sensación de realismo es la interacción del usuario con los objetos del entorno. Es la capacidad de dotarlo de “vida”.

4.4.1 Contenidos multimedia de realidad aumentada

El Museo de la Telecomunicación Vicente Miralles Segarra cuenta con una aplicación móvil de realidad aumentada. Esta aplicación se descarga a un Smartphone o Tablet y permite al usuario disfrutar de contenidos multimedia durante la visita. Una vez abierta la aplicación, se activa la cámara del dispositivo y al enfocar con ella unos códigos QR situados en determinadas vitrinas, se reproduce un vídeo o un audio referente al objeto expuesto correspondiente. Esto es una forma de enriquecer la visita mostrando ejemplos de uso o explicaciones de los objetos expuestos a la vez que se va recorriendo el museo.

4.4.2 Contenidos multimedia de realidad virtual

El objetivo de la aplicación de realidad virtual de este proyecto es el de desarrollar una versión del museo lo más fiel a la realidad posible. Así pues, se han añadido de forma virtual los mismos contenidos multimedia que usa la aplicación de realidad aumentada.

Son un total de trece, repartidos por todo el museo, de los cuales diez son vídeos y tres son grabaciones de audio. Los objetos expuestos que los contienen son los siguientes:

- **Telégrafo Morse.** Dispone de un vídeo que muestra un ejemplo de uso de un manipulador de telégrafo morse por parte de Sebastián Olivé, primer presidente de la Asociación de Amigos del Telégrafo de España e Ingeniero Técnico de Telecomunicación.

- **Mujeres Telegrafistas.** Dispone de una grabación de voz de la lectura de una circular de la Dirección General de Correos y Telégrafos del Ministerio de la Gobernación datado en 1864 sobre la admisión de mujeres para el servicio telegráfico en España, expuesta en el interior de la vitrina.

- **Sistema Telegráfico Baudot.** Dispone de un pequeño vídeo documental titulado *A bordo del Castillo Olmedo* y basado en el texto de Vicente Miralles Mora, el cual narra una experiencia de su juventud, en 1951, acompañando a su padre Vicente Miralles Segarra, primer Ingeniero de Telecomunicación valenciano que da nombre al museo, en la tarea de reparar una avería en el cable telegráfico submarino València - Palma de Mallorca.

- **Telefonía de sobremesa.** Dispone de un vídeo de que muestra un ejemplo de uso de un teléfono de sobremesa de disco de la década de los 50.

- **Receptor de TV B/N Philips.** Dispone de un vídeo con un extracto del famoso programa de RTVE de los años 70 *Un, dos, tres… responda otra vez.*

- **Radio Philips a válvulas.** Dispone de una grabación de audio de la sintonización de un fragmento de la conocida aria *La donna è mobile* del compositor italiano Giuseppe Verdi.

- **Radio-cassette.** Dispone de un vídeo que muestra dos de los radio-cassettes expuestos en funcionamiento: el Telefunken de los años 60 y el Sanyo de los 80.

- **Fonógrafo y Gramófono.** Ambos en la misma vitrina, disponen de un vídeo cada uno donde se muestra el montaje y el funcionamiento del fonógrafo de los años 70 y del gramófono de los 20 expuestos.

- **Magnetófono.** Dispone de un vídeo que muestra el funcionamiento del magnetófono Akai de los años 60 expuesto.

- **Tocadiscos.** Dispone de un vídeo que muestra la puesta a punto y el funcionamiento de un tocadiscos Picú.

- **Estación radiotelegráfica Marconi.** Dispone de una grabación de audio del sonido en funcionamiento de una estación radiotelegráfica y radiotelefónica Marconi.

- **E.A.J.30 Radio Onteniente.** Dispone de un vídeo donde aparece el profesor de la ETSIT Héctor García Miquel haciendo una explicación sobre una de las primeras emisoras de radio diseñadas y fabricadas por su abuelo Salvador Miquel, en funcionamiento en Onteniente desde 1933 y expuesta en el museo.

Para reproducir los contenidos multimedia, una vez dentro de la aplicación con las gafas puestas y los mandos en ambas manos, basta con situarse frente a una vitrina que disponga de pegatina con código QR, acercarse bastante al cristal y tocar el código con el dedo índice de cualquier mano. Inmediatamente se reproduce el vídeo o audio correspondiente al objeto u objetos expuestos.

4.4.2.1 Código QR

El hecho de que el activador de los contenidos multimedia sea un código QR es debido a querer respetar al máximo la apariencia del museo conforme a la realidad. Éste código tiene su función relacionada con la aplicación de realidad aumentada del museo. En lo que concierne a la realidad virtual y a este proyecto, podría haberse utilizado cualquier otro activador, ya que lo importante es la acción de tocarlo independientemente de qué se toque. Hubiera sido más intuitivo colocar, por ejemplo, un botón con el texto "Reproducir" o "Play", sin embargo, se ha optado por respetar el código para mantener la similitud con el mundo real.

El modelado es simplemente un cubo redimensionado como los carteles ya descritos en este mismo capítulo, colocado en el cristal de la vitrina a la que corresponda y con una fotografía del código QR como textura.

Figura 71. Pegatina real del código QR.

Figura 72. Modelo virtual del código QR.

Una vez hecho y colocado el modelo, es necesario crear en su árbol de jerarquía cuatro nuevos objetos. Un punto de aparición para la pantalla temporal en la que se reproducirá el vídeo (SpawnPointPantalla) y otro para la base que sostendrá la pantalla para que no caiga al suelo (SpawnPointBase), los cuales se detallarán en el siguiente punto; y una fuente de vídeo (VideoSource) y su asociada fuente de audio (AudioSource).

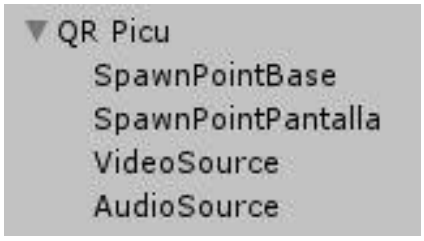

Figura 73. Árbol de jerarquía de un código QR.

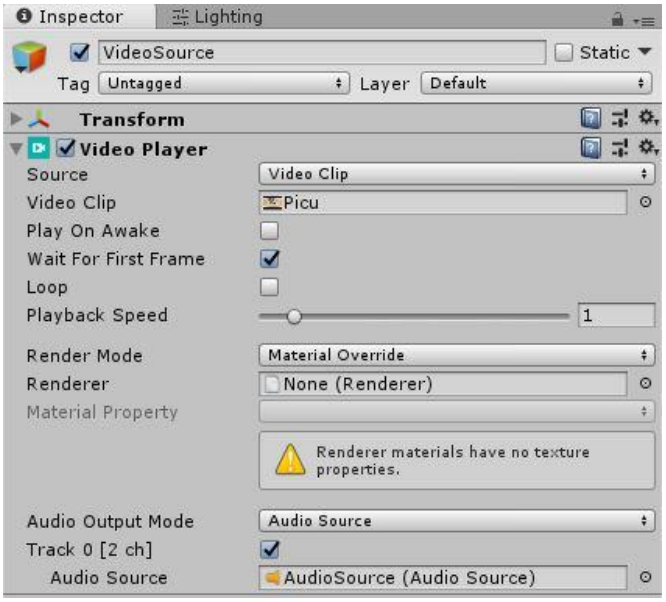

Figura 74. Ventana de *Inspector* de una VideoSource.

Como se puede apreciar en la Figura 74, el tipo de fuente de vídeo será un Video Clip. Luego, se añade el archivo de vídeo (Picu) al campo correspondiente y se elige el modo de renderizado "Material Override" para que el vídeo se reproduzca sobre un material existente como será una pantalla. Por último, se elige la fuente de audio asociada y se marca la casilla de 2 canales para activar el sonido estéreo.

En cuanto a los componentes para este objeto, como ya se ha visto en el punto *4.2.3.2 Objetos del museo*, es necesario añadirle un *Box Collider* para la detección de las colisiones, un *Rigidbody* para hacerle el cuerpo rígido y que pueda ser tocado, y un script de comportamiento para iniciar la reproducción de vídeo o audio al ser tocado. La parte del script, dado su complejidad, se tratará en el siguiente punto.

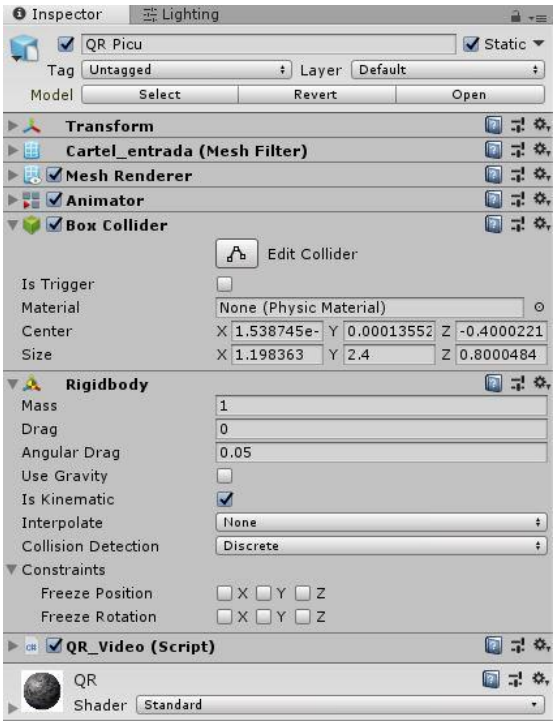

Figura 75. Ventana de *Inspector* de un código QR.

4.4.2.2 Script de comportamiento

Una vez todos los códigos correctamente colocados, se les tiene que dar el comportamiento de que cuando se toque dicho código se reproduzca el contenido. Para ello se hace uso de un script que se añadirá como componente al objeto del código QR. A continuación, se muestra un diagrama de flujo de dicho comportamiento.

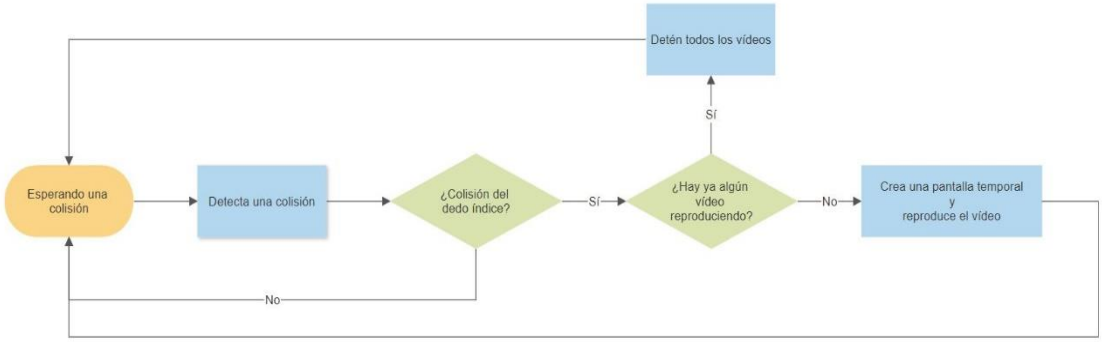

Figura 76. Diagrama de flujo de la función "OnTriggerEnter" para las colisiones.

El objeto se mantiene a la espera de detectar una colisión, ya sea por parte del usuario o por cualquier otro objeto de la escena. Al detectar una colisión, solo continuará con el proceso si esa colisión ha sido producida por el dedo índice, y solo el índice, de cualquier mano. De no ser así, la ignorará y volverá a mantenerse a la espera. Una vez detectada la colisión esperada, comprueba si el vídeo en cuestión ya está reproduciéndose y, de ser así, lo detendrá. Esto es de utilidad para el usuario porque hay dos vídeos de 2 y 5 minutos de duración y puede que no quiera verlos enteros. Por último, en caso de no haber nada en reproducción, procederá a crear una pantalla temporal e iniciará el vídeo.

Para plasmar este comportamiento, es necesario programarlo en lenguaje C# en un script para luego añadirlo como componente al objeto al que se le quiere dar dicho comportamiento, en este caso las pegatinas con los códigos QR.

```
void OnTriggerEnter(Collider other)
{
    if (other.gameObject.name.Contains("index3"))
    {
        if (!videoPlayer.isPlaying)
        {
            StopAllVideo();
            StopAllAudio();

            Transform baseT = Instantiate(prefabBase);
            baseT.transform.parent = transform;
            baseT.position = spawnPointBase.position;
            rotationBaseTx = transform.localEulerAngles.x;
            rotationBaseTy = transform.localEulerAngles.y;
            rotationBaseTz = transform.localEulerAngles.z;
            baseT.Rotate(rotationBaseTx, rotationBaseTy + 90, rotationBaseTz);

            Transform pantalla = Instantiate(prefabPantalla);
            pantalla.transform.parent = transform;
            pantalla.position = spawnPointPantalla.position;
            rotationPantallaX = transform.localEulerAngles.x;
            rotationPantallaY = transform.localEulerAngles.y;
            rotationPantallaZ = transform.localEulerAngles.z;
            pantalla.Rotate(rotationPantallaX, rotationPantallaY + 90, rotationPantallaZ);
            videoPlayer.targetMaterialRenderer = pantalla.GetComponent<Renderer>();

            baseT2 = GameObject.Find("BaseTope(Clone)");
            pantalla2 = GameObject.Find("PantallaPrefab(Clone)");

            videoPlayer.Play();
            StartCoroutine(CoUpdate());
        }
        else
        {
            StopAllVideo();
        }
    }
}
```

Figura 77. Código en C# de la función "OnTriggerEnter" para las colisiones.

La función principal es OnTriggerEnter(Collider), la cual detecta el contacto por colisión de cualquier objeto de la escena. Como ya se ha explicado en el punto *4.2.2 Creación de objetos*, las colisiones se producen cuando el *Collider* (silueta o contorno) de dos objetos de la escena se tocan.

Como el *Collider* que interesa es el del dedo índice, se establece un condicional para que solo continúe la ejecución del código si el nombre del objeto del que procede la colisión es "index3". Para conseguir que la colisión solo funcionara con la punta del dedo índice se tuvo que modificar el script original de las manos para separar cada dedo y cada falange de manera que la tercera falange del dedo índice recibiera el nombre de "index3". Una vez la función detecta la colisión correcta, si ese mismo vídeo ya está reproduciéndose, inmediatamente lo detiene. Si no, procede con el resto del código.

Primero, busca por toda la escena, y si cualquier contenido multimedia de otra vitrina está en funcionamiento, lo detiene. A continuación, crea un nuevo objeto que será la base sobre la que se apoyará la pantalla donde se reproducirá el vídeo y la coloca en el punto de aparición (SpawnPointBase) definido en el apartado anterior y que se muestra en la Figura 73. Esta base será invisible por razones de estética. Luego, de forma análoga, crea otro objeto que será la pantalla y la coloca en su respectivo punto de aparición (SpawnPointPantalla). Después, guarda las referencias de la base y la pantalla creadas en dos variables para posteriormente, cuando el vídeo finalice o cuando el usuario lo pare, destruirlas. Y, finalmente, reproduce el vídeo.

Para terminar con el script, cabe nombrar que hay más líneas de código para detalles como el cambio de textura del código QR una vez empieza a reproducir para sustituirlo por un botón de "stop"; o los cambios de volumen de la música ambiental mientras se reproducen los contenidos multimedia para que no interfiera en ellos.

Figura 78. Vista virtual de la activación de un contenido multimedia al tocar el código QR.

Capítulo 5. Resultados

Se ha conseguido terminar el desarrollo de esta versión virtual del Museo de la Telecomunicación Vicente Miralles Segarra. Partiendo del modelo del edificio de la ETSIT usado en el TFG de Luís Pérez Pau, se ha completado, actualizado, adaptado e incluido todo lo necesario para usar la aplicación final en las gafas 3D Oculus Rift.

El resultado es una aplicación para gafas 3D que ofrece una visita virtual completa al museo con una experiencia novedosa, intuitiva y fácil de usar. Permite recorrer libremente el museo e interactuar con algunos objetos del entorno para reproducir contenido multimedia que ayude al usuario a complementar y enriquecer la visita.

5.1 Galería de imágenes

Dado que el verdadero potencial de esta aplicación radica en su uso mediante las gafas 3D y no es posible plasmar en imágenes la sensación de inmersión y realismo que éstas ofrecen, además de que no es físicamente posible hacer capturas de pantalla de lo que el visor de las gafas muestra, se pretende intentar aproximar esas sensaciones mediante una recopilación de imágenes de la aplicación vista a través de la pantalla de un monitor.

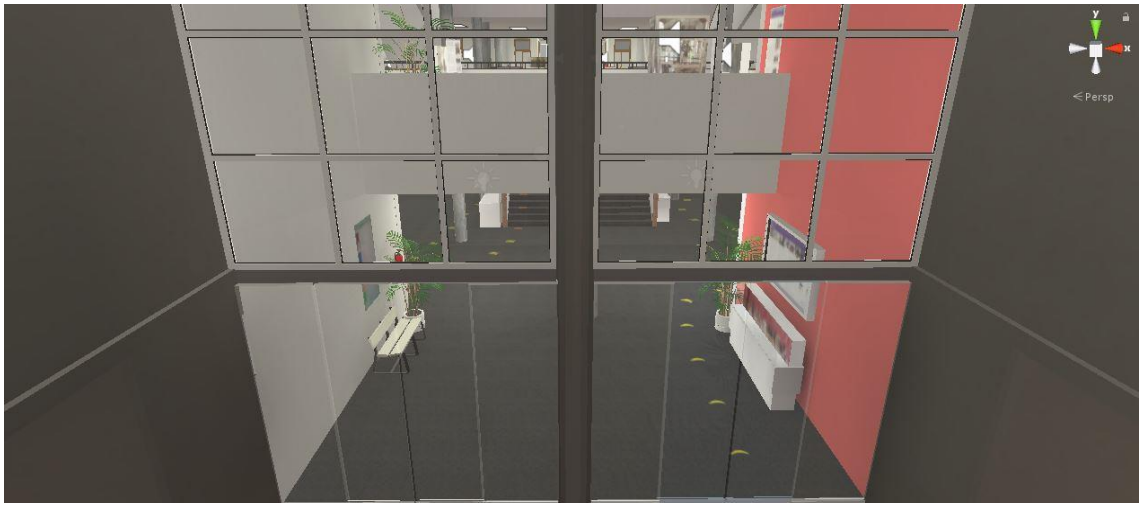

Figura 79. Entrada del edificio de la ETSIT.

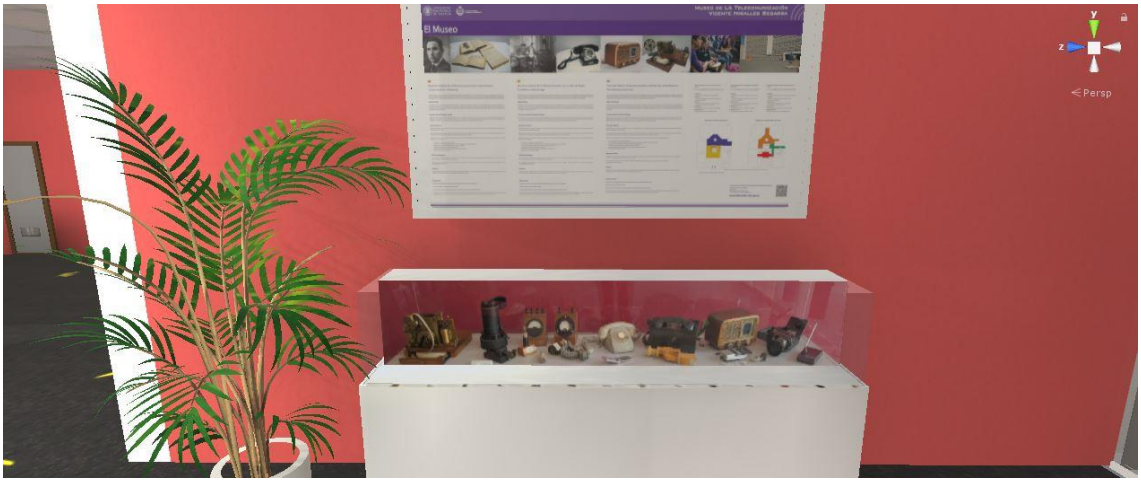

Figura 80. El Museo.

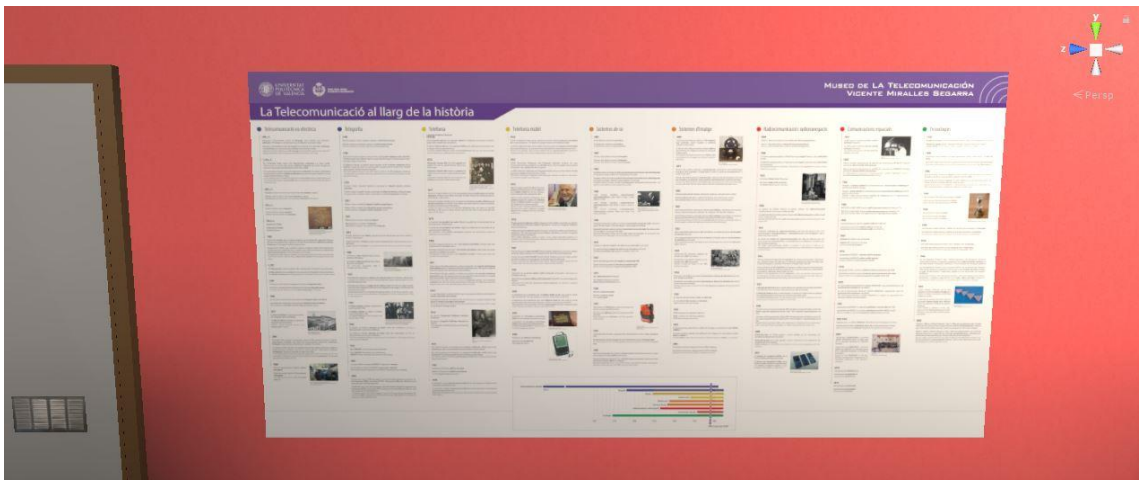

Figura 81. La Telecomunicació al llarg de la historia.

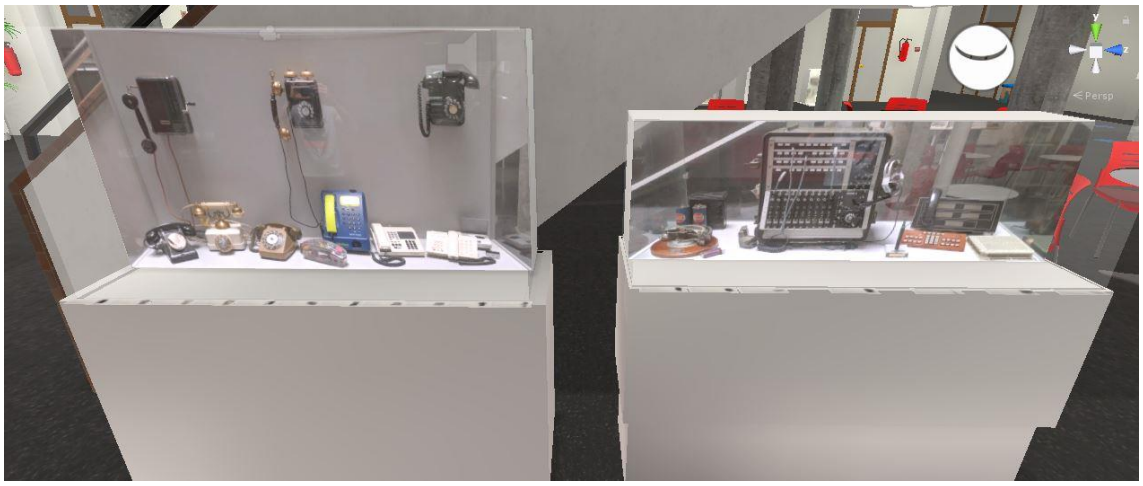

Figura 82. Conmutación Telefónica.

Figura 83. Heliógrafos de Campaña.

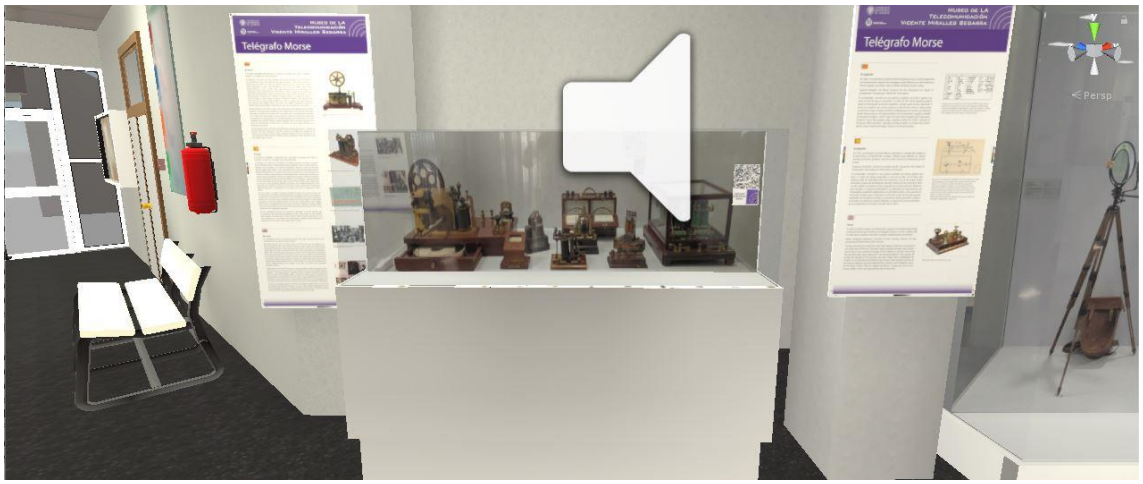

Figura 84. Telégrafo Morse.

Figura 85. Mujeres Telegrafistas y Telefonistas.

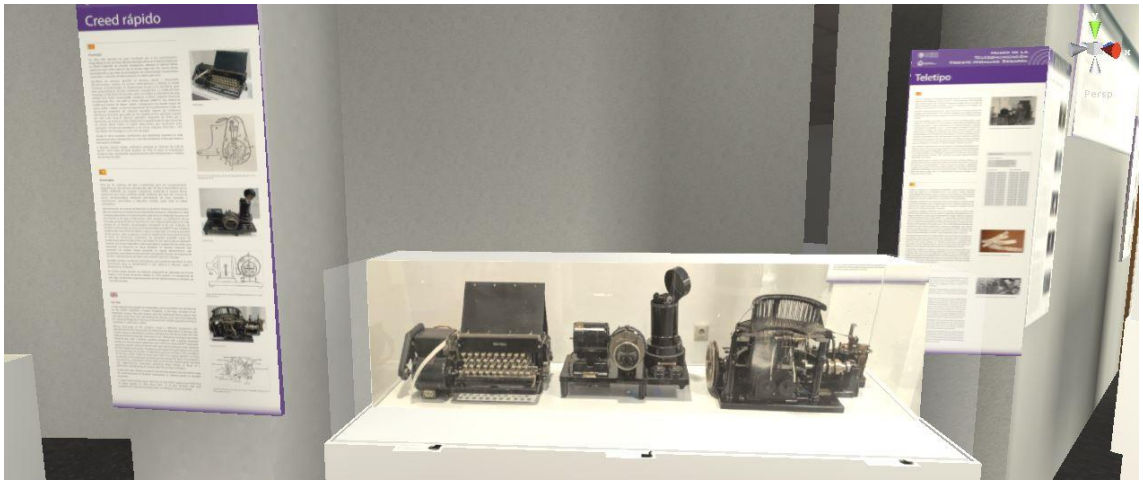

Figura 86. Creed rápido y Teletipo.

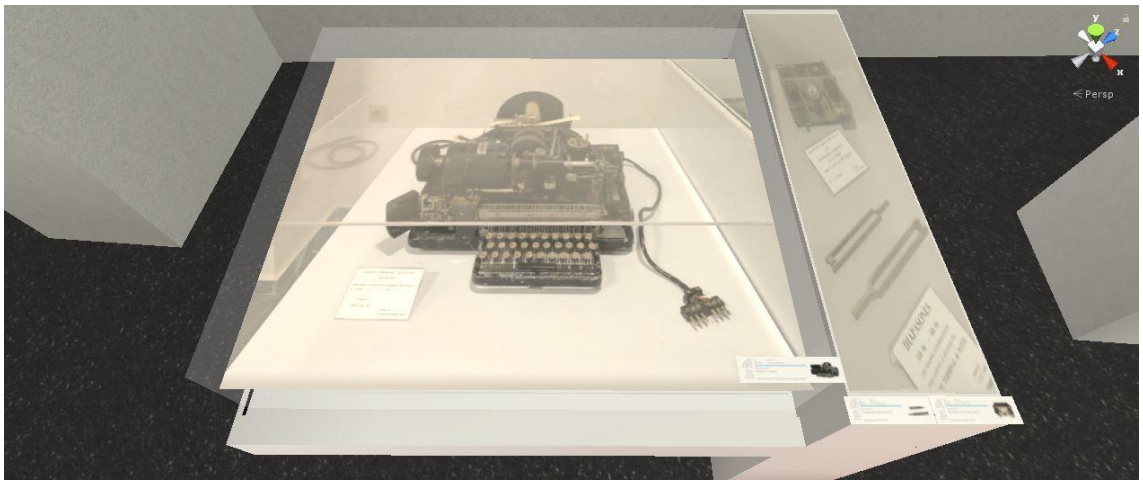

Figura 87. Teletipo Creed B7.

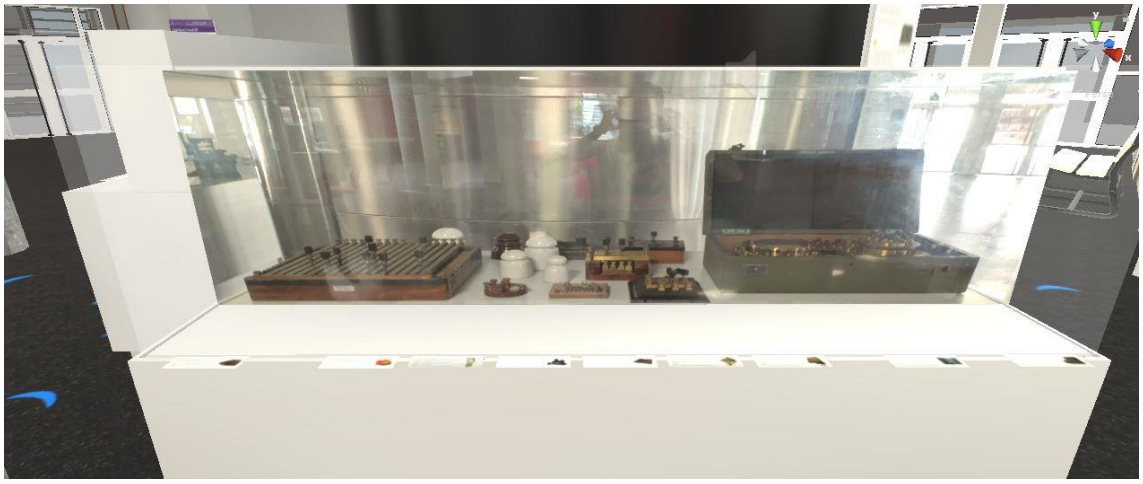

Figura 88. Galvanómetro universal Echenique.

Figura 89. Sistema Telegráfico Baudot.

Figura 90. Equipo multiplexor, Pentomat 600T, Sistema Télex y Olivetti ET300.

Figura 91. Conmutación Telefónica.

Figura 92. Conmutación Telefónica.

Figura 93. Evolución de la Telefonía Móvil.

Figura 94. Teléfonos y centralitas de campaña.

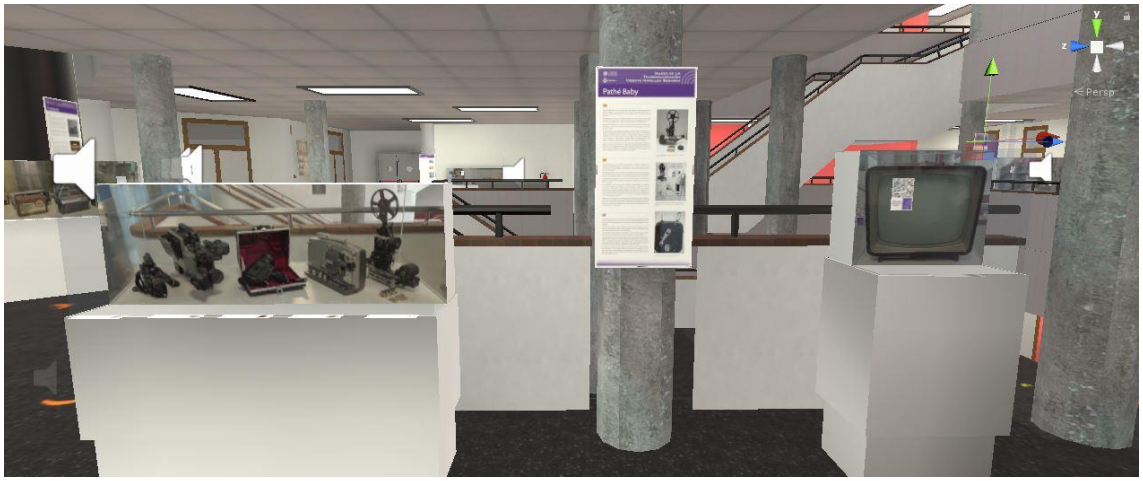

Figura 95. Pathé Baby y Receptor de TV B/N Philips.

Figura 96. Radio.

Figura 97. Radio.

Figura 98. Fonógrafo y Gramófono.

Figura 99. Magnetófono.

Figura 100. Tocadiscos.

Figura 101. Instrumentación de laboratorio.

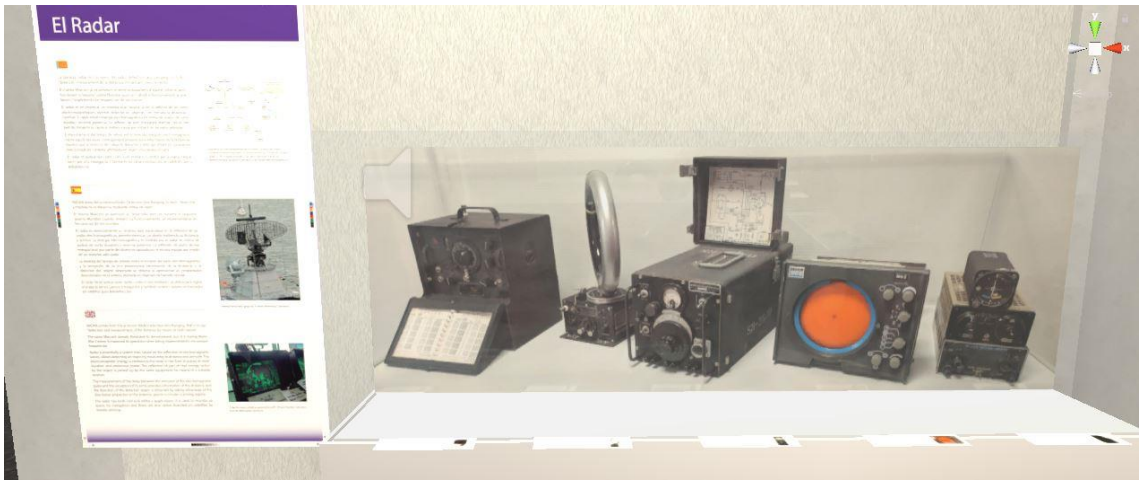

Figura 102. El Radar.

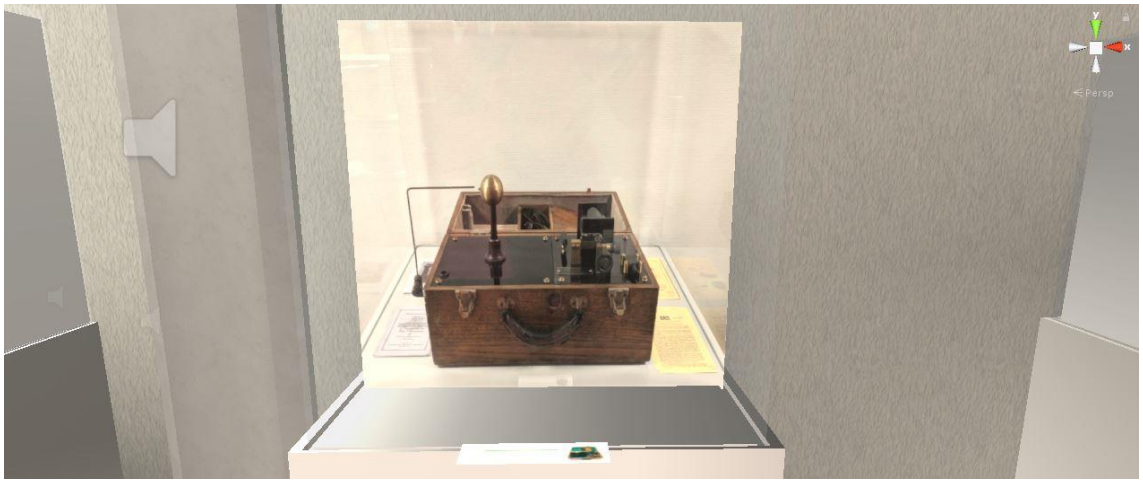

Figura 103. Aparato de Rayos X.

Figura 104. Instrumentación de medida.

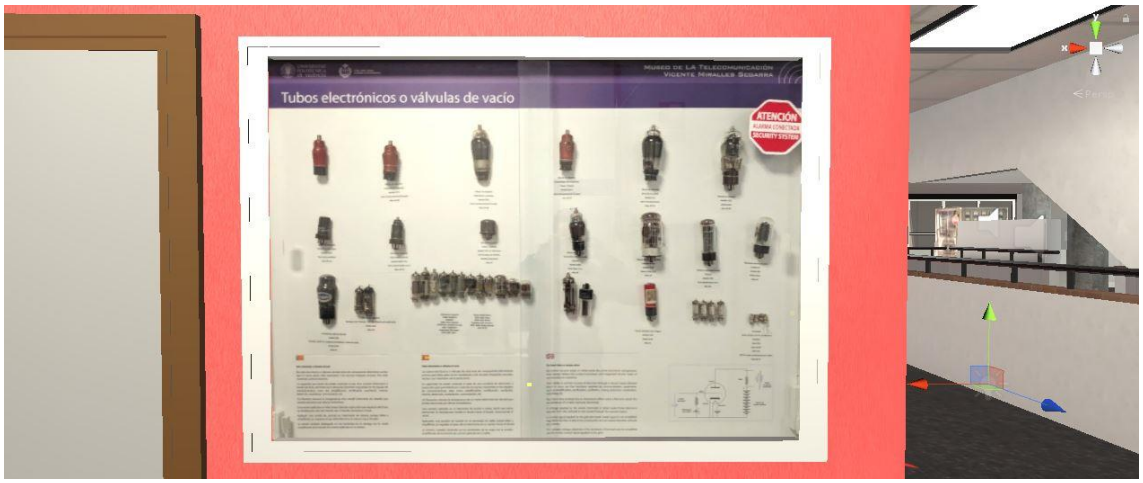

Figura 105. Tubos electrónicos o válvulas de vacío.

Figura 106. Walkie-talkie (transceptor).

Figura 107. Estación radiotelegráfica y radiotelefónica Marconi modelo A de 2W.

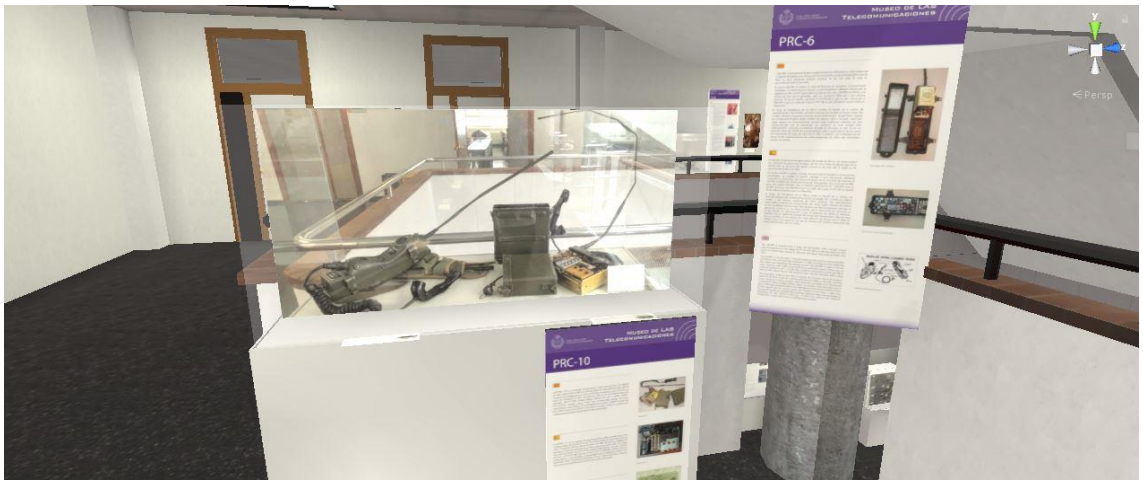

Figura 108. PRC-6 y PRC-10.

Figura 109. E.A.J.30 Radio Onteniente.

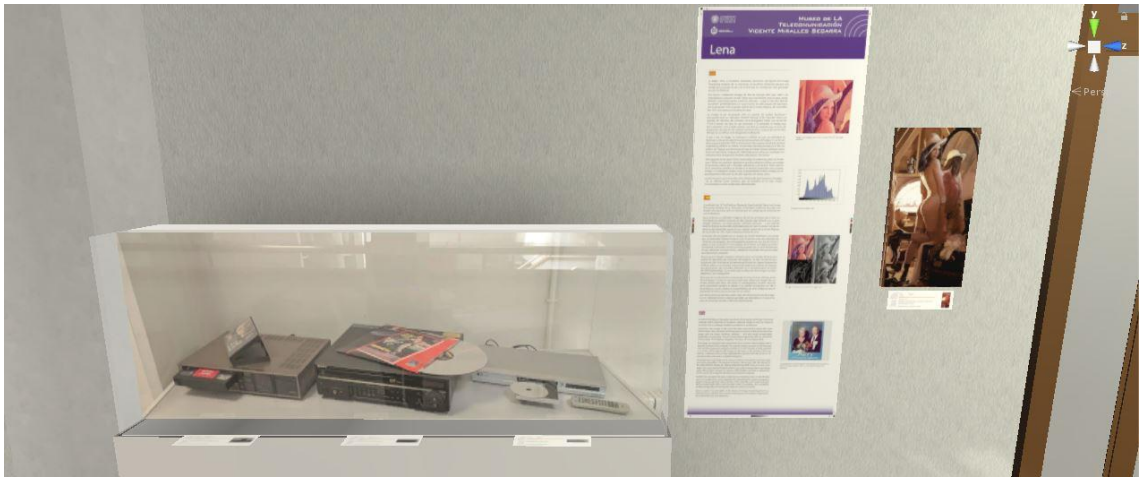

Figura 110. Lena.

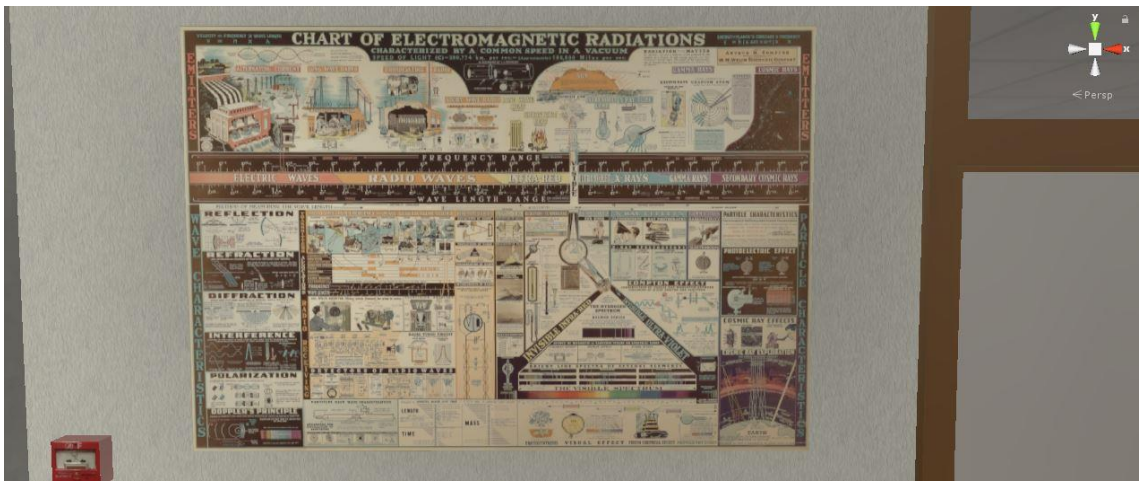

Figura 111. Chart of Electromagnetic Radiations.

Figura 112. Vista general del museo, Planta Baja.

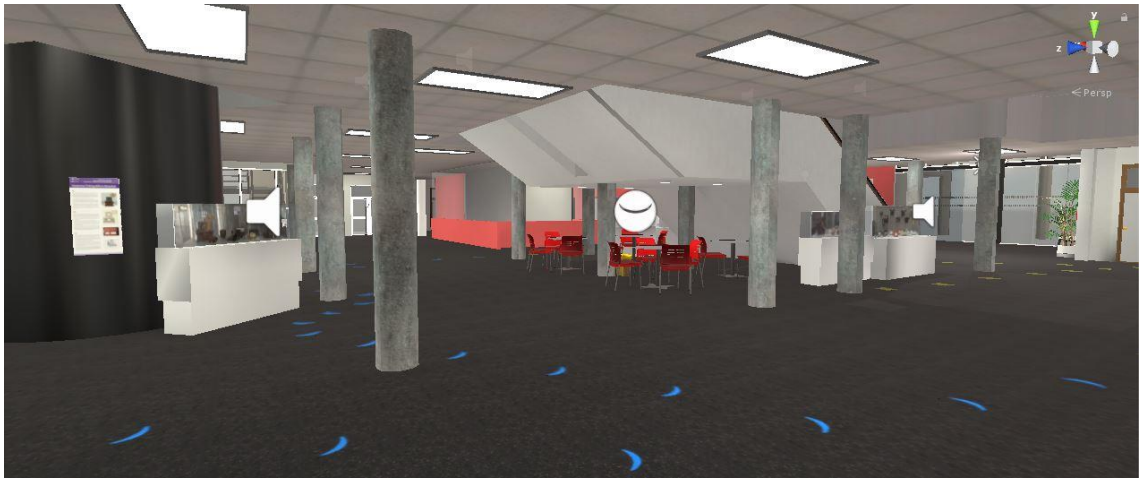

Figura 113. Vista general del museo, Planta Baja.

Figura 114. Vista general del museo, Planta Baja.

Figura 115. Vista general del museo, Planta Baja.

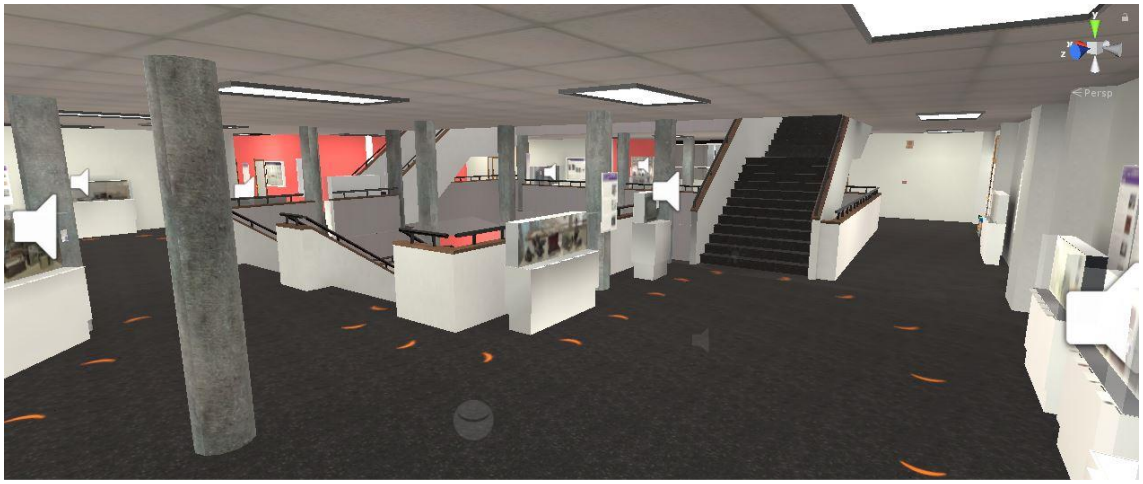

Figura 116. Vista general del museo, Planta Primera.

Figura 117. Vista general del museo, Planta Primera.

Figura 118. Vista general del museo, Planta Primera.

5.2 Posibilidad de ampliación

El proyecto está diseñado y estructurado para ser fácilmente entendido por cualquiera que se disponga a seguir trabajando en él. Lo más inmediato, puesto que un alumno está trabajando en ello, sería añadir los modelos en 3D de los objetos expuestos representados actualmente mediante fotografías. Además, como el museo está en continua evolución, sería posible y adecuado ir añadiendo en la versión virtual nuevas vitrinas con nuevos objetos expuestos, cambiar la ubicación de éstos, añadir luces, incluir más información o contenido multimedia, todo dependiendo de los cambios que se vayan haciendo en el museo real.

5.3 Pruebas con usuarios

Se ha realizado una prueba con cuatro usuarios. Uno de ellos es alumno de la escuela; otro ha experimentado ya con aplicaciones de realidad virtual en gafas 3D, pero sin haber estado nunca en la escuela; y los otros dos no tienen ninguna relación con la realidad virtual, muy poca con nuevas tecnologías (ordenadores, Smartphone, etc.) y tampoco son conocedores del edificio de la ETSIT.

La prueba consistía en explicarles el funcionamiento de las gafas y los mandos y dejarles libertad para recorrer el museo a su antojo pudiendo experimentar con todo lo que éste ofrece. Al finalizar la experiencia, rellenaban un pequeño cuestionario de siete preguntas sobre su apreciación y sensación de haber “estado ahí” y de cuán real ha sido todo lo que han visto y vivido.

Tras las pruebas, se ha obtenido un resultado medio de los cuatro cuestionarios de un 8,4 sobre 10. Con solamente cuatro pruebas, es difícil sacar una conclusión certera de la sensación de realismo que ofrece esta aplicación, pero dado que la finalidad de este trabajo de fin de grado no era la de realizar estudios con sujetos, a modo de valoración se puede concluir que la aplicación presenta un grado más que aceptable de realismo, consigue trasladar al usuario al mundo virtual sin provocar mareos y alcanza el objetivo de ofrecer una experiencia agradable y enriquecedora.

Capítulo 6. Conclusiones y líneas futuras

El uso de gafas 3D y ese tipo de dispositivos en la realidad virtual es una tecnología bastante nueva y en muchos aspectos desconocida. Hay mucho que avanzar y mejorar y eso se ha visto reflejado en la aplicación desarrollada en este TFG. Aun así, se han conseguido los objetivos propuestos.

La aplicación es totalmente intuitiva y funcional y contiene todo lo que ofrece una visita física al Museo de la Telecomunicación. Por lo cual, es posible realizarla desde cualquier lugar externo al museo siempre que se disponga de ordenador y gafas 3D. Además, el hecho de que la recreación del edificio junto con toda la exposición sea lo más fiel posible a la realidad, junto con la gran sensación de inmersión en el entorno que las gafas 3D son capaces de alcanzar, hace que el usuario pueda vivir una experiencia enriquecedora y, en muchos casos, experimentar la realidad virtual por primera vez.

Para finalizar, en cuanto al futuro más próximo, es fácil predecir un auge de este tipo de tecnología y de su aplicación no solamente a museos, arte y cultura sino a muchos otros ámbitos en la sociedad como son los videojuegos, la sanidad o la educación, uniendo diversión y entretenimiento con aprendizaje y su consecuente mejora de vida.

Bibliografía

[1] Luis Pérez Pau. *Visita virtual al Museo de las Telecomunicaciones Vicente Miralles,* junio 2015.

[2] MCN, advancing digital transformation in museums. "History," http://mcn.edu/about/history. [Online].

[3] "¿Qué es la realidad virtual?," *Mundo Virtual*. http://mundo-virtual.com/que-es-la-realidad-virtual/. [Online].

[4] "Realidad Virtual: Características, objetivos e historia. Lentes y juegos," *Tecnología&Informática*. https://tecnologia-informatica.com/realidad-virtual-caracteristicas-objetivos-historia-lentes-juegos/. [Online].

[5] Amparo Caballero. "Realidad Virtual aplicada a la salud," *innoarea projects,* enero 2017. http://www.innoarea.com/realidad-virtual-aplicada-a-la-salud/. [Online].

[6] Paula Corroto. "Museos: esto sí es adaptarse a las nuevas tecnologías", *eldiario,* marzo 2015. https://www.eldiario.es/cultura/politicas_culturales/Museos-adaptarse-tecnologia_0_370963719.html. [Online].

[7] Yeltic VR. "5 usos de la realidad virtual en museos," *Medium,* agosto 2016. https://medium.com/yeltic-vr/5-usos-de-la-realidad-virtual-en-museos-f0e945cb3390. [Online].

[8] Juan Manuel García Campos. "Google Art Project: visitas virtuales en 360º a los mejores museos del mundo," *La Vanguardia,* Febrero 2011. https://www.lavanguardia.com/enlaces-del-dia/20110201/54110099760/google-art-project-visitas-virtuales-en-360-a-los-mejores-museos-del-mundo.html. [Online].

[9] "The British Museum with Google," *The British Museum*. https://www.britishmuseum.org/with_google.aspx. [Online].

[10] "Museum Virtual Tours," *National Museum of Natural History*. https://naturalhistory.si.edu/about/virtual-tour. [Online].

[11] Ana Laura Peña Aguilar. "Google Arts & Culture, una herramienta para conocer el patrimonio cultural mundial," *InfoTecarios,* mayo 2018. http://www.infotecarios.com/google-arts-culture-una-herramienta-para-conocer-el-patrimonio-cultural-mundial/#.XRuDAegzaUl. [Online].

[12] Google Arts & Culture. https://artsandculture.google.com/. [Online].

[13] Google Earth. https://www.google.com/earth/ . [Online].

[14] Google Maps. https://www.google.com/maps . [Online].

[15] "Unity Manual," *Unity Documentation.* https://docs.unity3d.com/Manual/index.html . [Online].

[16] "Unity Scripting API," *Unity Documentation.* https://docs.unity3d.com/ScriptReference/index.html . [Online].

[17] Unity Asset Store. https://assetstore.unity.com/ . [Online].

[18] Oculus Developers. https://developer.oculus.com/ . [Online].

Anexo

Tutorial y ejemplo de uso

Al iniciar la aplicación, se sitúa al usuario en la entrada del edificio, al inicio del itinerario de visita al museo. Aunque se recomienda seguir dicho itinerario, marcado en el suelo, el usuario es libre de moverse por donde quiera.

Figura 119. Inicio de la visita virtual al iniciar la aplicación.

Con el joystick del mando controlador izquierdo es posible avanzar y retroceder en la dirección en la que se mira. Para cambiar de dirección, basta con girar suavemente la cabeza mientras se usa el joystick mencionado. También es posible activar la opción de girar con el joystick del mando derecho, aunque no es aconsejable porque puede provocar mareos a causa de la anti-naturalidad de ese movimiento para el cerebro humano; por tanto, esa opción está desactivada por defecto. Además, los mandos simulan las manos del usuario, las cuales podrá ver en el entorno virtual y mover y gesticular de forma muy intuitiva. El resto de los botones no tienen ninguna función.

Figura 120. Mandos controladores Oculus Touch.

En cuanto a la exposición, cada vitrina dispone de un cartel explicativo y cada objeto expuesto tiene su cartela identificativa. Además, algunas vitrinas cuentan con una pegatina de un código QR y el texto "REALIDAD AUMENTADA MUSEO" (Figura 40) en el cristal, la cual al tocar con el dedo índice reproduce un sonido o un vídeo ilustrativo sobre el objeto al que se refiere.

Figura 121. Ejemplo de reproducción de un vídeo ilustrativo al tocar el código QR.